docsvision

Web-клиент

Версия и дата сборки: 6.1, 2026-11-19

Web-клиент предоставляет возможность работать с карточками системы Docsvision и осуществлять основные операции с электронными документами в веб-браузере на персональном компьютере и на мобильном устройстве.

Web-клиент является веб-приложением с серверной и клиентской частями, функциональные возможности которых могут быть расширены с помощью дополнительных разрабатываемых компонентов.

Серверная часть Web-клиента:

- Отвечает за обработку клиентских запросов.
- Взаимодействие с Docsvision, осуществляемое через API Docsvision.
- Предоставляет серверную бизнес-логику.
- Реализует WebAPI.
- Содержит серверный кэш.

Клиентская часть Web-клиента обеспечивает функционирование пользовательского интерфейса Web-клиента; отвечает за взаимодействие с серверной частью; предоставляет клиентскую бизнес-логику карточек.

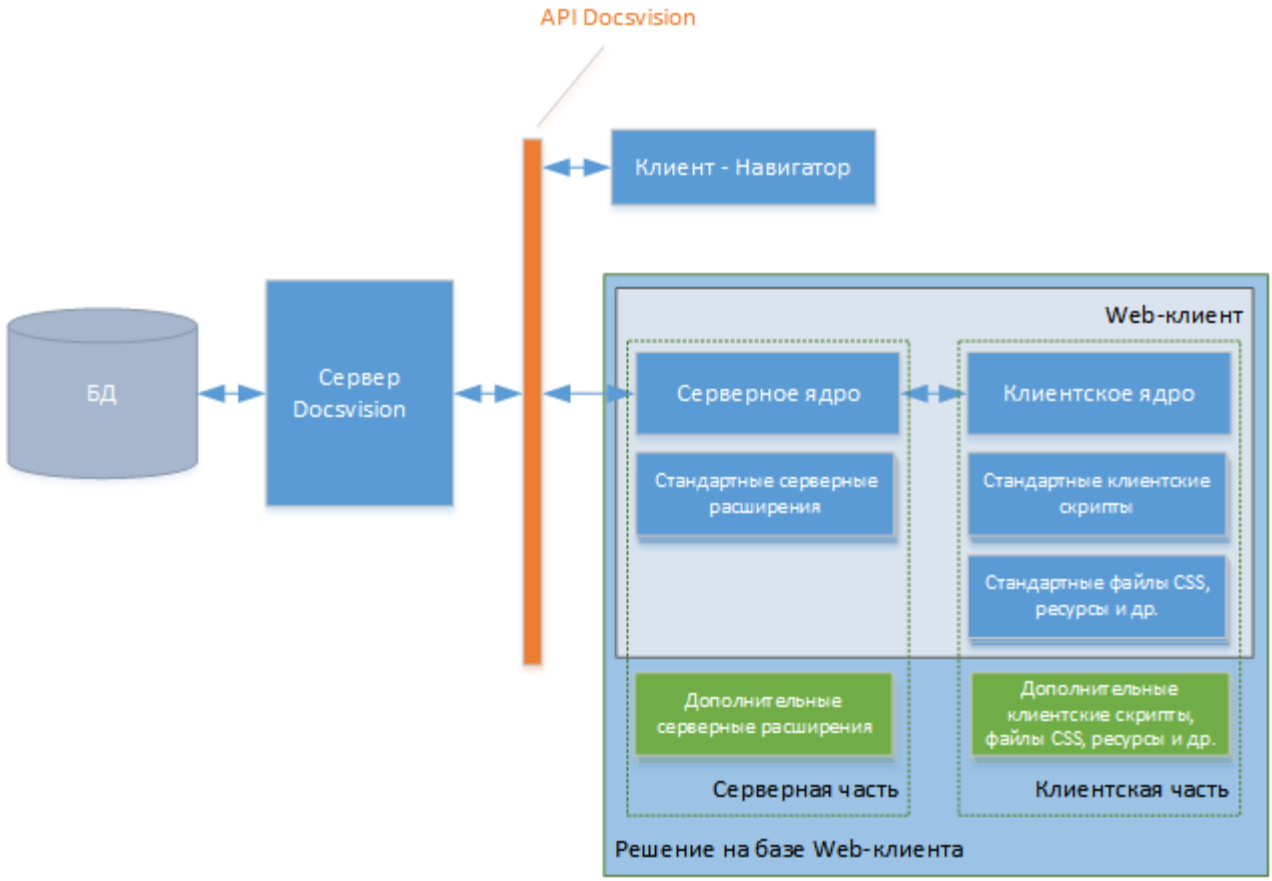


Рисунок 1. Схема взаимодействия Web-клиента с системой Docsvision

Модуль Web-клиент предназначен для применения в организациях, в которых используется СЭД Docsvision.

Функции Web-клиента

Web-клиент **не является автономным**, он обеспечивает выполнение следующих функциональных задач:

Работа с документами:

- Создание и просмотр документов.
- Просмотр в веб-браузере файлов офисных и графических форматов, приложенных к документам.
- Согласование документов с использованием расширенных маршрутов согласования приложения *Конструктор согласований*.

Работа с заданиями:

- Создание, просмотр и отправка на исполнение заданий.
- Исполнение или делегирование поступивших на исполнение заданий.
- Просмотр приложенных к заданиям файлов офисных и графических форматов в веб-браузере.
- Создание подчинённых заданий и групп заданий.

Поиск карточек:

- В выбранной папке.
- По всей базе данных с помощью полнотекстового поиска.

Работа со справочниками:

- Справочник контрагентов.
- Справочник сотрудников.
- Справочник номенклатуры дел.
- Справочник полномочий.



Web-клиент не поддерживает работу со справочником категорий.

[1] Установить. Настраивать не нужно.

Конструктор Web-разметок

Конструктор Web-разметок — программа для настройки разметок модуля Web-клиент.

Программа Конструктор Web-разметок предназначена для применения в организациях, в которых используется система Docsvision и модуль *Web-клиент*.

Возможности программы Конструктор Web-разметок:

- Настройка разметок карточек, дерева папок, дашборда, а также других элементов Web-клиента, с помощью встроенного редактора.
- Настройка условий использования разметки в зависимости от типа устройства, роли сотрудника, состояния карточки, а также других условий. Список возможных условий выбора разметки определяется её назначением.
- Настройка Решений — комплект настроек, содержащий разметки, данные локализации и условия выбора разметок.
- Экспорт/импорт Решений и др.
- Создание расширенных маршрутов согласования.
- Использование сценариев (язык JavaScript) для обработки событий разметки.
- Локализация разметок.
- Разработка новых элементов управления для настройки разметок.
- Разработка серверных расширений для получения дополнительной серверной бизнес-логики.

Администрирование

Раздел администрирование содержит инструкцию по установке Web-клиента, описание функций *администратора Docsvision*, инструкцию по настройке дерева папок и дашборда, а также описание других задач по администрированию:

1. Установка и настройка содержит инструкцию по установке и первоначальной настройке Web-клиента, инструкцию по обновлению и настройке кластера Web-клиента.
2. Функции администратора содержит описание программы Панель управления Web-клиентом, перечислены действия, выполняемые администратором Docsvision.

Расширение возможностей

Расширение функциональных возможностей

Функциональные возможности серверной части Web-клиента могут быть увеличены с помощью серверных расширений. С помощью данного типа расширений в сервер Web-клиента могут быть добавлены новые методы WebAPI (в т. ч. реализующие серверную бизнес-логику карточек); дополнительные данные, необходимые для формирования пользовательского интерфейса Модуля (в т. ч. новых элементов управления); методы, обеспечивающие возможность создания новых типов карточек из меню создания карточек; и др.

Функциональные возможности клиентской части Web-клиента могут быть увеличены с помощью клиентских расширений и клиентских скриптов. В клиентских расширениях могут быть реализованы новые общие функции, доступные всем клиентским компонентам Модуля, а также клиентские компоненты элементов управления. Клиентские скрипты добавляют в Модуль функции-обработчики для событий элементов разметок, настраиваемых в программе Конструктор Web-разметок.

Расширение возможностей программы Конструктор Web-разметок

Конструктор Web-разметок также предоставляет возможность расширять свою функциональность с помощью дополнительных компонентов. С помощью расширений программы Конструктор Web-разметок могут быть добавлены новые типы элементов управления и связанные с ними сущности.



Инструкция по разработке разметок приведена в Руководстве пользователя программы Конструктор Web-разметок.

Общие сведения о Web-клиенте

Уровень подготовки пользователя

Пользователи модуля Web-клиент

Web-клиент является основным рабочим местом пользователя Docsvision.

Пользователи модуля Web-клиент должны:

- Обладать навыками работы с веб-браузерами.
- Свободно осуществлять базовые действия в операционной системе клиентского устройства.
- Обладать знаниями предметной области.
- Обладать навыками работы с модулем *Windows-клиент*, а также с другими модулями и приложениями, которые установлены в системе Docsvision (желательно).
- Пройти обучение по работе с модулем Web-клиент.

Администраторы модуля Web-клиент

Администратор модуля устанавливает Web-клиент и выполняет первоначальную настройку,

Администратор Web-клиента должен:

- Обладать навыками администрирования поддерживаемых операционных систем.



При установке Р7-сервер для использования функции предварительного просмотра на Astra Linux требуется минимальная версия 1.7.3.

  - Astra Linux Special Edition 1.7.
  - Astra Linux Special Edition 1.8.
  - РЕД ОС 8 x86_64.
  - РЕД ОС 8 aarch64 (Байкал-М).
  - Альт Сервер 10.
  - Альт Сервер 11.

- Ubuntu 24.04.
- Microsoft Windows Server 2012.
- Microsoft Windows Server 2012 R2.
- Microsoft Windows Server 2016.
- Microsoft Windows Server 2019.
- Microsoft Windows Server 2025.
- Microsoft Windows 10 (Pro, Enterprise).

- Иметь опыт работы с json-структурой файлов, а также обладать достаточным знанием терминала Linux для запуска требуемых компонентов.
- Должен уметь установить и настроить необходимое ПО, указанное в разделе "Программное обеспечение для работы с электронной подписью".
- Быть способным распространить на клиентские рабочие места настройки адреса и порта, используемые модулем Web-клиент.
- Обладать навыками администрирования других модулей и приложений, установленных в системе Docsvision.

Пользователь программы Конструктор Web-разметок (инженер)

Основной обязанностью пользователя программы Конструктор Web-разметок (инженера) является создание и настройка решений и разметок.

Пользователи программы должны:

- Иметь опыт работы с персональным компьютером на уровне квалифицированного пользователя.
- Предпочтительно иметь опыт написания скриптов на языке JavaScript и создания CSS стилей.

После настройки и распространения решения его поддержка выполняется инженером Docsvision без привлечения разработчиков.

Разработчики расширений для Web-клиент

Конструктор Web-разметок позволяет объединять настройки разметок, условия выбора и данные локализации в решение, которое может быть выгружено для последующего распространения. При необходимости изменения стандартных решений для системы могут быть разработаны собственные решения на базе Web-клиента (опционально).

Для разработки решений необходимо:

- Обладать опытом разработки на языках TypeScript и C#, создания CSS стилей.
- Обладать навыками разработки веб-приложений с использованием библиотеки React.
- Иметь опыт разработки Решений на платформе Docsvision (желательно).
- Ознакомиться с разделами "Руководство по разработке решений" и "Руководство разработчика Docsvision".

Мобильная и полная версия Web-клиента

Интерфейс Web-клиента соответствует типу устройства, на котором выполняется работа: на персональном компьютере — полная версия; на мобильных устройствах — мобильная. Пользователь может переключить вид интерфейса из Основного меню.

Мобильная версия предоставляет все функции полной версии с некоторыми ограничениями, которые обусловлены возможностями мобильного устройства: специальное форматирование страниц Web-клиента для размещения элементов на небольшом экране, отсутствие функции сканирования документов и некоторые другие.

Функции Web-клиента, рассматриваемые в данном документе, соответствуют полной версии. Если реализация или способ использования функции в мобильной версии отличается, или данная функция не предоставляется, это будет отмечено отдельно.

Если для полной версии написано "щелкнуть" или "нажать", в мобильной версии нужно "коснуться" указанного элемента.

Сведения о составных частях модуля

Модуль Web-клиент представлен следующими компонентами:

- Служба Web-клиента и основные сборки, необходимые для работы модуля.
- Схемы метаданных для хранения в БД.
- Общие сборки Web-клиента, необходимые для взаимодействия с другими модулями.
- Комплект утилит Docsvision Web-клиент Toolkit поставляет программу

Конструктор Web-разметок, предназначенную для настройки разметок (интерфейса пользователя) Web-клиента. Также включает утилиту WebToolBuilder, предназначенную для сборки и настройки *DVWebTool*. *DVWebTool* позволяет пользователям Web-клиента открывать файлы в офисном редакторе и сканировать документы на компьютерах с ОС Windows.

- DvSupService — выполняет в ОС Linux функции, аналогичные *DVWebTool*. Утилита не входит в состав "Web-клиент Toolkit" и устанавливается на клиентские компьютеры Linux отдельно.

Взаимодействие Web-клиента с DVWebTool (DVSupService)

- *DVWebTool* и *DVSupService* выполняются на пользовательском компьютере и предоставляют возможность запускать определённые процессы с правами **текущего пользователя**.
  - *DVWebTool* предоставляет Web-клиенту доступ к окружению пользователя: установленным программам, локальным и сетевым устройствам.

    DVWebTool будет использоваться, если компьютер пользователя работает под управлением ОС Windows. Инсталлятор которой будет скачан и запущен автоматически. Установка и запуск *DVWebTool* описаны в разделе "Установка и запуск программы DVWebTool".

  - *DVSupService* предоставляет Web-клиенту доступ к окружению пользователя: установленным программам.

    DVSupService будет использоваться, если компьютер пользователя работает под управлением ОС Linux. Установка и запуск *DVSupService* описаны в разделе "Установка и запуск службы DVSupService".



- В одной ОС утилита *DVWebTool* может быть запущена одновременно только от одного пользователя.
- *DVSupService* запускается от имени суперпользователя. Если на компьютере работает несколько пользователей, службу нужно установить один раз.

Взаимодействие компонентов организовано следующим образом:

1. Пользователь Web-клиента инициирует операцию, требующую запуска процессов на компьютере пользователя. Например, вызов офисного приложения для автоконсолидации версий документа.

2. Веб-браузер пользователя подключается по локальному адресу к порту `5413`, который по умолчанию прослушивает программа *DVWebTool*. Подключение осуществляется по протоколу `WebSocket`.
3. Веб-браузер вызывает требуемую функцию программы *DVWebTool*. Например, запускает офисное приложение в режиме автоконсолидации документа.
4. Программа *DVWebTool* выполняет требуемую задачу — открывает офисное приложение, используемое по умолчанию.

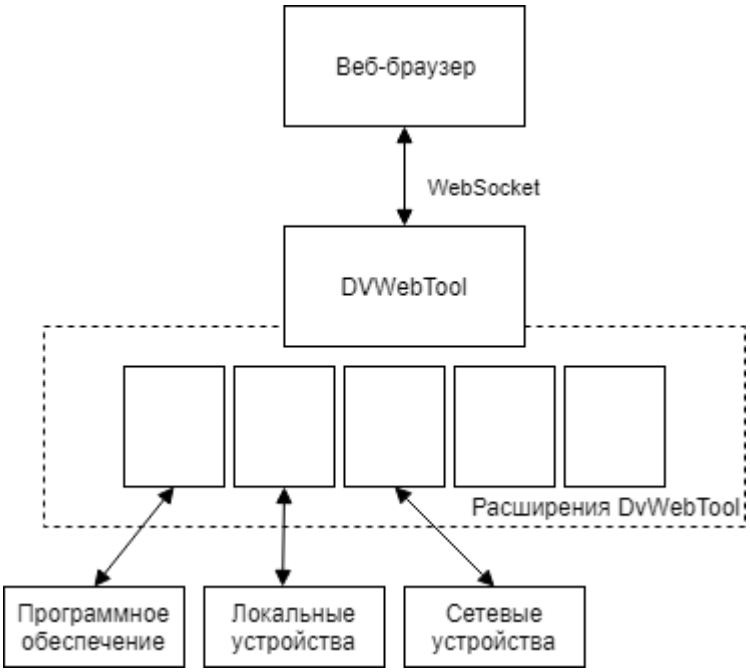


Рисунок 2. Схема взаимодействия программы DVWebTool с другими компонентами

Программа *DVWebTool* обеспечивает базовые функции взаимодействия с веб-браузером, функции для работы с окружением реализованы в расширениях программы. Расширения *DVWebTool* могут вызывать функции программ с компьютера пользователя или устройств. Расширения *DVWebTool* могут также выполнять другие операции, например, взаимодействовать с сетевыми ресурсами. Для реализации новых функций могут быть разработаны собственные расширения.

Поведение при запуске

По умолчанию программа запускается при запуске системы и не закрывается после выполнения действия. Данное поведение можно изменить, см. "Настройки запуска инструментов".

Поддерживаемые способы аутентификации

Web-клиент поддерживает аутентификацию Basic и Kerberos, аутентификация с помощью расширений аутентификации. Другие способы аутентификации не поддерживаются.



NTLM-аутентификация не поддерживается в Docsvision версии 6.1.

Необходимые ресурсы

Web-клиент

Требования для установки модуля

Информация к техническому, программному обеспечению, системе Docsvision для установки модуля и прочим необходимым ресурсам.

- Необходимое техническое обеспечение для Web-клиента
- Необходимое программное обеспечение
- Требования к модулям Docsvision
- Больше подробностей в навигационном меню слева…

Необходимое техническое обеспечение для Web-клиента

Ниже приведены минимальные аппаратные требования сервера Docsvision.

Обратите внимание, требования приведены, исходя из *нормальной* нагрузки от работы пользователей и должны занимать не более 80% доступных ресурсов системы.

Если на компьютере используется дополнительное ПО и/или модули Docsvision, при повышении нагрузки, повышении активности пользователей, усложнении бизнес-сценариев, использовании расширений и дополнений Web-клиента, добавьте требования этого ПО и/или модулей Docsvision к требованиям, указанным для модуля (в первую очередь оперативную память).

Таблица 1. Минимальные требования в зависимости от предполагаемого числа одновременно работающих пользователей[1 - Функции Web-клиента]:

Число пользователей	Аппаратные требования
до 50	**Процессор:** 2,5 ГГц и выше, 4 ядра. **Оперативная память:** от 6 Гб. **Система хранения данных:** от 12 Гб свободного места на диске. Рекомендуется использовать высокоскоростной HDD, массив или твердотельный накопитель.
Более 50	Дополнительное ядро процессора на каждые 50 дополнительных активных пользователей. Дополнительные 100 Мб оперативной памяти на каждого дополнительного активного пользователя. Дополнительные 200 Мб в системе хранения данных на каждого дополнительного активного пользователя.

Требования к клиентским рабочим местам

Следующие требования предъявляются к рабочему месту пользователя при работе на персональном компьютере.

Оборудование	Аппаратные требования
Персональный компьютер пользователя	**Процессор:** 2 ГГц и выше, Intel Core i3 или аналоги (двухъядерные). **Оперативная память:** от 2 Гб. Для запуска веб-браузера и открытия одной страницы Web-клиента должно быть свободно 512 Мб оперативной памяти.
Дополнительное оборудование	Для осуществления сканирования необходим WIA или TWAIN-сканер.

Необходимое программное обеспечение

Требования к серверу

В кросс-платформенной версии Docsvision сервер может работать под управлением следующих ОС:

- Astra Linux Special Edition 1.7.
- Astra Linux Special Edition 1.8.
- РЕД ОС 8 x86_64.
- РЕД ОС 8 aarch64 (Байкал-М).
- Альт Сервер 10.
- Альт Сервер 11.
- Ubuntu 24.04.
- Microsoft Windows Server 2012.
- Microsoft Windows Server 2012 R2.
- Microsoft Windows Server 2016.
- Microsoft Windows Server 2019.
- Microsoft Windows Server 2025.
- Microsoft Windows 10 (Pro, Enterprise).



При установке Р7-сервер для использования функции предварительного просмотра на Astra Linux требуется минимальная версия 1.7.3.

- Microsoft Windows Server 2012.
- Microsoft Windows Server 2012 R2.
- Microsoft Windows Server 2016.
- Microsoft Windows Server 2019.
- Microsoft Windows Server 2025.
- Microsoft Windows 10 (Pro, Enterprise).

Программное обеспечение:

- ASP.NET Core Runtime версии 8.0 — можно загрузить с сайта Microsoft.
- Только для сервера под управлением Linux:
  - Криптосервис необходим для проверки всех подписей и формирования CAdES-BES и CAdES-X Long Type 1.
- Если сервер Docsvision работает под управлением Windows, проверка и формирование подписи работает без криптосервиса и лицензии на него, но в таком случае отображение штампа времени будет недоступно.

Для работы функций **предпросмотра** *и* **автоконсолидации***:*

- Р7-Офис. Сервер документов/OnlyOffice с соответствующей лицензией, подробнее см. в документации по администрированию модуля "Настройка "Р7-Офис. Сервер документов"".

Требования к клиентским рабочим местам

Веб-браузер:

- Microsoft Edge 109 и выше (на базе Chromium).
- Google Chrome 109 и выше (для ОС Windows и для ОС Android).
- Google Chromium 114.0.5735 и выше (для ОС Linux).
- Яндекс.Браузер 23 и выше (для ОС Windows).
- Safari 16 и выше.
- Chromium-Gost 117.0.5938.132 и выше.

Для открытия файлов, приложенных к карточкам:

- Любой иной офисный пакет — для использования функции автоконсолидации.
- DVWebTool или DVSupService.
- ASP.NET Core Runtime версии 8.0 и выше — для работы DVWebTool и DVSupService. Можно загрузить с сайта Microsoft.
- .NET Desktop Runtime (x86/x64) версии 8.0 и выше — для работы DVWebTool. Можно загрузить с сайта Microsoft.

Для работы с ЭП:

- Программное обеспечение для формирования усиленной подписи.

Программное обеспечение для работы с электронной подписью

Web-клиент поддерживает два типа подписи: простая и усиленная. Формирование простой подписи доступно без дополнительных настроек, но должно быть предусмотрено сценарием подписания.

Для формирования усиленной подписи с использованием сертификатов ЭП должны быть установлены компоненты КриптоПро. Необходимые компоненты перечислены далее.



Для работы с электронной подписью необходимы компоненты Криптопро. По вопросам их приобретения и лицензирования

обращайтесь в компанию ООО "КРИПТО-ПРО".

Общие требования

- В справочнике *Системные настройки*, в разделе **Базовые объекты › Подписи и шифрование** должен быть выбран криптопровайдер и алгоритм подписи, соответствующие используемому сертификату ЭП. В частности, для работы с сертификатами КриптоПро должен быть выбран провайдер КриптоПро (*Crypto-Pro GOST R 34.10-2012 Cryptographic Service Provider*).



Дополнительная информация по настройке подписания и шифрования приведена в документации по администрированию модуля Базовые объекты, раздел "Подписи и шифрование".

- КриптоПро myDSS 2.0 или DSS Client для работы с облачной подписью на мобильных устройствах.
- Криптосервис на сервере Docsvision и необходимые для его работы лицензии и компоненты:



Функциональные возможности криптосервиса, необходимые для работы компоненты и лицензии см. в руководстве по настройке системы, на странице "Сервис обслуживания ЭП".

  - Сервер под управлением Linux требует Криптосервис для проверки и формирования CAdES-BES и CAdES-X Long Type 1.
  - Если сервер Docsvision работает под управлением Windows, проверка и формирование подписи работает без криптосервиса и лицензии на него, но в таком случае отображение штампа времени будет недоступно.

Использование неквалифицированных сертификатов

Требования к клиентским рабочим местам:

- Операционная система: Windows или Mac OS, Linux (версии см. в требованиях КриптоПро CSP).
- Веб-браузер из списка поддерживаемых, см. в пункте "Необходимое программное обеспечение").
- КриптоПро ЭЦП Browser plug-in версии 2.0. (Ссылка для загрузки).
- Неквалифицированный сертификат пользователя. Сертификат должен быть установлен в хранилище *Личное* на уровне *Текущий пользователь*.

- Опционально, для работы с подписью на мобильных устройствах. Приложение КриптоПро myDSS 2.0 или DSS Client. Приложения обладают схожей функциональностью, рекомендуется использовать DSS Client как официальное приложение КриптоПро.

Использование квалифицированных сертификатов

Требования к серверу:

- Корневой сертификат удостоверяющего центра, выпустившего сертификат пользователя.[1] Сертификат должен быть установлен в хранилище Криптосервиса.
- Только для сервера под управлением Linux:
  - Криптосервис необходим для проверки всех подписей и формирования CAdES-BES и CAdES-X Long Type 1.
- Если сервер Docsvision работает под управлением Windows, проверка и формирование подписи работает без криптосервиса и лицензии на него, но в таком случае отображение штампа времени будет недоступно.
- На сервере Docsvision требуется наличие библиотеки `libcapi20.so`, входящей в состав КриптоПро CSP версии 5.0 и выше с соответствующей лицензией для работы расширения аутентификации ЕСИА и корректного отображения меток усиленной подписи.
- При вычислении хэша подписываемого файла на сервере Web-клиента:

  Получите библиотеку `CryptoPro.Security.Cryptography.dll` и соответствующую лицензию КриптоПро на ПО КриптоПро .Net LibCore 2025.7.21[2]. Сборку поместите по адресу `/usr/lib/docsvision/common` для Linux и `C:\Program Files\Docsvision\Common` для Windows.



> Библиотека `CryptoPro.Security.Cryptography.dll` из состава КриптоПро .NET LibCore требуется для входа через ЕСИА, вычисления хэша подписываемого файла на сервере Web-клиента и подписания через модуль интеграции с УЦ Контур. Если эти функции не используются, устанавливать библиотеку и приобретать лицензию КриптоПро .NET не требуется.

Требования к клиентским рабочим местам:

- Операционная система: Windows или Mac OS, Linux (версии см. в требованиях КриптоПро CSP).

- Веб-браузер: Google Chrome, Яндекс.Браузер или Microsoft Edge на Chromium (поддерживаемые версии см. в пункте Необходимое программное обеспечение).
- Компоненты криптопровайдера КриптоПро:
  - КриптоПро CSP версии 5.0 и выше.
  - КриптоПро ЭЦП Browser plug-in версии 2.0.
  - CryptoPro Extension for CAdES Browser Plug-in версии 1.2.4 и выше (ссылка для загрузки).
    a. Если расширение плагина имеется в списке установленных, но отключёно, необходимо самостоятельно включить его в браузере.
    b. Если расширение отсутствует в списке или недоступно в магазине приложений, вы можете установить расширение, воспользовавшись инструкцией на сайте КриптоПро. Вы также можете продолжить работу с электронной подписью в браузерах Chromium-Gost или Яндекс Браузер с полной поддержкой необходимой функциональности.
  - Для создания усовершенствованной подписи Cades-X-Long-Type-1 могут потребоваться дополнительные компоненты Криптопро (TSP Client и OCSP Client).
- См. также требования для модуля "Базовые объекты".
- Квалифицированный сертификат пользователя. Должен быть установлен в хранилище *Личное* на уровне *Текущий пользователь*.



Шаги, необходимые для настройки КриптоПро в ОС Mac OS можно найти в документации КриптоПро.

По вопросам получения компонентов криптопровайдера КриптоПро следует обращаться в компанию ООО "КРИПТО-ПРО".

Требования к модулям Docsvision

Общие требования к системе Docsvision

- Данная версия Web-клиента не поддерживает работу с Docsvision версии 5.5 и ниже.
- Для корректной установки Web-клиента БД Docsvision должна быть обновлена до последней версии (см. администрирование модуля Платформа,

раздел "Обновить базу данных").

- Рекомендуется перевести БД системы Docsvision на работу с расширенными метаданными. Web-клиент может работать с БД с динамическими метаданными, но динамические метаданные будут недоступны в модуле.
- Если в Web-клиенте планируется работать с функциями модуля *Управление архивом*, указанный модуль обязательно должен быть установлен на сервере Docsvision перед установкой Web-клиента.
  Поддерживаемая версия модуля *Управление архивом* указана в пункте ниже.

Модули системы Docsvision[1]:

1. Серверная часть модуля *Платформа* версии 6.1 и выше.
2. Серверная часть модуля *Базовые объекты* версии 6.1 и выше.



Для сценариев работы с МЧД потребуется сборка версии 6.1, сборка 176 и выше.

3. Серверная часть приложения *Управление документами* версии 6.1 и выше.
4. Серверная часть модуля *Управление процессами* версии 6.1 и выше.
5. Серверная часть приложения *Конструктор согласований* версии 6.1 и выше.
6. Серверная часть модуля *Управление архивом* версии 6.1 и выше.
   Необязательный компонент. Обязателен, если требуется поддержка работы с функциями модуля Управление архивом.

Требования к системной учётной записи модуля

Системная учётная запись Web-клиента, должна удовлетворять следующим требованиям:

- Быть включена в группу **DocsVision Administrators** в конфигурационном файле Сервера приложений (модуля Платформа).
- Если учётные записи пользователей Docsvision являются доменными учётными записями, то системная учётная запись также должна быть доменной. Доменная учётная запись также должна иметь доступ к атрибуту ***tokenGroupsGlobalAndUniversal*** пользователей Web-клиента.



Если запуск Web-клиента завершается с ошибкой `System.Runtime.InteropServices.COMException (0x8007200A): Указанное значение или атрибут службы каталогов не существует`, предоставьте

системной учётной записи права на чтение всех атрибутов домена: разрешение *Read all properties* на объекты домена.

Данное требование необходимо для корректной проверки авторизации в домене с помощью логина и пароля.

Требования установки программы Конструктор Web-разметок

Для работы с программой Конструктор Web-разметок лицензия на систему Docsvision должна содержать дополнительную опцию: *Конструктор карточек*.



Начиная с версии Web-клиента 6.1 сборки 1265 и выше, программа Конструктор Web-разметок не входит в комплект Web-клиента и не требует установки компонентов модуля. Установка выполняется из отдельного инсталлятора комплекта утилит Web-клиента — `Docsvision web-client toolkit.msi`.

Аппаратные требования

Требования совпадают с требованиями сервера Web-клиент в пункте Необходимое техническое обеспечение для Web-клиента.

Программные требования

Операционная система:

- ОС Windows.

Обязательное программное обеспечение:

1. ASP.NET Core Runtime версии 8.0. Можно загрузить с сайта Microsoft.

Требования к ЛВС

Настройки ЛВС должны допускать возможность подключений:

1. С компьютера с сервером Web-клиента к компьютеру с сервером Docsvision на адрес и порт, которые используются системой Docsvision.
2. С клиентских рабочих мест к компьютеру Web-клиента на адрес и порт, которые используются Web-клиентом.
3. Рекомендуется настроить защищённое соединение по протоколу HTTPS, см. подробнее "Подключение к Docsvision по HTTPS" в руководстве по настройке

системы.

При использовании программы *DVWebTool* также осуществляются локальные подключения (на адрес localhost) веб-браузера пользователя к программе *DVWebTool* на порт `5413` (по умолчанию), по протоколу `WebSocket`. В определённых случаях программа *DVWebTool* может осуществлять подключения к серверу Web-клиент.

Задержка передачи данных по сети между сервером Web-клиента и сервером Docsvision не должна превышать 5 мс.

Требования к лицензии

- Для работы с модулем Web-клиент лицензия на систему Docsvision должна содержать дополнительную опцию *Docsvision Thin client*. В дополнительной опции указано максимальное количество пользовательских подключений к Web-клиенту.
- Для работы с программой Конструктор Web-разметок лицензия на систему Docsvision должна содержать дополнительную опцию: *Docsvision Конструктор карточек*.
- Для работы с функцией агрегации потребуется дополнительная опция лицензии: *Docsvision Модуль Интеллектуальный поиск*. См. подробнее в разделе Настройки агрегации.
- Для работы с цитатами в гриде в режиме ленты также потребуется дополнительная опция лицензии: *Docsvision Модуль Интеллектуальный поиск*. В противном случае в режиме ленты цитаты не будут отображаться.
- Для работы с договорами лицензия Docsvision должна содержать лицензионную опцию *Docsvision Contracts*. При отсутствии лицензии пользователь не сможет создавать и открывать карточки *Договор*, *Акт*, *Дополнительное соглашение*.
- Для использования функции авторизации с помощью OpenID требуется соответствующая опция лицензии. Например, *Docsvision Коннектор к AzureAD*, *Docsvision Коннектор к ЕСИА* или *Docsvision Коннектор к ADFS*.
- Для использования кэш-сервера Redis потребуется опция лицензии *Docsvision кэш-сервер Redis*.

[0] Подразумеваются пользователи с активной сессией. Время активности может быть настроено администратором.

[1] Например, корневой сертификат КриптоПро

[2] КриптоПро .NET LibCore обозначена как "предварительная версия" на сайте КриптоПро и доступна для скачивания

на GitHub в виде файла `cryptopro.security.cryptography.2025.7.21.nupkg`. Файл формата `.nupkg` может быть распакован как архив, он содержит необходимую библиотеку, которую необходимо поместить по указанному адресу.

Изменения, обновления и исправленные ошибки

Web-клиент

Общая документация

Общая информация об изменениях, исправленных ошибках и накопительных обновлениях.

- Изменения в релизной версии
- Исправленные ошибки
- Накопительные обновления

Изменения в релизной версии

Функциональные изменения

Новый адрес подключения к серверу Web-клиента

Адрес подключения к серверу Web-клиента теперь указывается в новом формате: `http://WebC-server.domain.com:5004/`.

Новый способ установки DVWebTool

Изменён способ установки утилиты *DVWebTool*. Когда сервер системы расположен на Linux, необходимо выполнить сборку утилиты при помощи стандартных средств Docsvision, подробнее см. "Установка и запуск службы DVWebTool".

Служба DVSupService

Разработан аналог DvWebTool — служба DVSupService для открытия файлов в офисном редакторе. Служба устанавливается только в ОС Linux и позволяет открывать офисные файлы в ассоциированном редакторе. Подробнее см. "Установка и запуск службы DVSupService".

Консолидация с использованием Р7

Стандартный механизм консолидации был заменён на серверную консолидацию через Р7-Офис. Вместо использования офисного приложения на клиентском компьютере для консолидации используется Р7-Офис,

установленный на сервере. При объединении версий файлов в карточке открывается окно Р7-Офис. Подробная инструкция по настройке приведена в разделе "Настройка "Р7-Офис. Сервер документов"".

Предпросмотр с использованием Р7

Предварительный просмотр файлов в карточках теперь по умолчанию отключен. Чтобы включить предварительный просмотр, потребуется установить "Р7 Сервер документов" и настроить Web-клиент на использование HTTPS. Подробная инструкция по активации предварительного просмотра приведена в документации по администрированию модуля, раздел "Настройка "Р7-Офис. Сервер документов"".

Количество вариантов в агрегате

Количество выводимых вариантов агрегата увеличено до 100. Настройку по умолчанию можно изменить через БД, подробнее см. "Изменить количество вариантов агрегата".

ПО для формирования подписи

Обновлены требования для формирования электронной подписи. В новой версии модуля для проверки и формирования простой и усиленной ЭП требуется CSP версии 5.0 и выше, также требуется Криптосервис, см. подробнее в руководстве по настройке системы Сервис обслуживания ЭП.

Переход на .NET 6 и установка Linux

Новая версия модуля переведена на .NET 6.0 с возможностью установки на ОС Astra Linux.

Изменён процесс обновления

Процесс обновления серверного расширения Web-клиент изменён. Теперь при установке серверного расширения диалог с обновлением БД не появляется, требуется обновить БД через Консоль настройки Docsvision.

Новый формат версии сборки

Был обновлён формат версий сборки модуля, web-расширений и решений. В окне свойств инсталлятора и в окне "О программе" версия отображается в новом формате: `Мажорная.Минорная.Сборка+Хэш-коммита`, например: `6.1.1234+fbf13f3159`. В программе Конструктор Web-разметок версия решений отображается в виде `Мажорная.Минорная`, например, `6.1`.

Обновление версии модуля

Версии управляемых компонентов Docsvision (.NET) были изменены с 5.5 на 6.0. Версии неуправляемых компонентов Docsvision (C++, VB 6.0) не изменились.

Формирование подписи CAdES-T

Подпись типа CAdES-T больше не может быть сформирована в Docsvision. Проверка подписи типа CAdES-T остаётся неизменной, как и формирование подписей типов CAdES-X Long Type 1 и CAdES-BES.

Изменения в API

Изменения отсутствуют.

Исправленные ошибки

ERR-5048 (SUP-8313)	В документации отсутствовало примечание о том, что Web-клиент не поддерживает работу со справочников категорий.
ERR-5004 (SUP-8316)	В ЭУ `Строка конструктора справочников` не отображались подразделения при поиске, если флаг **Подразделение** был снят.
ERR-4990 (SUP-8358)	Исправлена ошибка, при которой в журнале подписей не отображались недействительные подписи.
ERR-4906 (SUP-8273)	При добавлении ЭУ `Файловая панель согласования` в разметку возникала ошибка `При рендеринге контрола mainBlock типа Block произошла ошибка`.
ERR-4872 (SUP-8260)	Не сохранялись изменённые настройки представления.
ERR-4868 (SUP-8252)	При атрибутивном поиске с отключенной проверкой прав и использованием не всех полей не возвращалось ни одной карточки.
ERR-4855 (SUP-8258)	При сохранении карточки и отсутствии прав на операцию редактирования в `Строка конструктора справочников` не прописывалось значение по умолчанию.

ERR-4820 (SUP-8224)	В API ЭУ `Таблица` отсутствовал метод `clear()`.
ERR-4788 (SUP-8211)	При установке значения кодом в ЭУ `Ссылка на карточку` дублировалось сохранение.
ERR-4772 (SUP-8203)	Возникала ошибка `System.IO.IOException: Файл уже существует` при попытке скачать файл с подписями.
ERR-4726 (SUP-8160)	При использовании просроченного сертификата в конфигурации ЕСИА мог происходить вход под УЗ пула.
ERR-4700 (SUP-8180)	Исправлена ошибка, из-за которой столбцы в таблице в разделе *Заместители* профиля пользователя были смещены.
ERR-4682 (SUP-8102)	Исправлена ошибка, при которой не удавалось вычислить регистратора в при запуске согласования без указания регистратора.
ERR-4670 (SUP-7974)	В API ЭУ Table отсутствовал метод `onReorderRows`.
ERR-4644 (SUP-8096)	Исправлена ошибка, вызывавшая задвоение строк, если в представлении использовался фильтр.
ERR-4642 (SUP-8083)	Исправлена ошибка, при котрой окно ввода отчета задания не увеличивалось вниз.
ERR-4631 (SUP-8147)	В ЭУ `Строка конструктора справочников` наблюдалось дёрганье прокрутки в окне выбора ЭУ, если количество записей с описанием было большим.
ERR-4574 (SUP-8087)	В документации отсутствовало уточнение о использовании поисковых словах в `ЭУ Записи справочника сотрудников`.
ERR-4547 (SUP-8105)	Возникала ошибка 503 при авторизации с использованием расширения ЕСИА и подключении по `http` с указанием порта.
ERR-4546 (SUP-8080)	Исправлена ошибка, вызывавшая некорректное отображение столбцов таблицы при использовании ЭУ `Файл`.

ERR-4544 (SUP-8062)	В карточке вида `Принятие решения по СЗ` при нажатии на кнопки **Согласовано** и **К подготовке нового цикла** которые возникало окно с ошибкой `[Error][DocsVision.Platform.WebClient.Diagnostics.Trace.TraceError] [LayoutTasks][MakeDecision]`.
ERR-4501 (SUP-8075)	Исправлена ошибка, при которой можно было открыть узел, закрытый согласно настройкам ЭУ и выбрать из него строки в ЭУ "Строка конструктора справочников", если в настройках был указан только самый низкостоящий узел.
ERR-4492 (SUP-7927)	Возникала ошибка `Internal exception, error code: SQLFailure` при открытии дашборда, если в дашборд было добавлено несколько ЭУ Виджет.
ERR-4457 (ERR-4492 (SUP-8001)	Не работал предпросмотр R7 из-за пробелов в параметрах запроса.
ERR-4452 (SUP-7995)	Обрезались страницы с альбомной ориентации при показе документа с ЭП.
ERR-4451 (SUP-8009)	Возникала ошибка доступа при открытии карточки задания, в котором отсутствовала строка секции *Согласование*.
ERR-4449 (SUP-8003)	В документации название модуля ЕСИА было указано некорректно.
ERR-4438 (SUP-7990)	При создании задания для всех пользователей выводился общий список последних пользователей.
ERR-4417	Исправлена ошибка, при которой в Справочнике сотрудников было невозможно назначить себя заместителем другого человека.
ERR-4414 (SUP-7946)	ЭУ `Записи справочника сотрудников` сохранял значение при изменении в режиме редактирования ***Редактирование***.
ERR-4413 (SUP-7936)	Поиск сотрудников по справочнику сотрудников в программе Конструктор Web-разметок всегда выполнялся по организации.

ERR-4384 (SUP-7962)	Ширина столбцов в ЭУ `Таблица` при выборе режима редактирования ***Редактирование*** оказывалась неравной.
ERR-4382 (SUP-7966)	При снятом флаге **`Отключен`** поле *Регистрационный номер* (ЭУ `Нумератор`) отображалось некорректно относительно других ЭУ по ширине.
ERR-4350 (SUP-7925)	Если в этапе согласования был добавлен вариант решения ***Добавление согласующих***, после добавления согласующего возникала ошибка `Для завершения необходимо указать итоговую версию`.
ERR-4337 (SUP-7932)	Исправлена ошибка, из-за которой не рабатывала настройка **`Показать облачные сертификаты`**.
ERR-4321 (SUP-7882)	Исправлена ошибка, вызывавшая блокировку временного файла.
ERR-4319 (SUP-7914)	Не срабатывала настройка завершения задания `ShowDIalog = IfEmpty`.
ERR-4308 (SUP-7587)	При поиске в журнал сервера записывалась ошибка `Не удается вставить повторяющийся ключ в объект dbo.dvsys_search_results_data`.
ERR-4306 (SUP-7906)	Сбрасывалось заданное дробное число в ЭУ `Число` после повторного открытия разметки в Конструкторе разметок.
ERR-4292 (SUP-7666)	Возникала ошибка `TypeError: Cannot read properties of undefined (reading 'params'` при создании подчиненного к согласованию задания.
ERR-4288 (SUP-7889)	Исправлена ошибка, при которой карточка автоматически сохранялась при добавлении файла.
ERR-4287 (SUP-7886)	При открытии ЭУ `Строка конструктора справочников` не отображался список с узлами конструктора.
ERR-4280 (SUP-7886)	Исправлена ошибка, не позволявшая скачать архив из хода согласования.
ERR-4274 (SUP-7629)	При поиске в окне добавления ссылки возникала ошибка `TypeError: Cannot read properties of undefined (reading 'cards')`.

ERR-4272 (SUP-7874)	Некорректно работала нумерация строк в ЭУ `Таблица`.
ERR-4269 (SUP-7869)	Не сохранялось значение ЭУ `Строка конструктора справочников`.
ERR-4264 (SUP-7808)	Когда флаги доступности редактирования этапов были неактивны, автором заданий являлась УЗ Службы фоновых операций.
ERR-4252 (SUP-7721)	При прокручивании мышкой окна справочника сотрудников, сотрудники переставали отображаться.
ERR-4251 (SUP-7721)	При большом количестве сотрудников в справочнике сотрудников полоса прокрутки замирала в нижней точке в результате чего не отображались новые сотрудники.
ERR-4224 (SUP-7798)	Окно с дополнительной информацией в ЭУ `Ссылки` не закрывалось при нажатии за пределами окна.
ERR-4195 (SUP-7870, SUP-7787)	Уменьшалась длина строки в ЭУ `Таблица` при добавлении настройки **`Сохранять номер последовательности`**.
ERR-4191 (SUP-7799)	В строках таблицы после редактирования отображалось некорректное значение, если ЭУ `Строка конструктора справочников` был добавлен в таблицу.
ERR-4190 (SUP-7799)	Добавлялась лишняя пустая строка, если ЭУ `Строка конструктора справочников` был добавлен в таблицу.
ERR-4167 (SUP-7612)	Идентификатор (`resource`) сервиса подписи в коде был указан некорректно.
ERR-4139 (SUP-7773)	Исправлена ошибка, при которой редактирование сотрудника или организации контрагента было недоступно для редактирования.
ERR-4117 (SUP-7758)	Возникала ошибка `Встречено неверное значение тега ASN1` при загрузке отсоединённой подписи в `base64`, когда отсутствовал заголовок CMS-пакета.
ERR-4103 (SUP-7754)	При отмене согласования требовалось указывать итоговую версию файла.

ERR-4095 (SUP-7743)	При добавлении новой версии файла в ЭУ Файл старые версии удалялись.
ERR-4089 (SUP-7720)	При установке дат в профиле пользователя отсутствовала возможность уточнить время.
ERR-4085 (SUP-7733)	При импорте разметок снимался флаг **Активно**.
ERR-4048 (SUP-7599)	В документации было некорректно указано об условиях для отображения колонки *Штамп времени* в журнале подписи.
ERR-4018 (SUP-7679)	В таблице исполнения отсутствовала возможность выделять текст во всплывающей справке
ERR-4015 (SUP-7633)	При быстром переключении между представлениями удалялись ещё используемые курсоры.
ERR-4013 (SUP-7664)	Исправлена ошибка, при которой применение связанных с операцией редактирования изменений для определенного пользователя в разметке занимало продолжительное время.
ERR-4012 (SUP-7660)	Исправлена ошибка, при которой не сохранялись настройки представления виртуальных папок.
ERR-4009 (SUP-7276)	При автоматическом обновлении представления сбрасывались настройки представления.
ERR-3999 (SUP-7626)	При использовании групповых операций скачивались не все файлы карточек.
ERR-3985 (SUP-7635)	Неверно срабатывала сортировка по убыванию для полей с дробными числами.
ERR-3984 (SUP-7540)	В полнотекстовом поиске с использованием агрегатов отображались результаты в области "Всего карточек" с учётом прав на карточку.
ERR-3978 (SUP-7519)	Возникала ошибка SQLFaliure при поиске с пробелом в справочнике сотрудников.
ERR-3960 (SUP-7599)	В документации не было указано требование к наличию компонента КриптоПро TSP Client для создания усиленной подписи.

ERR-3915 (SUP-7575)	Некорректно отображалось время завершения задания делегатом.
ERR-3889 (ERR-3915 / SUP-7555)	Не отображалась дата в параметре поисковой формы, если в поисковом запросе она была указана конкретным значением.
ERR-3887 (SUP-7541)	В настройках представления отображались скрытые системные колонки.
ERR-3885 (SUP-7545)	Файл в области предварительного просмотра оборажался с некорректной кодировкой, если в конфигурационном файле Web-клиента была активна настройка `DeclinedConvertExtensions`.
ERR-3876 (SUP-7542)	Ранее удалённый параметр verify при проверке простых подписей был возвращён.
ERR-3836 (SUP-7519)	Отсутствовала возможность выбора сотрудника при поиске внутри группы.
ERR-3835 SUP-7506)	Не отображалась иконка в ЭУ `Записи справочника сотрудников` если ЭУ был отключен.
ERR-3828 (SUP-7519 / SUP-7518)	Некоторые группы переставали отображаться при прокрутке списка элементов справочника сотрудников.
ERR-3773 (SUP-7293)	При попытке экспортировать файл с подписями, которые до этого были импортированы возникала ошибка `Illegal path`.
ERR-3759 (SUP-7469)	В разметке *Справка сотрудника* не были настроены источники данных для ЭУ `E-mail` и `Комментарий`.
ERR-3755 (SUP-7402)	При запуске согласования, если в документе не было заполнено поле *Автор*, возникала ошибка `Object reference not set to an instance of an object`.
ERR-3696 (SUP-7404)	Не удалялся файл у ЭУ `Файлы`, если активно свойство **`Обязательное`**.
ERR-3684 (SUP-7409)	В файле при предварительном просмотре не отображались печать и подписи.

ERR-3680 (SUP-7399)	В документации не было указано о том, что кнопка 🗑 может отображаться в карточке недоступной для удаления из-за сильной ссылки.
ERR-3678 (SUP-7398)	Поле `Записи справочника сотрудников` очищалось после повторного открытия и сохранения.
ERR-3674 (SUP-7376)	В документации отсутствовало примечание о том, что из дочернего к документу задания, невозможно удалить ссылку на родительский документ.
ERR-3663 (SUP-7388)	При попытке сохранить отредактированное неуникальное поле в разметке *Дела* возникала ошибка `Дело с таким индексом, названием и ответственным подразделением уже существует`.
ERR-3616 (SUP-7385)	При повторном переходе по ссылке ЭУ `Ссылка / Локация` возникала ошибка `Страница не найдена`.
ERR-3611 (SUP-7312)	Невозможно было удалить подчиненное задание в состоянии `Подготовка` или `Не начато`.
ERR-3582 (SUP-7307)	Скачивался дополнительный файл, если добавить файл с ЭП в дополнительные.
ERR-3579 (SUP-7366)	В документации отсутствовало указание о том, что необходимо экспортировать закрытый ключ, чтобы скопировать сертификат в файл в формате `X.509` в кодировке `DER`.
ERR-3567 (SUP-7188)	В документации не было указано, что если в ЭУ `Управление согласованием` и `Панель выбора исполнителей` выбор ограничен группой, то свойство **`Использовать последних`** не работает.
ERR-3566 (SUP-7324)	При сохранении данных карточки возникала ошибка: `Item is not found`.
ERR-3499 (SUP-7335)	В разметке WebFrame не срабатывали функции на событиях **`После загрузки всех ЭУ`** и **`После открытия карточки`**.
ERR-3492 (SUP-7332)	Пакет установки Web-клиента в режиме изменения не обнаруживал установленные компоненты архива.

ERR-3455 (SUP-6901)	Описание ошибки при превышении лимита подключений не было локализовано.
ERR-3429 (SUP-7311)	Сортировка в ЭУ `Подразделение контрагента` не срабатывала при поиске организации контрагента.
ERR-3425 (SUP-7211)	Методы серверного расширения не срабатывали, вместо этого продолжали вызываться стандартные методы.
ERR-3410 (SUP-7310)	В документации отсутствовала информацию о настройке `MaxPreviewFileSize` в конфигурационном файле Web-клиента.
ERR-3336 (SUP-7260)	Циклы отправки сессий в сон заканчивались ошибкой `InvalidSession`.
ERR-3335 (SUP-7284)	В строке полнотекстового поиска не удавалось ввести с клавиатуры пробел.
ERR-3329 (SUP-7277)	В ЭУ `Папка` отображалось сообщение `Выбрать папку (в текущей запрещено)`, если текущий вид был запрещён для создания в папке.
ERR-3324 (SUP-7272)	Некорректно работало округление в ЭУ `Число`.
ERR-3279 (SUP-7193)	В ЭУ `Управление согласованием` отсутствовали события сохранения согласования.
ERR-3263 (SUP-7132)	В новом гриде не сохранялась ширина колонок, если имелась группировка.
ERR-3256 (SUP-7226)	В ЭУ `Ссылки` всегда приходило пустое поле `cardKindId`.
ERR-3211 (SUP-7154)	Не запускалось согласование, если для этапов согласования не была задана настройка, разрешающая или запрещающая их исключение.
ERR-3172 (SUP-6665)	При закрытии DVWebTool во время консолидации блокировалась карточка без возможности разблокировки
ERR-3164 (SUP-7149)	Если ЭУ `Дерево исполнения` был добавлен в разметку задания, в режиме ***Кнопка*** не отображалось полное дерево исполнения.

ERR-3162 (SUP-7122)	При активном свойстве **Показывать ссылки для файлов** ЭУ Ссылки не отображался файл из связанной карточки.
ERR-3151 (SUP-7144)	В документации было неверно изложено назначение ЭУ Таблица параметров поискового запроса.
ERR-3149 (SUP-7058)	В разметке с режимом создания в ЭУ Файл отсутствовало контекстное меню для файла.
ERR-3144 (SUP-7111)	При удалении строки таблицы со скриптом для события **При удалении строки** кнопка **Удалить** для таблицы переставала работать.
ERR-3130 (SUP-7099)	В задании на консолидацию при нажатии на файл открывался основной файл, а не выбранный.
ERR-3125 (SUP-6691)	В документации была отмечена некорректная информация об отображении файлов в превью.
ERR-3082 (SUP-6990)	Не сохранялась ориентация страниц файла при печати из окна предварительного просмотра.
ERR-3060 (SUP-7058)	Не работал предварительный просмотр файла в ЭУ Файл из таблицы
ERR-3049 (SUP-7067)	Картинка в ЭУ Изображение обрезалась вне зависимости от размера отведенной области.
ERR-3028 (SUP-7059)	Выбранное значение в ЭУ Конструктор справочников не добавлялось в поле после нажатия клавиши `Enter`, если перед этим было выбрано значение из списка нажатием левой кнопки мыши.
ERR-3017 (SUP-7058)	Новая версия файла в ЭУ Файл отображалась только после обновления страницы.
ERR-2986 (SUP-6825)	При открытии файла в Microsoft Office 2013 возникала ошибка Не удается зарегистрировать данный документ. Будет невозможно установить связь между этим документом и другими документами.
ERR-2978 (SUP-7027)	Исправлена ошибка, при которой поиск по времени возвращал некорректные результаты.

ERR-2957 (SUP-6983)	Была возможность создать группу заданий с датой исполнения раньше времени создания карточки.
ERR-2943 (SUP-6665)	Объединение версий занимало длительное время.
ERR-2940 (SUP-6994)	В документации отсутствовала информация о необходимости лицензии для работы с договорами.
ERR-2939 (SUP-6991)	Исправлена ошибка, при которой инициализация предварительного просмотра файла в карточке занимала длительное время.
ERR-2916 (SUP-6965)	При отмене согласования документа БП сменял состояние документа ещё раз.
ERR-2848 (SUP-6918)	При использовании нескольких `Ссылка на карточку` с разными поисковыми запросами в одном ЭУ `Таблица`, использовался поисковый запрос первого ЭУ `Ссылка на карточку`.
ERR-2783 (SUP-6854)	После того как основной исполнитель завершал задание без замечаний, комментарий делегата отображается в области исполнителя.
ERR-2764 (SUP-6834)	Исправлена ошибка, при которой не отображалась поисковая форма в папке делегата.
ERR-2740 (SUP-7258 / SUP-6816)	Исправлена ошибка, при которой не срабатывал флаг **`Подсвечивать непрочитанные карточки`**.
ERR-2736 (SUP-6426)	При сканировании в формате `.jpeg` кнопка **Несколько страниц** не становилась недоступной.
ERR-2733 (SUP-6792)	Сотрудники, ответственные за раздел справочника номенклатуры дел не становились ответственными за подчинённые разделы.
ERR-2732 (SUP-6805)	В счетчике группировки в Web-клиенте 16 отображалось некорректное количество карточек при использовании PostgreSQL.
ERR-2731 (SUP-6754)	Некорректно отображались сотрудники при выборе из справочника сотрудников: сначала отображались сотрудники с подходящим отчеством, а не фамилией.

ERR-2729 (SUP-6426)	В отсканированном файле после сохранения присутствовали большие отступы на полях.
ERR-2697 (SUP-6769)	Не срабатывал ЭУ проверки на уникальность если у пользователя, создавшего карточку, отсутствовал доступ к другой карточке с уникальным значением.
ERR-2598 (SUP-6426)	После сканирования файла размер страницы становился больше чем А4.
ERR-2572 (SUP-6675)	Отсутствовала возможность выбрать время отсутствия при выборе периода отсутствия.
ERR-2548 (SUP-6675)	При изменении статуса сотрудника возникала ошибка `Недостаточно прав для выполнения операции`.
ERR-2526 (SUP-6671)	Если в названии карточки использовался символ `/`, в названии созданной карточки обрезалось всё, что следовало после `/`.
ERR-2467 (SUP-6603)	На мобильном устройстве Android адресная строка закрывала элементы, расположенные в нижней части экрана.
ERR-2422 (SUP-6589)	Флаг **Недоступен для выбора** в справочнике видов карточек влиял на доступность ссылок вида *Связано с (ЮЗДО)* и *Ссылается на (ЮЗДО)* для выбора в конструкторе разметок.
ERR-2257 (SUP-6450)	Возникала ошибка `Разметка не найдена или локация не определена` при открытии карточки, на вид которой отсутствовали права.
ERR-2251 (SUP-6440)	При экспорте представления с группировкой по скрытому полю в Excel экспортировался только верхний уровень группировки без карточек.
ERR-2184 (SUP-6414)	В журнале подписей не отображались подписи, сделанные в прошлых циклах согласования.
ERR-2137 (SUP-6355)	При скачивании импортированной подписи, файл ЭП назывался именем сотрудника, выполнившего скачивание.

ERR-2125 (SUP-6361)	При нажатии на ссылку, указанную в столбце представления, открывалась карточка представления, а не ссылка.
ERR-2112 (SUP-6359)	Возникала ошибка `Uncaught TypeError: Cannot read properties of undefined (reading 'data')` при переключении между версиями.
ERR-1978 (SUP-6285)	Не отображались значения полей справочника сотрудников в Web-клиенте.
ERR-1889 (SUP-6201)	Возникала ошибка `Cannot open database "DVManagement" requested by the login. The login failed. Login failed for user 'sa'. при попытке войти в Web-клиент`.
ERR-1784 (SUP-6044)	При открытии другой версии файла возникала ошибка: `Uncaught TypeError: Cannot read properties of null (reading 'getAttribute')`.
ERR-1722 (SUP-6105)	Если изменить колонку в представлении с типа *Вычисляемое поле* на *Поле раздела* в фильтрах представления в старом гриде изменения вступали в силу только после перезапуска IIS.
ERR-1635 (SUP-6044)	При переключении между версиями в меню предварительного просмотра всегда отображалась последняя добавленная или загруженная версия.
ERR-1630 (SUP-6045)	Исправлена ошибка, вызывавшая запрет создания карточки дополнительного соглашения в справочнике видов карточек.
ERR-1486 (SUP-5964)	В предварительном просмотре документа в формате `.tif` отображалась только первая страница.
ERR-1409 (SUP-5909)	В Web-клиенте 15 не работала фильтрация в папках, если в папке-делегате использовалось отличное от папки представление.
ERR-1385	Счетчик непрочитанных карточек не менялся в родительской карточке, если в дочерней папке отмечалась непрочитанной карточка с тем же поисковым запросом.

ERR-1343 (SUP-5759)	Возникала ошибка при расчёте счетчиков для папки `A card FolderCard was not loaded`.
ERR-1229 (SUP-5719)	Сессия со счётчиками продолжала быть активной после истечения времени основной сессии.
ERR-1217 (SUP-5719)	Когда сессия со счётчиками завершилась при открытии папки счётчики не пересчитывались.
ERR-1216 (SUP-5719)	После скрытия и раскрытия папки в боковом меню счетчик непрочитанных сбивался на предыдущее значение.
ERR-1208 (SUP-5766)	Сохранение карточки превышало время ожидания, если в качестве с крипта для события **`Подготовка к сохранению карточки`** указывалось: `args.cancel();`.
ERR-1072 (SUP-5706 / SUP-5370 / SUP-5694)	Наблюдалось частое завершение запросов с ошибкой под нагрузкой при использовании REST.
ERR-1056 (SUP-5678)	Не выделялась иконка ЭУ `Строка конструктора справочников` при получении фокуса по `Tab`.
ERR-1052 (SUP-5676)	Некорректно экспортировались гиперссылки из представления в формат Excel.
ERR-1024 (SUP-6185 / SUP-5658)	Не удавалось установить отдельную дату завершения для исполнителей в группе заданий с параллельным исполнением.
ERR-928 (SUP-5533)	Не сортировались группы в представлении при двухуровневневой группировке.
ERR-887 (SUP-5551)	В Web-клиенте 16 не отображалась дата завершения согласования.
ERR-855 (SUP-5517)	В новом гриде не раскрывались вложенные группы.
ERR-326 (SUP-5398)	Не заполнялось значение в ЭУ `Записи справочника сотрудников` в режиме *Редактирование по месту*.

Накопительные обновления

Администрирование модуля

Установка и настройка

Web-клиент

Установка и настройка

Информация об установке модуля и необходимых подготовительных этапах.

- Установка и конфигурация сервера Web-клиента
- Больше подробностей в навигационном меню слева…

Установка и конфигурация сервера Web-клиента



Перед установкой ознакомьтесь с разделом "Необходимые ресурсы".

Пользователь, выполняющий установку, должен обладать правами локального администратора.

Если в Web-клиенте планируется работать с функциями модуля *Управление архивом*, установите серверные компоненты модуля *Управление архивом* на компьютер с сервером Docsvision **перед** установкой Web-клиента.

Если Web-клиент уже установлен и планируется дополнительно установить модуль *Управление архивом* выполните следующие действия:

1. Установите сначала модуль *Управление архивом* на сервер Web-клиента.
2. Затем обновите Web-клиент с установкой компонентов управления архивом.



Используйте только совместимые версии модулей Управление архивом и Web-клиент. Версия 6.2 и несовместима с версией 5.5.

По вопросам совместимости Управление архивом версий 6.1 и 6.2 с Web-клиентом версии 6.1 и выше проконсультируйтесь со специалистом Docsvision.

Установка Web-клиента на сервер под управлением Linux

1. Установите серверные компоненты Web-клиента следующей командой, предварительно обновив индекс пакетов:

   Astra Linux / Альт Сервер / Ubuntu

   ```
   sudo apt-get update
   sudo apt-get install docsvision-webclient
   ```

   РЕД ОС

   ```
   sudo dnf makecache
   sudo dnf install docsvision-webclient
   ```

2. Если данный модуль устанавливается последним, запустите Консоль управления Docsvision и проследуйте инструкции в документации по администрированию модуля "Консоль управления Docsvision":

   - "Обновление БД"
   - "Подключение существующей БД"
   - "Создание БД"

Установка расширения для Консоли управления Docsvision на сервер под управлением Linux

Чтобы установить расширение Web-клиента для Консоли управления Docsvision на сервер под управлением Linux:

1. Установите компоненты Расширения Web-клиента для Консоли управления Docsvision Web-клиента для Консоли управления Docsvision следующей командой, предварительно обновив индекс пакетов:

   Astra Linux / Альт Сервер / Ubuntu

   ```
   sudo apt-get update
   sudo apt-get install docsvision-ext-webc-mc
   ```

   РЕД ОС

   ```
   sudo dnf makecache
   sudo dnf install docsvision-ext-webc-mc
   ```

Расширение Web-клиента для Консоли управления Docsvision необходимо для получения списка загруженных решений, импорта решений Web-клиента, получения списка локализаций и видов карточек, а также загрузки новых значений в базу данных.



Обратите внимание, версия устанавливаемых расширений для Консоли управления Docsvision должна быть выше или равна версии модуля, для которого оно предназначается. Например, расширение модуля Платформа для Консоли управления Docsvision должна быть одной версии с самим модулем Платформа.

2. Перезапустите службы
   - Службу **dvwebclient**.

     В ОС Linux, в 6.1 сборки 1110 и выше, служба запускается от имени УЗ Docsvision и не требует root-привилегий.

   - Службу **dvconsole**.

     В ОС Linux, в 6.1 сборки 1110 и выше, служба запускается от имени УЗ Docsvision и не требует root-привилегий.

Установка Web-клиента на сервер под управлением Windows

Чтобы установить Web-клиент на сервер под управлением Windows:

1. Запустите пакет установки `Docsvision web-client.msi`.

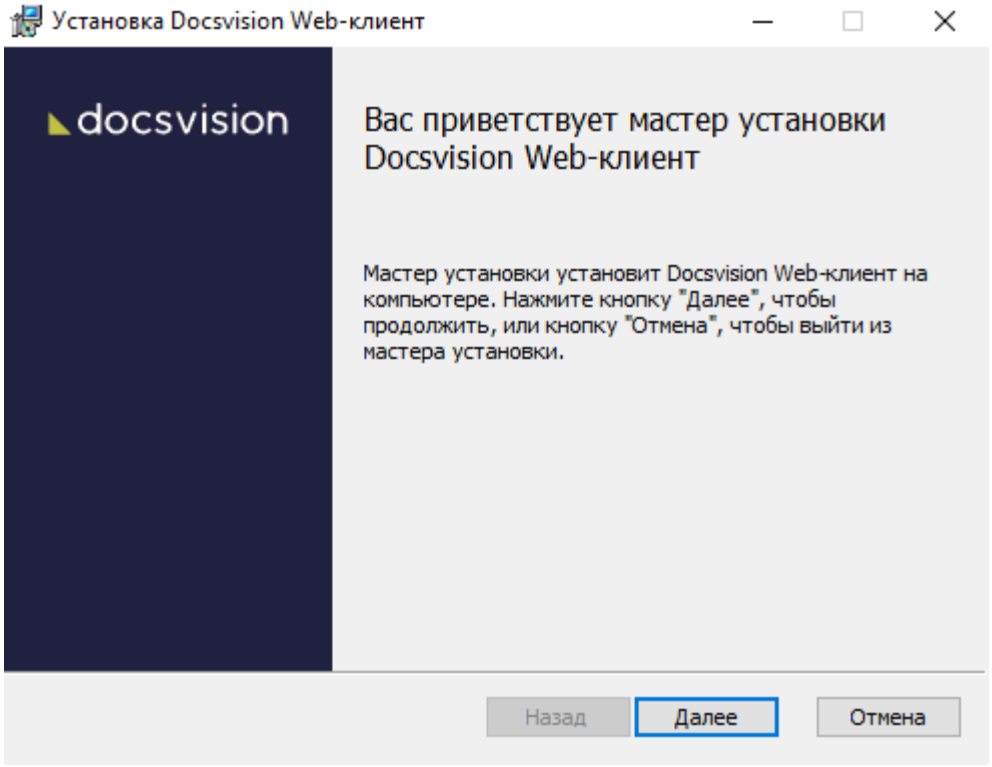


Рисунок 3. Запуск пакета установки

2. Примите условия лицензионного соглашения, чтобы продолжить установку.

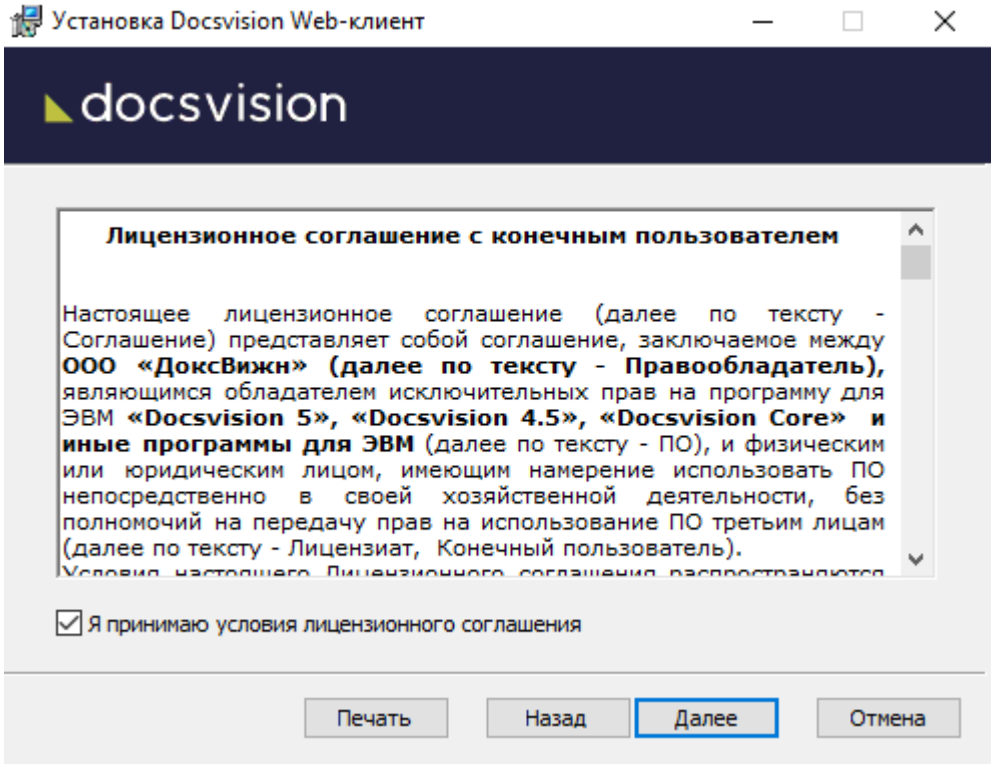


Рисунок 4. Условия лицензионного соглашения

3. Выберите компоненты, которые требуется установить, и нажмите **Далее**.

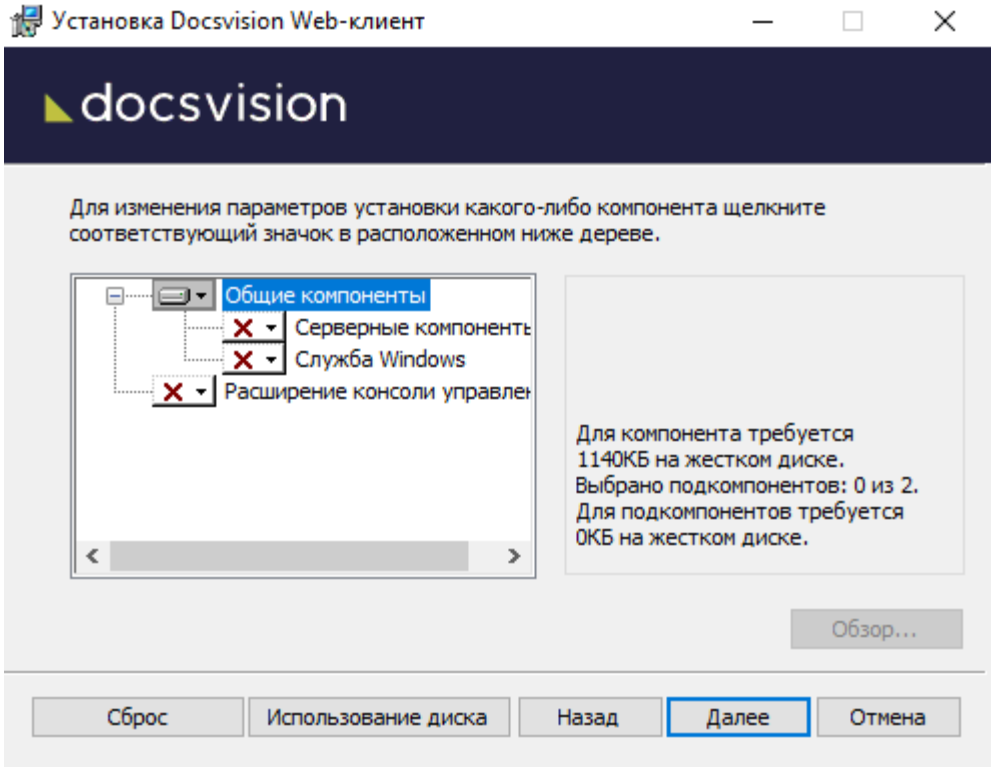


Рисунок 5. Выбор устанавливаемых компонентов

Выбирайте необходимые компоненты в зависимостми от настраиваемого сервера. На основном сервере Docsvision обязательно должны быть установлены "Серверные компоненты" модуля. Для работы модулей, устанавливаемых отдельно (например, Консоль управления Docsvision, Служба фоновых операций, Web-клиент и т.д.), достаточно устанавливать только расширения (при наличии) и общие компоненты. См. "Требования к системе Docsvision" в разделе "Необходимые ресурсы", чтобы определить связь с другими модулями.

Выбранные компоненты сохраняются при обновлении модуля, т.е. выбирать компоненты повторно при обновлении не потребуется. Если все модули установлены вместе, достаточно выбрать компоненты только при первой установке.

Общие компоненты

Обеспечивают взаимодействие с другими сервисами Docsvision, обязательны для установки на сервер Docsvision и сервер Web-клиента.

Серверные компоненты

Основные серверные компоненты Web-клиента, обязательны для установки на сервер Docsvision.

Служба Windows

Основные компоненты Web-клиента, предоставляющие базовую функциональность, может устанавливаться как на сервер Docsvision, так и отдельно от него.

Расширение консоли управления

Расширение добавляет для Консоли управления Docsvision возможности получения списка локализаций и видов карточек, загрузки новых значений в базу данных, импорт решений, обязательно для установки на сервер Консоли управления Docsvision.

Кнопка Сброс

Сбрасывает выбор компонентов на стандартные.

Кнопка Использование диска

Позволяет посмотреть свободное место на системных дисках.

4. Нажмите на кнопку **Установить** и дождитесь завершения установки.

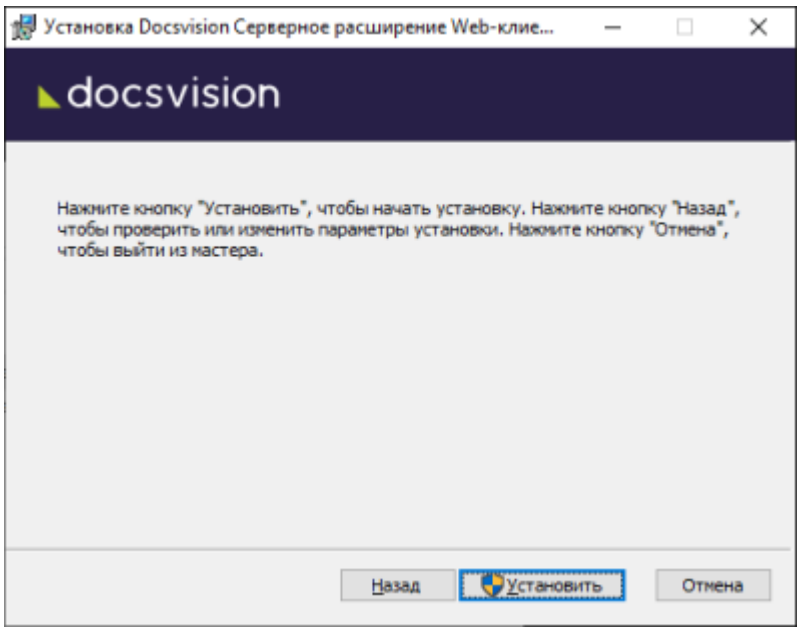


Рисунок 6. Установка расширения

5. Нажмите кнопку **Готово** для завершения работы мастера установки.

6. Если данный модуль устанавливается последним, запустите Консоль управления Docsvision и проследуйте инструкции в документации по администрированию модуля "Консоль управления Docsvision":

   - "Обновление БД"
   - "Подключение существующей БД"
   - "Создание БД"

Установка расширения для Консоли управления Docsvision на сервер под управлением Windows

Чтобы установить расширение Web-клиента для Консоли управления Docsvision на сервер под управлением Windows:

1. Запустите инсталлятор Web-клиента `Docsvision web-client.msi`, на шаге выбора компонентов выберите компонент "Расширение консоли управления" для Web-клиента:

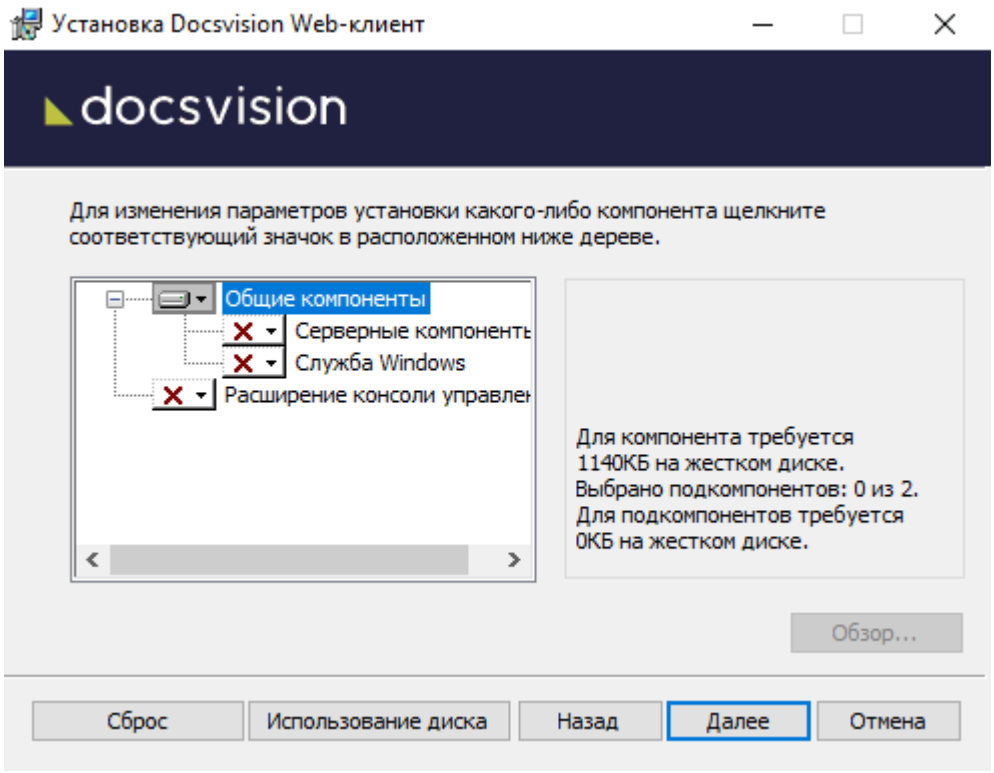


Рисунок 7. Выбор компонентов Web-клиента для установки

2. Перезапустите службы Docsvision Web-client и Docsvision Management Console Service.

   В ОС Windows службы Docsvision по умолчанию запускаются от системной учётной записи *NT AUTHORITY\Network Service*. Любое значение, установленное администратором, сбрасывается на значение по умолчанию после обновления версии системы.

Чтобы язык интерфейса (например, названий этапов согласования) отображался корректно, убедитесь, что в параметрах языка Windows для системных учётных записей установлена нужная локаль. Учётной записи *NT AUTHORITY\Network Service* должны быть предоставлены все необходимые права.

Установка Docsvision Web-клиент Toolkit

Docsvision Web-клиент Toolkit — пакет утилит для Web-клиента, включающий Конструктор Web-разметок и WebToolBuilder. Web-клиент Toolkit поставляется отдельно от инсталлятора Web-клиент и не требует дополнительной установки серверной и клиентской частей модуля.



Системные требования приведены в пункте Требования отдельной установки программы Конструктор Web-разметок.

Установка Docsvision Web-клиент Toolkit выполняется из пакета установки `Docsvision web-client toolkit.msi`.

Пользователь, выполняющий установку, должен обладать правами локального администратора.

Чтобы установить Docsvision Web-клиент Toolkit:

1. Запустите пакет установки `Docsvision web-client toolkit.msi`.

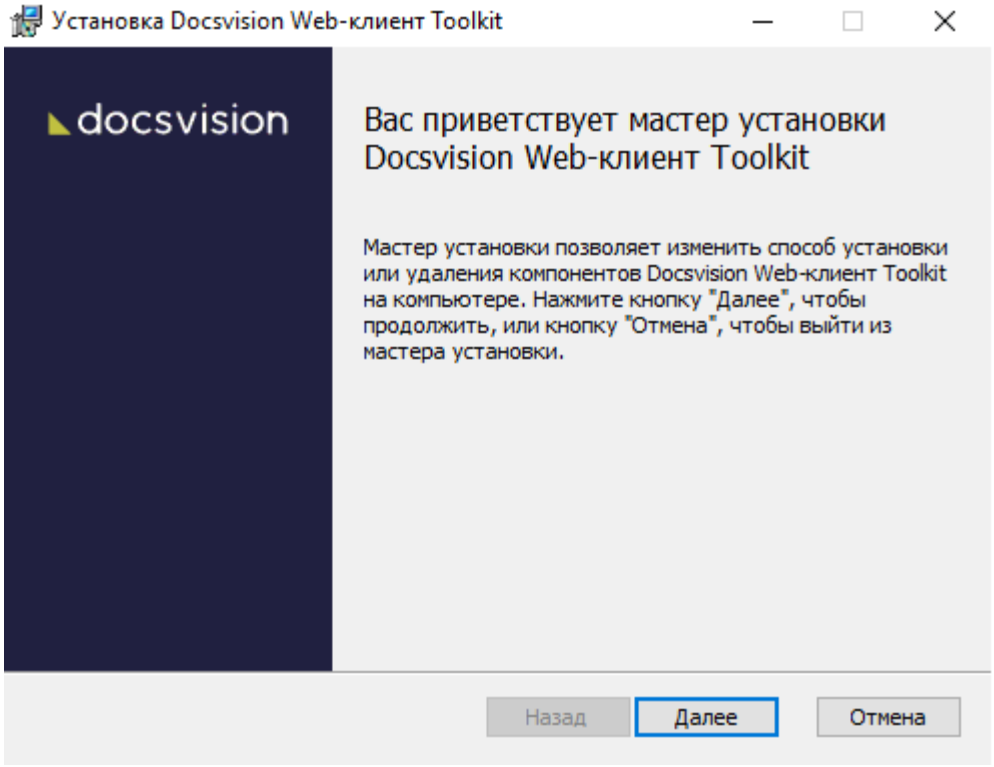


Рисунок 8. Мастер установки Docsvision Web-клиент Toolkit

2. Примите условия лицензионного соглашения, чтобы продолжить установку.

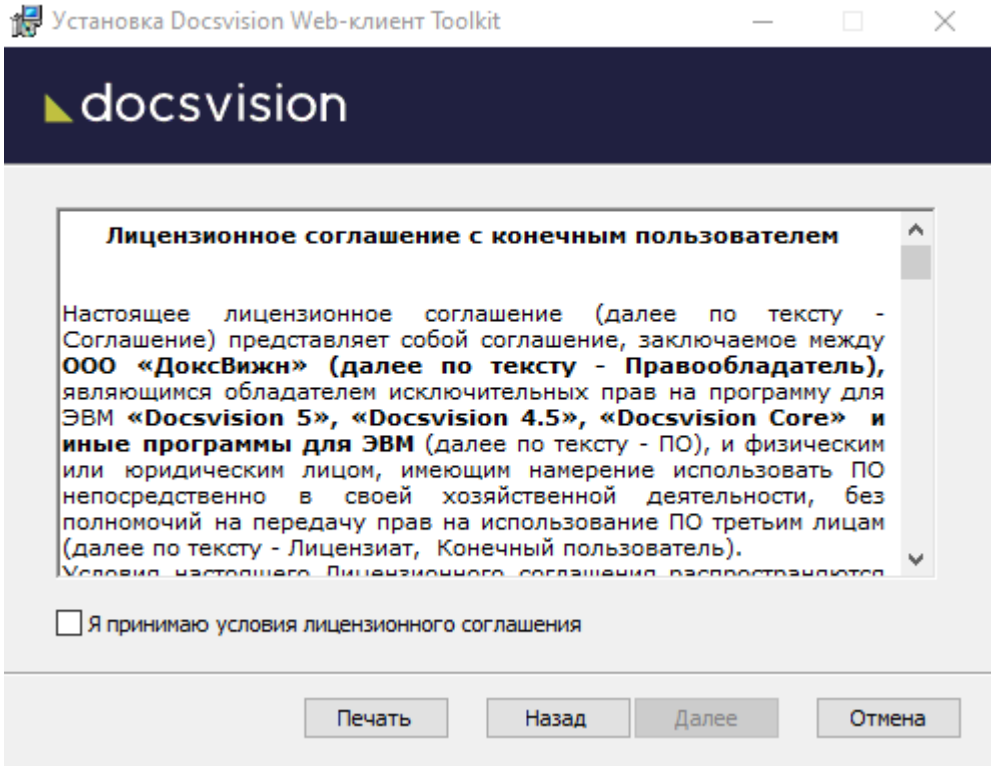


Рисунок 9. Условия лицензионного соглашения Docsvision Web-клиент Toolkit

3. Нажмите кнопку **Установить** и дождитесь завершения установки.

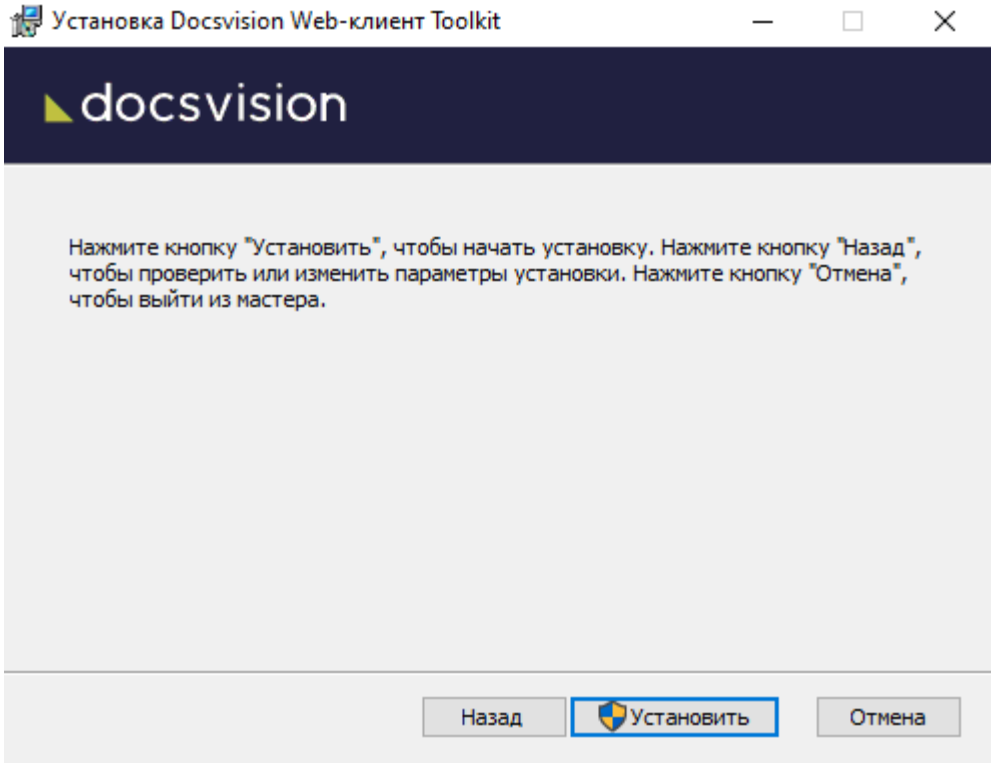


Рисунок 10. Установка Docsvision Web-клиент Toolkit

В меню Пуск, в группу "Docsvision Web-клиент" будут добавлены ярлыки Конструктор Web-разметок и WebToolBuilder.

4. Укажите адрес подключения к серверу Docsvision в файле конфигурации:

> ⚠ Обратите внимание, указанный ниже файл конфигурации расположен в ОС Windows по адресу `C:\Program Files (x86)\Docsvision\WebClientToolkit\WebLayoutsDesigner.appsettings.json`.

   - `WebLayoutsDesigner.appsettings.json` — конфигурация программы *Конструктор Web-разметок*.

```
{
  "Docsvision": {
    "WebClient": {
      "Server": {
        "SessionManagerConnection": {
          "ConnectionAddress": "http://dvserver.domain.com:5000", ①
          "BaseName": "alias" ②
        }
      }
    }
```

```
  }
}
```

① Адрес сервера Docsvision.

② Псевдоним БД, от имени которой будет запускаться программа Конструктор Web-разметок.

5. В Консоль управления Docsvision, на странице **Серверы** › **Имя сервера** › **Web-клиент** откройте группу настроек "Решения" и выполните импорт стандартных решений для просмотра стандартных разметок Web-клиента.
6. Нажмите кнопку **Применить** в правом верхнем углу окна Консоли, чтобы применить настройки Web-клиента и перезапустите службу Сервера Web-клиента.
7. Если планируется разрешить пользователям открывать файлы из карточек в офисных приложениях, установите и настройте DVWebTool или DVSupService по инструкции в разделе "Установка DVWebTool и DVSupService".

Первичная конфигурация Web-клиента



Модуль "Web-клиент" поддерживает следующие способы первоначальной конфигурации модуля:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше, см. ниже раздел Настройки в Консоли управления Docsvision
- При помощи настройки в конфигурационном файле модуля, см. ниже раздел Настройки в конфигурационном файле

Возможность указать настройки модуля в конфигурационном файле оставлена для обратной совместимости, первоначальная конфигурация должна выполняться в Консоли управления Docsvision, существующие настройки также рекомендуется перенести в Консоль управления Docsvision.

Конфигурация в Консоли управления Docsvision

1. Убедитесь, что в конфигурационном файле Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux или `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows указаны описанные ниже настройки. Файл может быть изменён в любом текстовом редакторе (потребуются права суперпользователя/администратора). Настройки в

конфигурационном файле:

Фрагмент конфигурационного файла Web-клиента:

```
{
  "Docsvision": {
    "ServiceId": "00000000-0000-0000-0000-000000000000", ①
    "SettingsService": {
      "ConnectionString": "http://localhost:5200/api", ②
      "ApiKey": "anything" ③
    }
  }
}
```

① `ServiceId` — уникальный идентификатор данного экземпляра модуля, генерируется и записывается автоматически.

② `ConnectionString` — полный адрес Сервиса настроек.

③ `ApiKey` — API-ключ для подключения к Сервису настроек, можно оставить значение по умолчанию или изменить, главное условие, чтобы значения в конфигурации модулей Сервис настроек и Web-клиент были одинаковые.

2. Дальнейшие настройки модуля выполняются в Консоли управления Docsvision. Откройте Консоль управления Docsvision на странице **Серверы › Имя сервера › Web-клиент › "Группа настроек "Подключения"** и настройте следующие параметры:
   - Адрес сервера приложений
   - Имя пользователя — имя системной учётной записи Docsvision
   - Пароль — пароль системной учётной записи

     Пароль может быть зашифрован, штамп сертификата шифрования паролей указывается в разделе "Общие настройки сервера приложений".

     Отпечаток сертификата указывается без двоеточия, например, `D8602179888DC8402B393F11DCA16A3376DDF879`, см. подробнее в документации по установке системы, раздел "Защита системной учётной записи". Отпечаток можно не указывать, если шифровать пароль системной учётной записи не планируется.

3. Откройте Консоль управления Docsvision на странице **Серверы › Имя сервера › Web-клиент › "Группа настроек "Подключение"**.

4. Перейдите к настройкам "Настройка доменных каталогов" и настройте следующие параметры:
   - Тип аутентификации в LDAP
   - Полное имя домена
   - Логин — имя учётной записи, под которой процесс обращается к ActiveDirectory
   - Пароль — пароль учётной записи пользователя для аутентификации LDAP
   - Домен по умолчанию — полное имя домена для входа по умолчанию в формате FQDN
5. При необходимости настройте использование расширенной аутентификации на странице **Серверы** › **Имя сервера** › **Сервер приложений** › **"Группа настроек "Расширенная аутентификация"**.
6. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

   Служба запускается от имени УЗ Docsvision и не требует административных привилегий.

(Старый метод) Конфигурация без Консоли управления Docsvision

1. Убедитесь, что в конфигурационном файле Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux или `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows указаны описанные ниже настройки. Файл может быть изменён в любом текстовом редакторе (потребуются права суперпользователя/администратора). Настройки в конфигурационном файле:

```
{
  "Docsvision": {
    "WebClient": {
      "Authentication": {
        "DefaultDomain": "DOMAIN.COM" ①
      },
      "Server": {
        "SessionManagerConnection": {
          "ConnectionAddress": "http://dvserver.domain.com:5000" ②
        }
      }
    },
    "ServiceId": "3ed2fd20-eaf9-8846-7089-b34dd8192328", ③
    "SettingsService": {
```

```
      "ConnectionString": "http://storage-server.com:5200/api", ④
      "ApiKey": "dvwebclientapikey" ⑤
    }
  },
  "Ldap": { ⑥
    "AuthType": "Basic", ⑦
    "Credential": {
      "UserName": "account@domain.com", ⑧
      "Password": "password" ⑨
    }
  },
  "DataProtectCertificateThumbprint": "thumbprint", ⑩
  "SystemUserAccount": "account@domain.com", ⑪
  "SystemUserPassword": "password", ⑫
}
```

① `DefaultDomain` — домен по умолчанию в формате FQDN.

② `ConnectionAddress` — адрес сервера Docsvision.

③ `ServiceId` — уникальный идентификатор Сервера приложений, с базами которого будет работать серверная консоль.

④ `ConnectionString` — адрес Сервиса настроек.

⑤ `ApiKey` — API-ключ Сервиса настроек.

⑥ `Ldap` — имя и пароль учётной записи, под которой процесс обращается к ActiveDirectory.
Пароль можно указывать в открытом или в зашифрованном виде, см. подробнее о шифровании в документации по установке системы, раздел "Защита системной учётной записи"

⑦ `AuthType` — тип аутентификации: `Basic` — аутентификация через LDAP (подробнее см. в документации модуля Платформа Настройка работы с несколькими доменами), `Oauth` — аутентификация через ЕСИА или Azure.

⑧ `UserName` — логин пользователя модуля для аутентификации LDAP.

⑨ `Password` — пароль учётной записи пользователя для аутентификации LDAP.

⑩ `DataProtectCertificateThumbprint` — отпечаток закрытого ключа шифрования SHA1.
Указывается без двоеточия, например, `D8602179888DC8402B393F11DCA16A3376DDF879`, см. подробнее в документации по установке системы, раздел "Защита системной учётной записи".
Если шифровать пароль системной учётной записи не планируется,

параметр можно удалить.

⑪ `SystemUserAccount` — имя системной учётной записи Docsvision в формате UPN.

⑫ `SystemUserPassword` — пароль системной учётной записи рекомендуется хранить в зашифрованном виде. Подробнее про шифрование см. в документации по установке системы, раздел "Защита системной учётной записи".

2. Запустите службу Web-клиента. В ОС Linux, в 6.1 сборки 1110 и выше, служба запускается от имени УЗ Docsvision и не требует root-привилегий.

   В ОС Windows службы Docsvision по умолчанию запускаются от системной учётной записи *NT AUTHORITY\Network Service*. Любое значение, установленное администратором, сбрасывается на значение по умолчанию после обновления версии системы.

   Чтобы язык интерфейса (например, названий этапов согласования) отображался корректно, убедитесь, что в параметрах языка Windows для системных учётных записей установлена нужная локаль. Учётной записи *NT AUTHORITY\Network Service* должны быть предоставлены все необходимые права.

3. Выполните настройку Web-клиента, см. "(Старый метод) Настройка через Панель управления Web-клиентом".

Удаление модуля

- Чтобы удалить модуль в ОС Windows, воспользуйтесь стандартными средствами системы (Установка и удаление программ).
- Чтобы удалить модуль в ОС Linux, выполните команду:

Astra Linux

```
$ sudo apt-get purge docsvision-webclient
```

РЕД ОС

```
$ sudo dnf remove docsvision-webclient
```

Альт Сервер

```
$ sudo apt-get remove docsvision-webclient
```

- Чтобы удалить раширение для Консоли управления Docsvision в ОС Linux, выполните:

Astra Linux

```
$ sudo apt-get purge docsvision-ext-webc-mc
```

РЕД ОС

```
$ sudo dnf remove docsvision-ext-webc-mc
```

Альт Сервер

```
$ sudo apt-get remove docsvision-ext-webc-mc
```

(Старый метод) Настройка через Панель управления Web-клиентом

Модуль "Web-клиент" поддерживает следующие варианты первоначальной настройки Web-клиента:

- Из интерфейса Консоли управления Docsvision, без взаимодействия с программой Панель управления Web-клиентом, начиная с версии 6.1 сборки 1265 и выше: "6.2@mgmtconsole:servers:webclient-connection.pdf"
- При помощи настройки в программе Панель управления Web-клиентом, в версиях ниже указанной, см. ниже

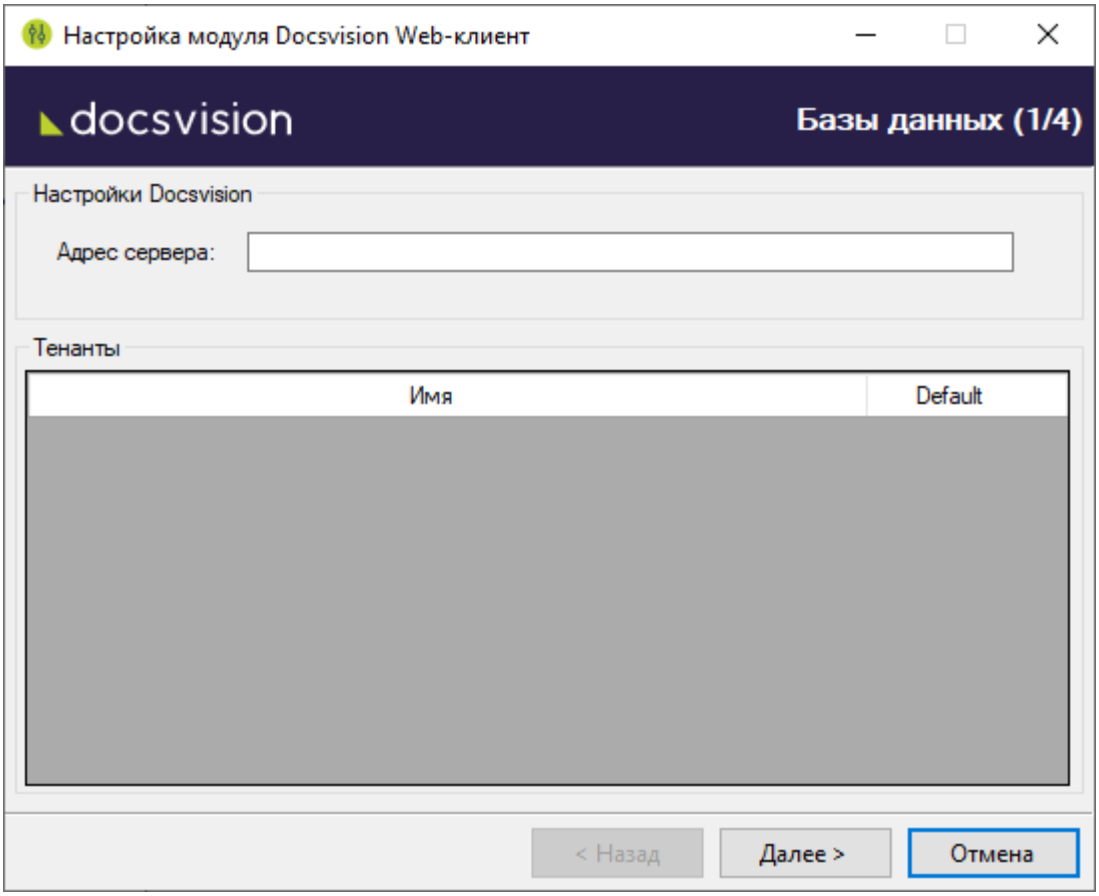


Рисунок 11. Расширенная настройка Web-клиента

Запускает мастер расширенной настройки. Данные, необходимые для настройки Web-клиента, будут последовательно запрошены мастером настройки.

Чтобы приступить к настройке:

1. Откройте стартовую страницу Web-клиента и нажмите кнопку **Настроить**.
2. Будет запущена расширенная настройка.

   Простая настройка не поддерживается в данной версии модуля.

   Если в системе не установлены какие-либо обязательные компоненты, будет показано название требуемого компонента и возможность продолжения настройки будет заблокирована. Для продолжения настройки нужно установить требуемые компоненты и повторно запустить настройку БД.

3. Следуйте соответствующим инструкциям.

Расширенная настройка

В расширенном варианте настройки данные для подключения Web-клиент к Docsvision должен будет указать администратор.

Модуль "Web-клиент" поддерживает следующие варианты расширенной настройки Web-клиента:

- Из интерфейса Консоли управления Docsvision, без взаимодействия с программой Панель управления Web-клиентом, начиная с версии 6.1 сборки 1265 и выше: "6.2@mgmtconsole:servers:webclient-connection.pdf"
- При помощи настройки в программе Панель управления Web-клиентом, в версиях ниже указанной, см. ниже

Нажмите на кнопку **Настроить** на стартовой странице Панели управления Web-клиентом, чтобы перейти к расширенной настройке Web-клиента.

Расширенная настройка включает следующие шаги:

1. На странице "Базы данных" настраиваются параметры подключения Web-клиента к платформе Docsvision.

Адрес сервера Адрес подключения к серверу Docsvision.

Тенанты БД Docsvision, которые подключены к Web-клиенту.

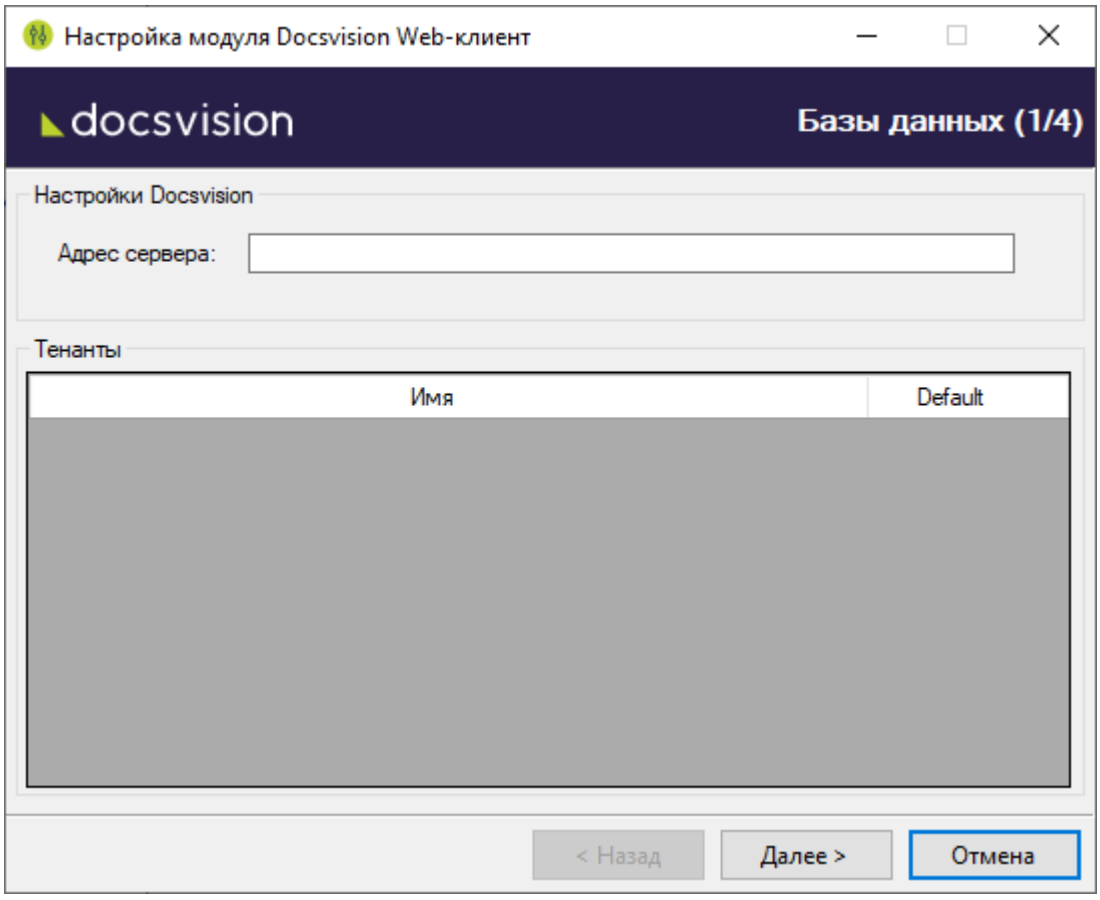


Рисунок 12. Страница настроек "Базы данных"

Тенанты:

- Если нужная БД в списке отсутствует, её можно добавить через *Консоль настройки Docsvision* при помощи Мастера баз данных.

  Пользователи Web-клиент могут работать с несколькими БД. Текущей активной считается БД, псевдоним которой указан в адресной строке до первой точки, например: `http://DocsvisionDB.company:5004`.

2. На странице *Языки* отметьте языки, которые пользователи смогут выбрать языком интерфейса. Также можно изменить язык интерфейса, который будет использоваться по умолчанию.

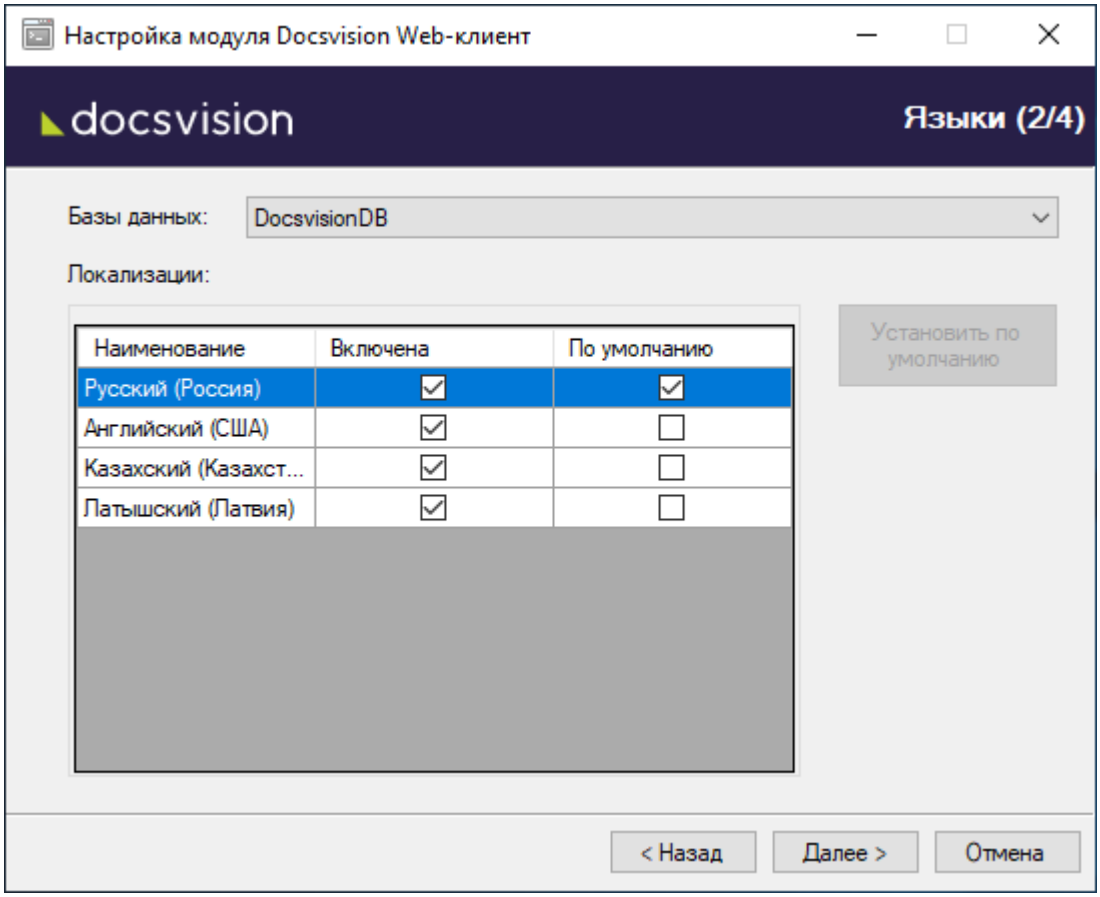


Рисунок 13. Страница настроек "Языки"

3. На странице *Решения* выполняется загрузка настроек стандартных карточек в БД Docsvision.

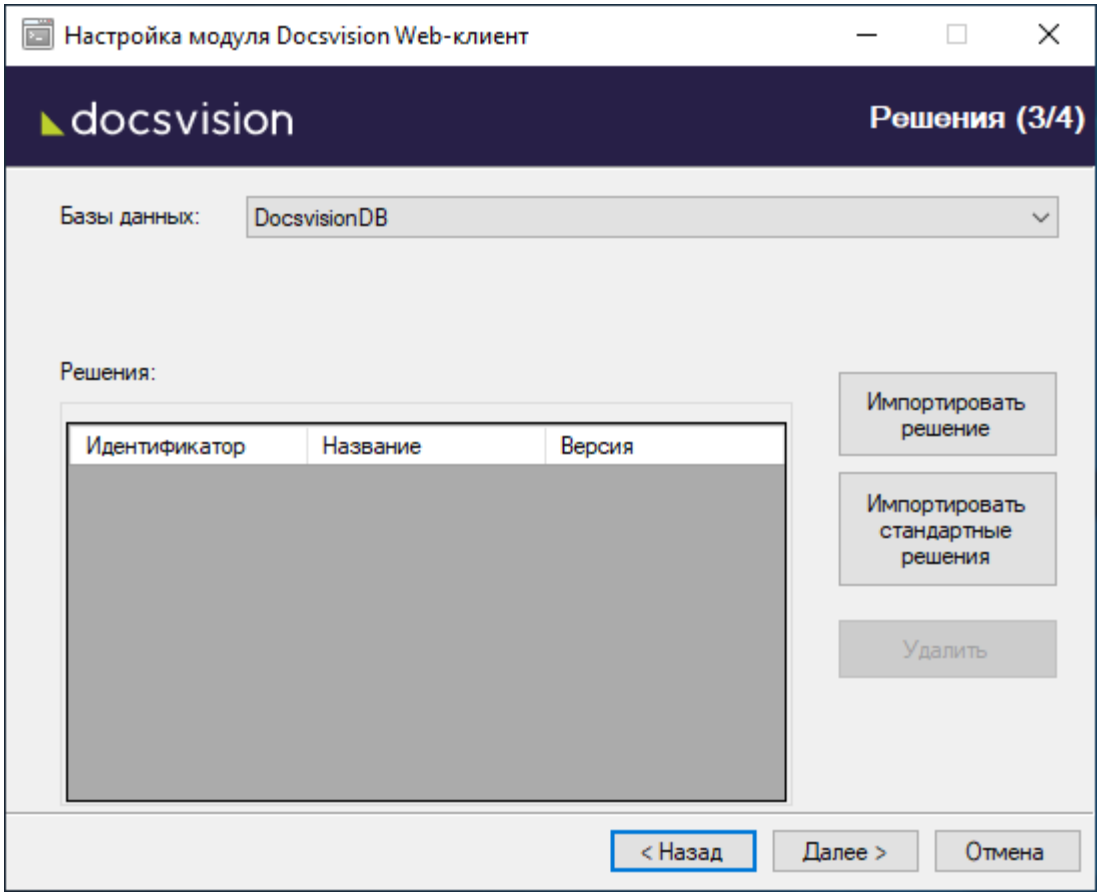


Рисунок 14. Страница настроек "Решения"

Если БД не содержит Решений, администратор может загрузить стандартные Решения или пропустить загрузку.

- Для загрузки стандартных Решений нажмите кнопку **Импортировать стандартные решения**.
- Если загрузка решений пропущена, нужно будет настроить разметки самостоятельно, иначе открытие карточек будет невозможно.

  Решения загружаются только в выбранную БД. Если к Web-клиенту подключено несколько БД, то *решения* нужно поочередно загрузить во все требуемые базы данных. Это же относится к обновлению Решений.

  Нажмите кнопку **Далее**, чтобы продолжить настройку.

4. На странице *Виды карточек по умолчанию* настраиваются виды карточек. При создании новой карточки указанного типа, по умолчанию будет выбран соответствующий вид.

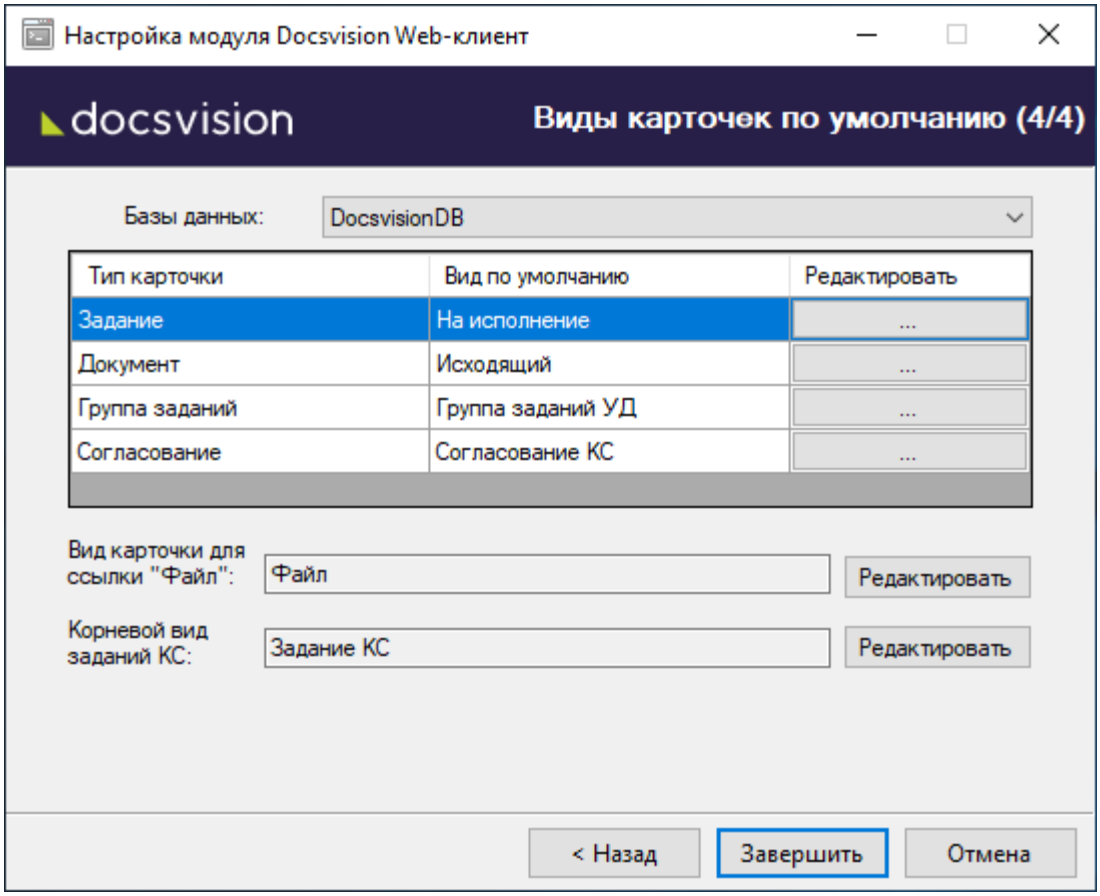


Рисунок 15. Страница настроек "Виды карточек по умолчанию"

Изменение стандартного вида для соответствующего типа выполняется при нажатии кнопки **Редактировать**, а для списка — кнопки **...**.

Используйте Панель управления Web-клиентом, чтобы изменить настройки Web-клиента. Работа с Панелью управления рассмотрена в разделе Функции администратора.

Настройка Web-клиента в сети без Active Directory

Web-клиент поддерживает подключение к серверу Docsvision под системной учётной записью из каталога ALD Pro, указанной в параметрах `SystemUserAccount` и `SystemUserPassword` конфигурационного файла.



Модуль "Web-клиент" поддерживает следующие способы настройки коннектора к каталогу ALD Pro:

- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже
- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1

> сборки 1265 и выше: "Авторизация в Консоли управления по протоколу LDAPS без Active Directory"

Условия работы коннектора:

- Для работы коннектора авторизации ALD Pro не требуется дополнительных опций лицензии.
- Авторизация с помощью ALD Pro поддерживается в Docsvision при установке модуля Web-клиент 6.1 сборки 1110 и выше, 6.1 сборки 546 и выше и других модулей, входящих в обновление.
- Если база данных была создана, но не использовалась, перед использованием коннектора необходимо завести как минимум одного сотрудника в справочнике сотрудников и авторизоваться под ним в Web-клиенте.
- При использовании конфигурации с несколькими контроллерами, объединенными через балансировщик нагрузки, необходимо использовать сертификат с указанием домена в формате `*.domain.com`. Альтернативный вариант — в конфигурационном файле сервиса указать адрес балансировщика в параметре `LdapServerAddresses`.

В конфигурационном файле Web-клиента укажите следующие параметры:

```
{
  "SystemUserAccount": "account@domain.com", ①
  "SystemUserPassword": "password", ②
  "Docsvision": {
    "WebClient": {
      "Server": {
        "SessionManagerConnection": {
          "ConnectionAddress": "http://dvserver.domain.com:5000", ③
          "BaseName": "alias" ④
        }
      }
    }
  }
}
```

① Учётная запись из каталога ALD Pro для подключения к серверу Docsvision.

② Пароль УЗ из каталога ALD Pro для подключения к серверу Docsvision.

③ Адрес сервера Docsvision.

④ Псевдоним БД, от имени которой будет запускаться программа Панель управления Web-клиентом.

Для работы укажите УЗ из каталога ALD Pro на машине с Панель управления Web-клиентом:

```
setx DV_SystemUserAccount account@domain.com
setx DV_SystemUserPassword password
```

Функциональность доступна в Web-клиент с версии 6.1 сборки 1110 и выше при условии установки версий модулей, входящих в накопительное обновление 25.3.

Обновление Web-клиента

Web-клиент

Обновление Web-клиента

Информация об обновлении модуля и необходимых подготовительных этапах.

- Обновление Web-клиента
- Актуализация клиентского кода

Обновление Web-клиента

Список изменений в текущей версии см. здесь: Изменения в релизной версии.

Условия обновления

- При обновлении разных серверов компоненты модуля должны быть одной версии.
- Версии **всех** серверных и клиентских компонентов должны быть одинаковыми.
- Обновление должно выполняться в нерабочее время.
- Перед обновлением создайте резервную копию БД Docsvision.

- Если в БД PostgreSQL используется pgAgent, остановите планировщик до начала обновления.
- Обновите версию модуля на всех узлах кластера Docsvision.

При обновлении Web-клиента происходит проверка параметра `<LoadLayouts></LoadLayouts>` в конфигурационном файле панели управления Web-клиента (`WebClientAdminConsole.config`), расположенного по адресу `%LocalAppData%\Docsvision\WebClientAdminConsole.config`.

Если файл отсутствует или значение `<LoadLayouts>true</LoadLayouts>`, Web-клиент будет обновлён в стандартном режиме с импортом решений в базу данных.

Если значение `<LoadLayouts>false</LoadLayouts>`, при обновлении Web-клиента решения не будут автоматически загружено в базу данных, администратор должен будет сделать это самостоятельно позднее.

Обновление модуля на сервере под управлением Linux

Если к новой версии модуля прилагается отдельная инструкция по обновлению, следуйте ей.

Чтобы установить новую версию модуля:

1. Создайте резервную копию БД.
2. Если в БД PostgreSQL используется pgAgent, остановите планировщик.
3. Обновите серверные компоненты на сервере Linux командой:

Astra Linux

```
sudo apt-get update
sudo apt-get install --only-upgrade 'docsvision-webclient*'
```

РЕД ОС

```
sudo dnf makecache
sudo dnf update 'docsvision-webclient*'
```

Альт Сервер

```
sudo apt-get update
sudo apt-get install --only-upgrade=true 'docsvision-webclient*'
```

4. При обновлении сервера Docsvision предыдущих версий под управлением Astra Linux будет предложено перезаписать текущий конфигурационный файл `appsettings.json`.

   ▼ *Доступные варианты:*

   - Сохранить текущий `appsettings.json`.
     - Все выполненные настройки останутся без изменений, стандартный файл конфигурации будет сохранён как `appsettings.json.dpkg-dist`.
   - Показать различия между версиями.
     - В окне командной строки будут отображены отличия между старой и новой версией. Знаком `+` обозначаются добавленные строки, знаком `-` — удалённые.
   - Перезаписать текущий конфигурационный файл.
     - Текущий файл будет перезаписан стандартным, все выполненные настройки будут сохранены в файле `appsettings.json.dpkg-old`, их необходимо будет перенести в Консоль управления Docsvision.
   - Запустить оболочку командной строки для проверки ситуации.
     - Возвращает в окно командной строки, дальнейшие действия зависят от администратора.

5. Запустите *Консоль управления Docsvision* и перейдите в раздел **Серверы › Имя-сервера › Сервер приложений › Базы данных**.

   > Пользователь, от имени которого запускается *Консоль управления Docsvision* должен являться администратором Консоли управления Docsvision — быть добавленным в группу *Docsvision Management Console Administrators* в конфигурационном файле Консоли управления Docsvision, см. раздел "Первичная настройка Консоли управления Docsvision".

6. Обновите существующую БД по инструкции, приведённой в документации модуля *Консоль управления Docsvision*, раздел "Обновление БД".
7. Дойдите до выбора обновляемых библиотек карточек. Выберите библиотеки и нажмите **Далее**.

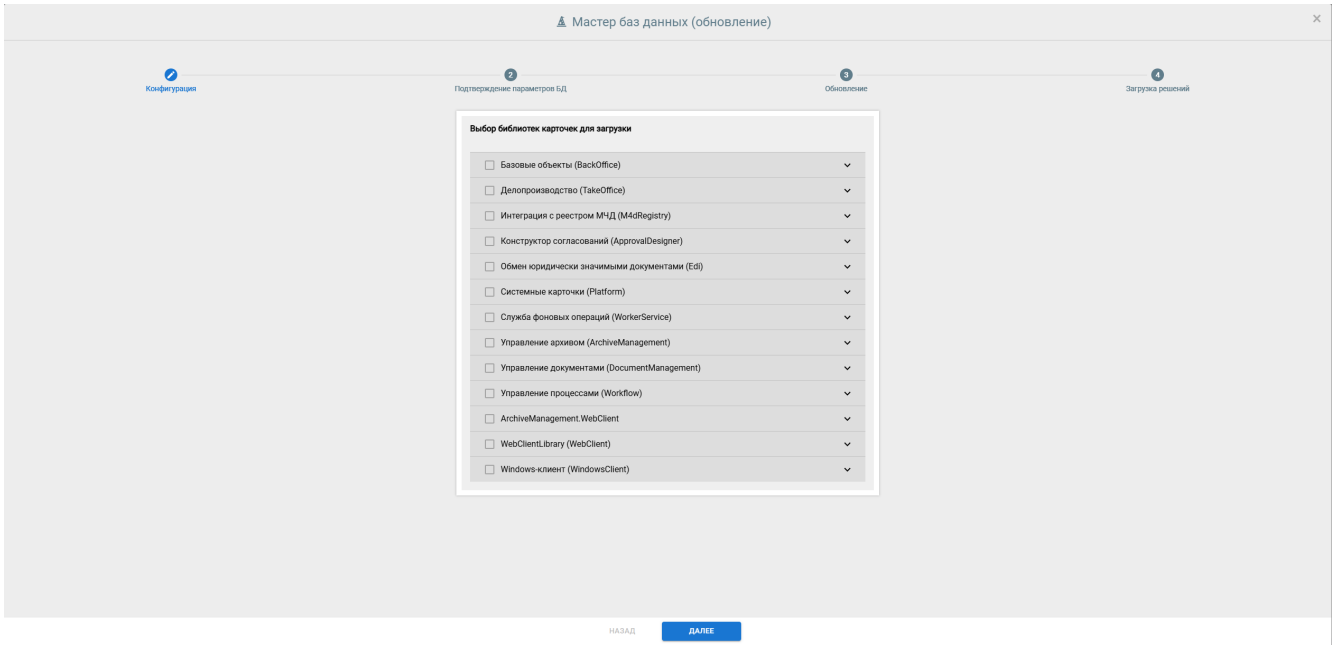


Рисунок 16. Обновление библиотек карточек

8. Перезапустите службу Сервера Docsvision на Linux.
9. Выберите модули Docsvision, настройки которых должны быть загружены в БД и нажмите **Загрузить**.

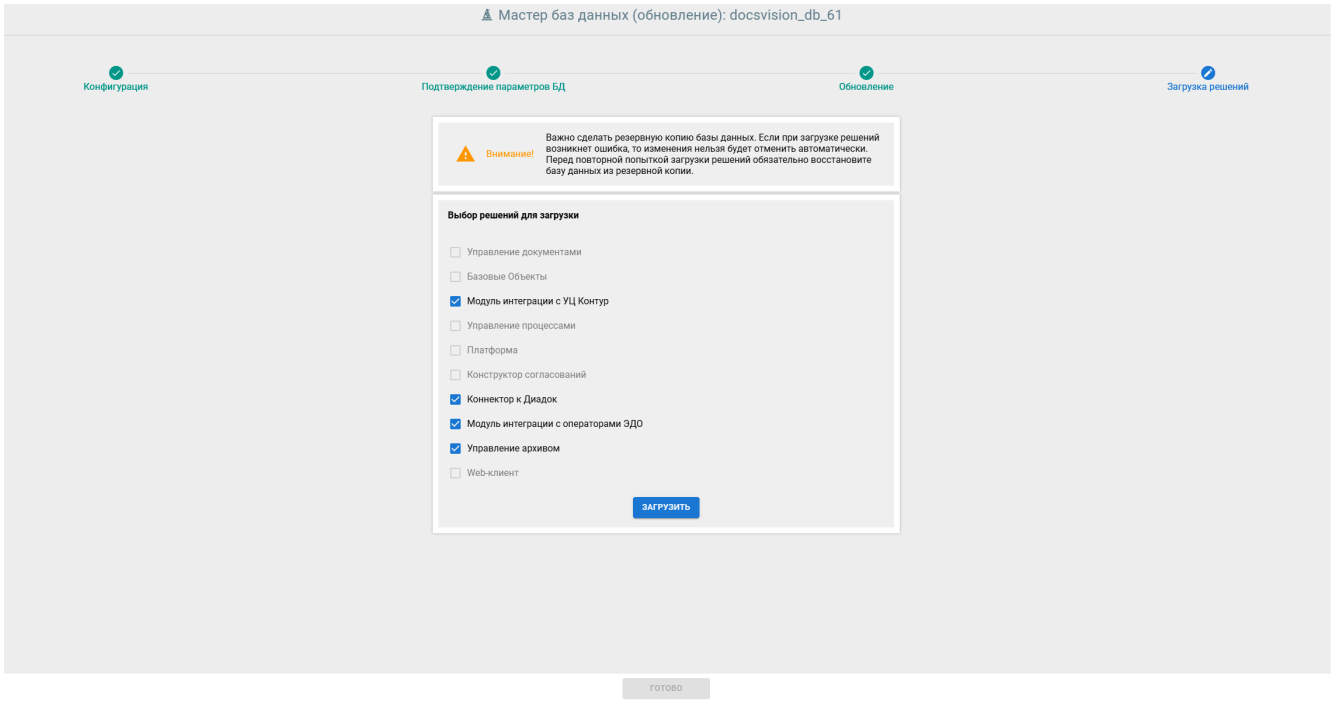


Рисунок 17. Шаг загрузки настроек модулей

Обновление модуля на сервере под управлением Windows

Если к новой версии модуля прилагается отдельная инструкция по обновлению, следуйте ей.

Чтобы установить новую версию модуля:

1. Создайте резервную копию БД.
2. Если в БД PostgreSQL используется pgAgent, остановите планировщик.
3. Обновите необходимые компоненты модуля на сервере Windows из инсталлятора `Docsvision web-client.msi`.
4. На сервере Docsvision под управлением Windows при обновлении сервисов текущий конфигурационный файл `appsettings.json` не перезаписывается. При обновлении системы с предыдущих версий можно сравнить существующий конфигурационный файл со стандартной версией из комплекта поставки — `appsettings.json.dist`, расположенный в том же каталоге.
5. Запустите *Консоль управления Docsvision* и перейдите в раздел **Серверы › Имя-сервера › Сервер приложений › Базы данных**.

   Пользователь, от имени которого запускается *Консоль управления Docsvision* должен являться администратором Консоли управления Docsvision — быть добавленным в группу **Docsvision Management Console Administrators** в конфигурационном файле Консоли управления Docsvision, см. раздел "Первичная настройка Консоли управления Docsvision".

6. Обновите существующую БД по инструкции, приведённой в документации модуля *Консоль управления Docsvision*, раздел "Обновление БД".
7. Дойдите до выбора обновляемых библиотек карточек. Выберите библиотеки и нажмите **Далее**.

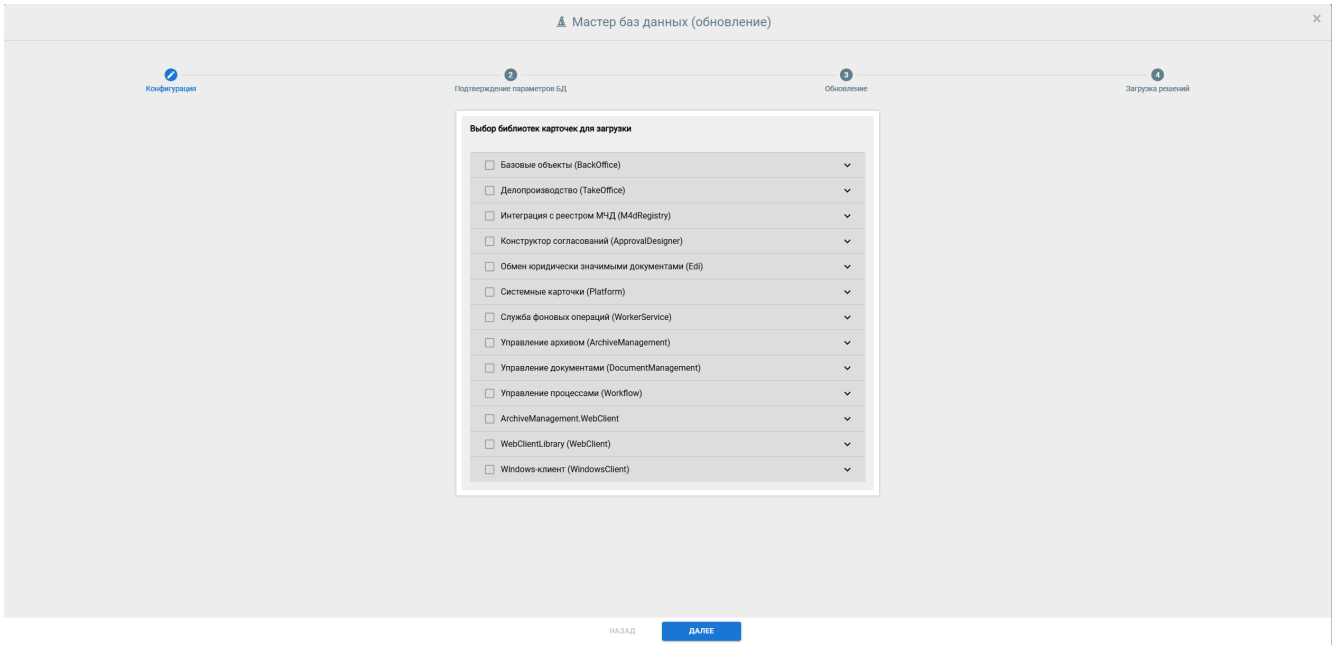


Рисунок 18. Обновление библиотек карточек

8. Перезапустите службу Сервера Docsvision на Windows.
9. Выберите модули Docsvision, настройки которых должны быть загружены в БД и нажмите **Загрузить**.

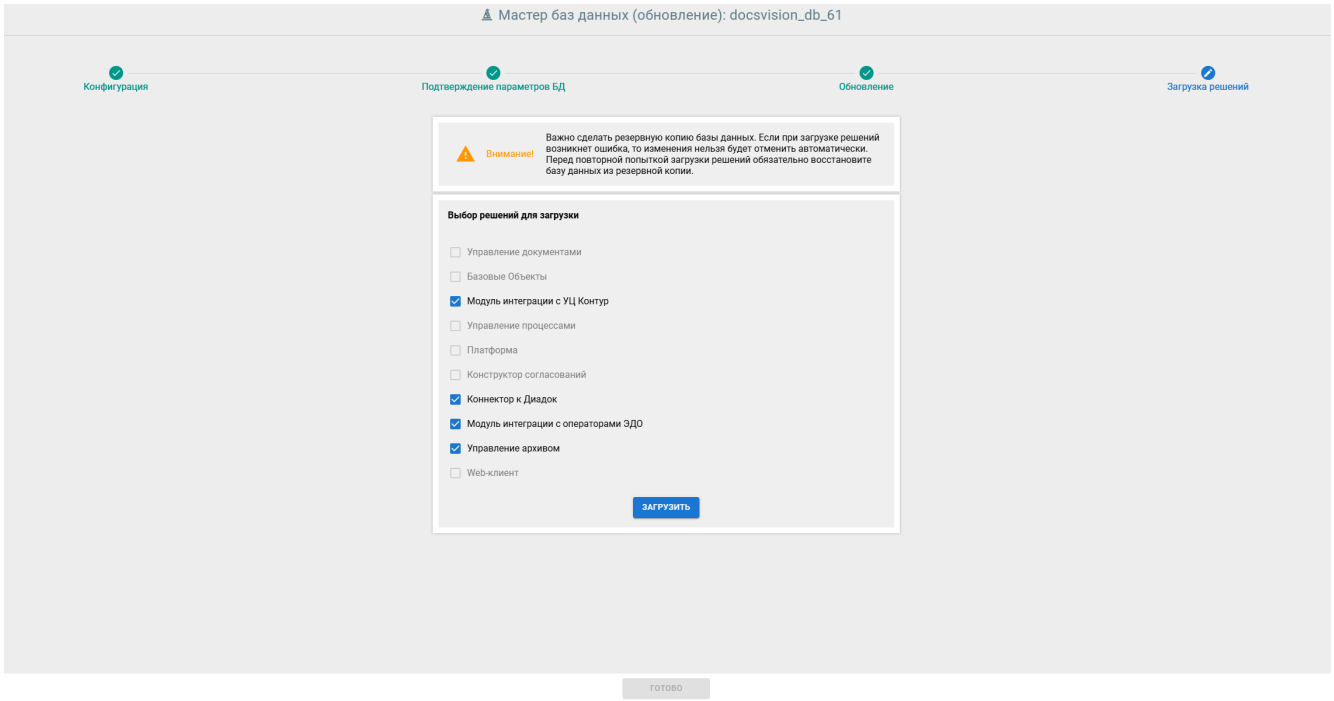


Рисунок 19. Шаг загрузки настроек модулей

10. Обновите версию DVWebTool/DVSupService на клиентских рабочих местах.

Обновление расширения Web-клиента для Консоли управления Docsvision на сервере под управлением Linux

Чтобы установить новую версию:

1. Обновите компоненты расширения Web-клиента для Консоли управления Docsvision на сервере Linux командой:

Astra Linux

```
sudo apt-get update
sudo apt-get install --only-upgrade 'docsvision-ext-webc-mc*'
```

РЕД ОС

```
sudo dnf makecache
sudo dnf update 'docsvision-ext-webc-mc*'
```

Альт Сервер

```
sudo apt-get update
sudo apt-get install --only-upgrade=true 'docsvision-ext-webc-mc*'
```

Обновление расширения Web-клиента для Консоли управления Docsvision на сервере под управлением Windows

1. На сервере Консоли управления Docsvision запустите инсталлятор `Docsvision web-client.msi`.
2. На шаге выбора компонентов выберите компонент "Расширение консоли управления" и нажмите **Далее**.
3. Нажмите на кнопку **Установить** и дождитесь завершения установки модуля.
4. Нажмите на кнопку **Готово**, чтобы закрыть мастер установки.

Актуализация клиентского кода

Применение собственных стилей к основной разметке Web-клиента

В связи с изменением приоритетов некоторых стилей основной разметки:[0] собственные стили, изменяющие основную разметку, могут перестать применяться. Повысьте приоритет селекторов в стилях, чтобы переопределить стандартные.

Устаревшие элементы API

Убедитесь, что вы не используете устаревшие функции API. Если в решении использовались устаревшие функции, измените реализацию, используя альтернативные возможности. Со списком изменений клиентского API можно ознакомиться в разделе Изменения в API.

Панель управления Web-клиентом



Модуль "Web-клиент" поддерживает следующие настройки Web-клиента:

- Из интерфейса Консоли управления Docsvision, без взаимодействия с программой Панель управления Web-клиентом, начиная с версии 6.1 сборки 1265 и выше: "6.2@mgmtconsole:servers:webclient-connection.pdf"
- При помощи настройки в программе Панель управления Web-клиентом, в версиях ниже указанной, см. ниже

Администратору представляется программа *Панель управления Web-клиентом*, с помощью которой можно:

- Настраивать языки локализации.
- Загружать *решения* программы *Конструктор web-разметок*.
- Настраивать виды карточек по умолчанию, которые будут использоваться при создании карточек.

Открыть Панель управления Web-клиентом

Чтобы запустить Панель управления Web-клиентом, откройте программу **Пуск › Все программы › Docsvision › Панель управления Web-клиентом** от имени администратора Web-клиента.

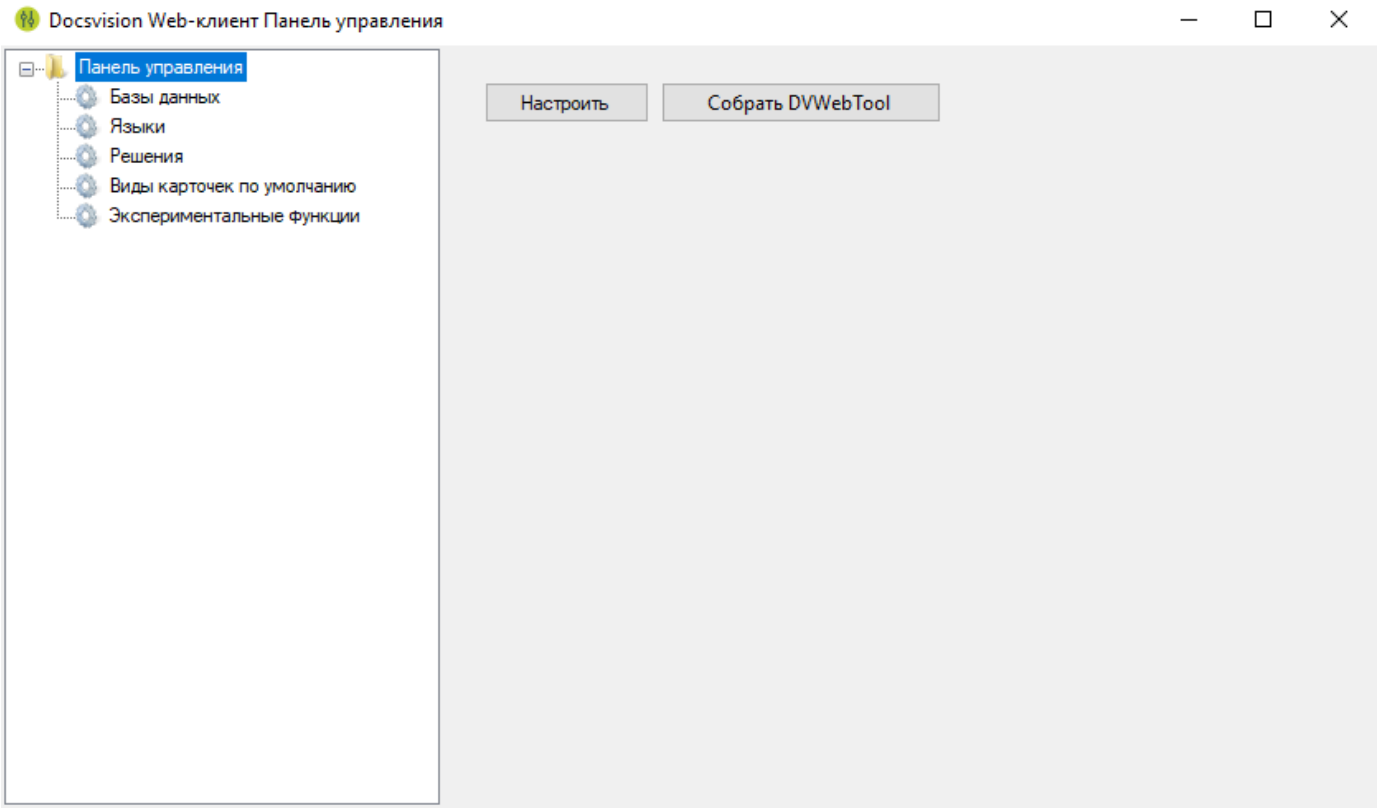


Рисунок 20. Главное окно программы Панель управления Web-клиентом

В окне панели управления Web-клиентом отображаются: дерево настроек — слева, и область отображения настроек — справа.

Дерево настроек предназначено для быстрого перехода к требуемому разделу настроек.

В области отображения настроек по умолчанию расположены кнопки:

- **Настроить** — запускает расширенную настройку. Подробности в разделе Первоначальная настройка Web-клиента.
- **Собрать DVWebTool** — позволяет повторно собрать инсталлятор DVWebTool. Подробнее см. "Настройки запуска инструментов".
  Программа недоступна в ОС linux.

Раздел "Базы данных"

Настройки подключения к серверу

Раздел *Базы данных* открывает страницу настроек подключения Web-клиент к серверу Docsvision.



Модуль "Web-клиент" поддерживает следующие варианты указания адреса подключения к серверу Docsvision:

- Из интерфейса Консоли управления Docsvision, без взаимодействия с программой Панель управления Web-клиентом, начиная с версии 6.1 сборки 1265 и выше: "6.2@mgmtconsole:servers:webclient-connection.pdf"
- При помощи настройки в программе Панель управления Web-клиентом, в версиях ниже указанной, см. ниже

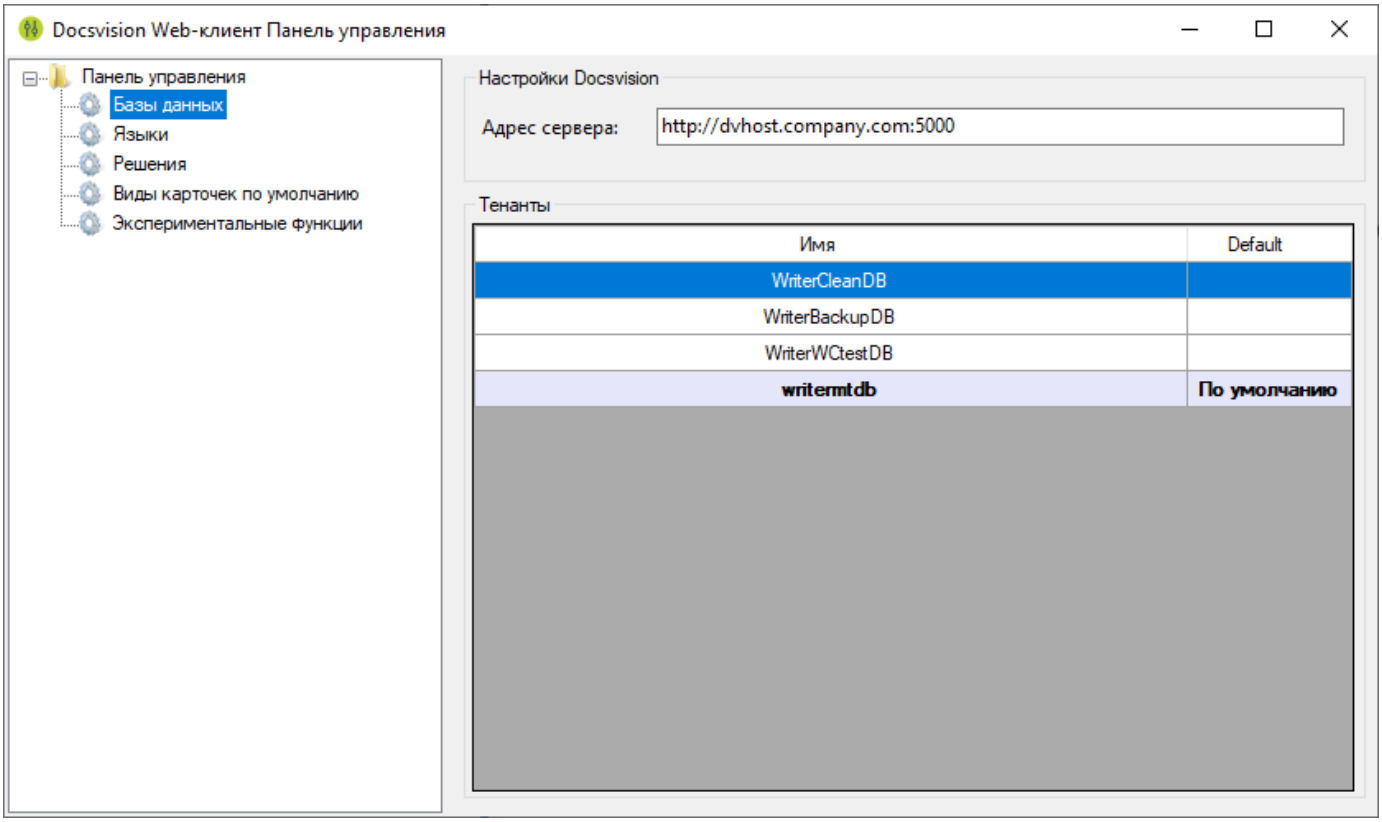


Рисунок 21. Страница настроек "Базы данных"

Таблица 2. Страница настроек "Базы данных"

Адрес сервера	Полный или сокращенный адрес сервера платформы Docsvision, к которому подключается Web-клиент.
Тенанты	Список подключенных к Web-клиент баз данных указанного сервера Docsvision.

Если нужная БД в списке отсутствует, её можно добавить через *Консоль настройки Docsvision* при помощи Мастера баз данных.

После добавления новой БД необходимо её настроить. Потребуется проверить локализации, импортировать решения и настроить виды карточек по умолчанию. Подробнее см. п. Расширенная настройка.

Изменение адреса подключения к серверу

При необходимости адрес сервера Docsvision может быть изменён.

В поле `Адрес сервера` указывается адрес подключения к серверу Docsvision. Пользователи Модуля могут работать с несколькими БД. Текущей активной считается БД, псевдоним которой указан в адресной строке до первой точки, например: `http://DocsvisionDB.company.com:5004`.

Раздел "Языки"

В разделе *Языки* можно выбрать язык по умолчанию и языки локализации, доступные в Web-клиенте и программе Конструктор Web-разметок.



Модуль "Web-клиент" поддерживает следующие варианты языковых настроек:

- Из интерфейса Консоли управления Docsvision, без взаимодействия с программой Панель управления Web-клиентом, начиная с версии 6.1 сборки 1265 и выше: "6.2@mgmtconsole:servers:webclient-locales.pdf"
- При помощи настройки в программе Панель управления Web-клиентом, в версиях ниже указанной, см. ниже

Языковые настройки

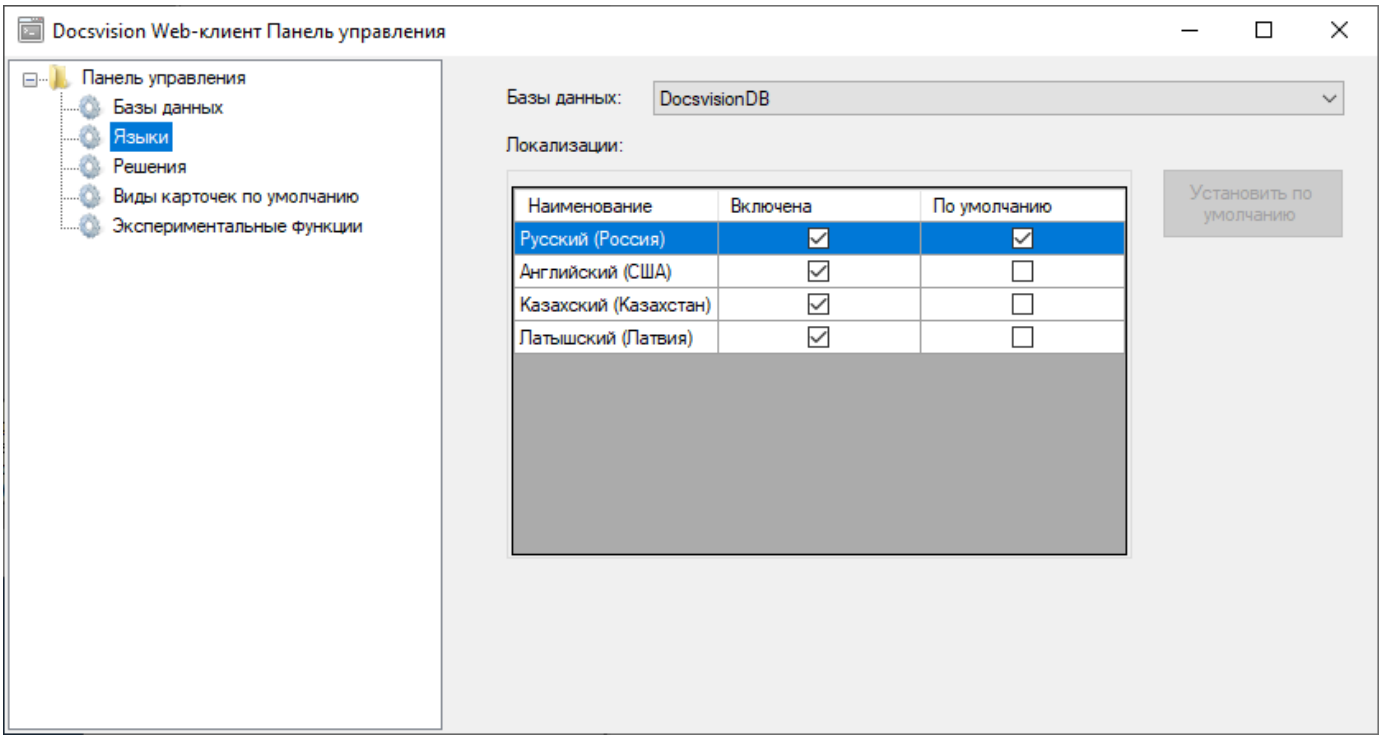


Рисунок 22. Страница настроек "Языки"

Таблица 3. Страница настроек "Языки"

Базы данных	Выбирает БД Docsvision, для которой настраиваются языки. В данном списке отображаются БД, которые были добавлены через *Консоль настройки Docsvision*.
Локализации	Список доступных языков.
Установить по умолчанию	Назначает выбранный язык языком по умолчанию.

Локализации с флагом **Включена** доступны пользователям для выбора в качестве языка пользовательского интерфейса. Чтобы отключить или включить язык локализации, снимите или установите флаг в столбце `Включена` в строке требуемого языка.

Чтобы выбрать локализацию по умолчанию, выберите язык в списке и нажмите кнопку **Установить по умолчанию**. В столбце *По умолчанию* данного языка будет установлен флаг. Данный язык будет использоваться в качестве языка по умолчанию в пользовательском интерфейсе Web-клиента и в программе Конструктор Web-разметок. Язык будет переключен, если локализация пользователя отсутствует в системе Docsvision.

Добавление нового языка

Чтобы добавить новый язык, выполните следующие действия:

1. Откройте для редактирования конфигурационный файл `C:\Program Files (x86)\Docsvision\WebClient\WebClientAdminConsole.appsettings.json`.
2. Добавьте в секцию **"Docsvision" > "WebClient" > "SettingGroups" > "Locales"** элемент `Setting`. В значении параметров `Name` и `Value` должна быть указана локаль.

   В следующем примере добавлена новая локаль — немецкий язык (идентификатор — 1031):

   ```
   <SettingGroup Name="Locales">
       <Setting Name="1049" Value="1049" />
       <Setting Name="1033" Value="1033" />
       <Setting Name="1058" Value="1058" />
       <Setting Name="1087" Value="1087" />
       <Setting Name="1062" Value="1062" />
   ```

```
    <Setting Name="1031" Value="1031" />
</SettingGroup>
```

3. Перезапустите сервер Web-клиента.

После перезапуска сервера язык станет доступен:

- Для включения/отключения, назначения языком по умолчанию в программе Панель управления Web-клиентом.
- Для выбора языком локализации в Web-клиенте.

  Если для текстовой метки не настроено локализованное значение, в такой метке отображается значение из английской локализации.

- Для локализации ресурсов, используемых в разметках в программе Конструктор Web-разметок.

Раздел "Решения"

Раздел *Решения* содержит инструменты импорта *решений* с разметками карточек.



Модуль "Web-клиент" поддерживает следующие варианты импорта решений:

- Из интерфейса Консоли управления Docsvision, без взаимодействия с программой Панель управления Web-клиентом, начиная с версии 6.1 сборки 1265 и выше: "6.2@mgmtconsole:servers:webclient-solutions.pdf"
- При помощи настройки в программе Панель управления Web-клиентом, в версиях ниже указанной, см. ниже

Инструменты импорта решений

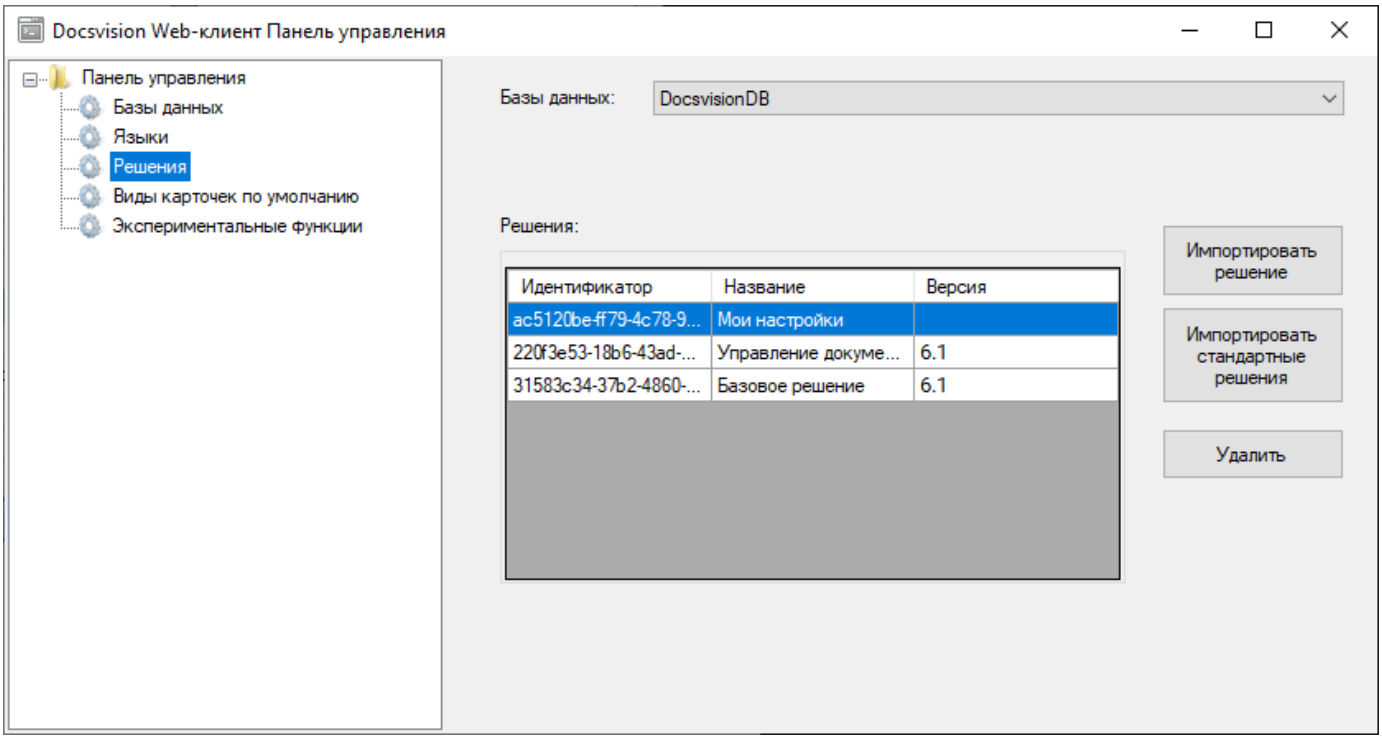


Рисунок 23. Страница настроек "Решения"

Таблица 4. Страница настроек "Решения"

Базы данных	Выбор БД Docsvision, в которую импортируются *решения*. В каждую БД можно импортировать собственный набор *решений*. Новые БД, можно добавить через *Консоль настройки Docsvision*.
Решения	Список решений, которые загружены в выбранную БД.
Импортировать решение	Импортирует в БД решение из файла XML. Расположение стандартных решений: `/lib/docsvision/webclient/LayoutSolutions/`.



В Web-клиенте версии 6.1 сборки 1265 и выше и выше расположение стандартных решений изменено: `/lib/docsvision/solutions/LayoutSolutions/WebClient` для Linux и `C:\Program Files\Docsvision\Solutions\LayoutSolutions\WebClient` для Windows.

Импортировать стандартные решения	Импортирует в БД все решения из папки `/lib/docsvision/webclient/LayoutSolutions/`.



В Web-клиенте версии 6.1 сборки 1265 и выше и выше расположение стандартных решений изменено: `/lib/docsvision/solutions/LayoutSolutions/WebClient` для Linux и `C:\Program Files\Docsvision\Solutions\LayoutSolutions\WebClient` для Windows.

Удалить	Удаляет выбранное решение из базы данных.



Перед загрузкой решений убедитесь, что служба Сервера Web-клиента запущена, чтобы избежать ошибок импорта.

Загрузка стандартных решений

Чтобы загрузить стандартные решения:

1. В раскрывающемся списке ***Базы данных*** выберете подключенную БД для загрузки стандартных *решений*.
2. Нажмите на кнопку **Импортировать стандартные решения**.

   Если база данных Docsvision содержит аналогичные загружаемым *решения* более ранних версий, существующие *решения* будут обновлены. Данное обновление не затрагивает собственные *решения*.

   После завершения загрузки *решений* в БД, они отобразятся в списке **Решения**.



Решения загружаются только в выбранную БД. Если к Web-клиенту подключено несколько БД, то *решения* нужно будет поочередно загрузить во все требуемые базы данных.

Загрузка прочих решений

Помимо загрузки всех стандартных *решений* администратор может загрузить отдельные *решения* из файлов.

Чтобы загрузить решения в БД:

1. Выберите в раскрывающемся списке ***Базы данных*** БД, в которую будут

загружены *решения*. В данном списке отображаются БД, которые были добавлены на странице через *Консоль настройки Docsvision*.

2. Нажмите на кнопку **Импортировать решение**.
3. Выберите файл с данными *решения*:

   - *Базовое решение*. Содержит разметки и условия выбора **типа** карточек Документ. Расположено в каталоге `/lib/docsvision/webclient/LayoutSolutions/LayoutSolutionsDocsvision.BackofficeSolution.xml`.

     В Web-клиенте версии 6.1 сборки 1265 и выше и выше расположение стандартных решений изменено: `/lib/docsvision/solutions/LayoutSolutions/WebClient` для Linux и `C:\Program Files\Docsvision\Solutions\LayoutSolutions\WebClient` для Windows.

   - Решение *Управление документами*. Содержит разметки и условия выбора стандартных **видов** карточек Документ приложения *Управление документами*. Расположено в каталоге. `/lib/docsvision/webclient/LayoutSolutions/LayoutSolutionsDocsvision.DocumentManagementSolution.xml`.

     В Web-клиенте версии 6.1 сборки 1265 и выше и выше расположение стандартных решений изменено: `/lib/docsvision/solutions/LayoutSolutions/WebClient` для Linux и `C:\Program Files\Docsvision\Solutions\LayoutSolutions\WebClient` для Windows.

   - Файл собственного *решения*.

   При импорте стандартных решений приоритет отдаётся решениям в виде пакета, в виде файла будут импортированы только те решения, у которых отсутствует вариант импорта в виде пакета.

Удаление решений

Для удаления *решения* из БД нужно выбрать его в списке *решений* и нажать на кнопку **Удалить**.

Преобразование решений

Если база данных содержит *решения*, формат которых не соответствует установленной версии Web-клиента, на странице будет отображаться предупреждение о необходимости конвертации и кнопка для её выполнения. Решения, нуждающиеся в конвертации, будут выделены курсивом.

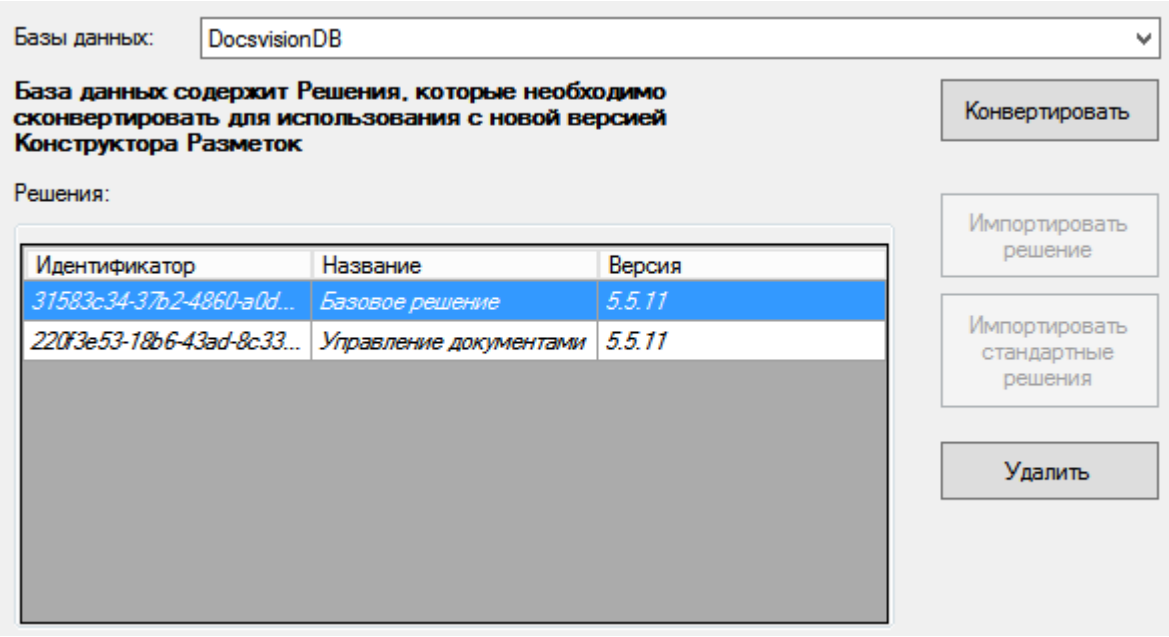


Рисунок 24. БД с Решениями, требующими конвертации

Для выполнения преобразования *решений*, нажмите кнопку **Конвертировать**.

Новые версии стандартных *решений* при этом не будут импортированы.

Раздел "Виды карточек по умолчанию"

Раздел *Виды карточек по умолчанию* открывает страницу настройки видов карточек, с которыми по умолчанию будут создаваться новые карточки соответствующих типов.

Модуль "Web-клиент" поддерживает следующие варианты настройки видов карточек по умолчанию:

- Из интерфейса Консоли управления Docsvision, без взаимодействия с программой Панель управления Web-клиентом, начиная с версии 6.1 сборки 1265 и выше: "6.2@mgmtconsole:servers:webclient-kinds.pdf"
- При помощи настройки в программе Панель управления Web-клиентом, в версиях ниже указанной, см. ниже

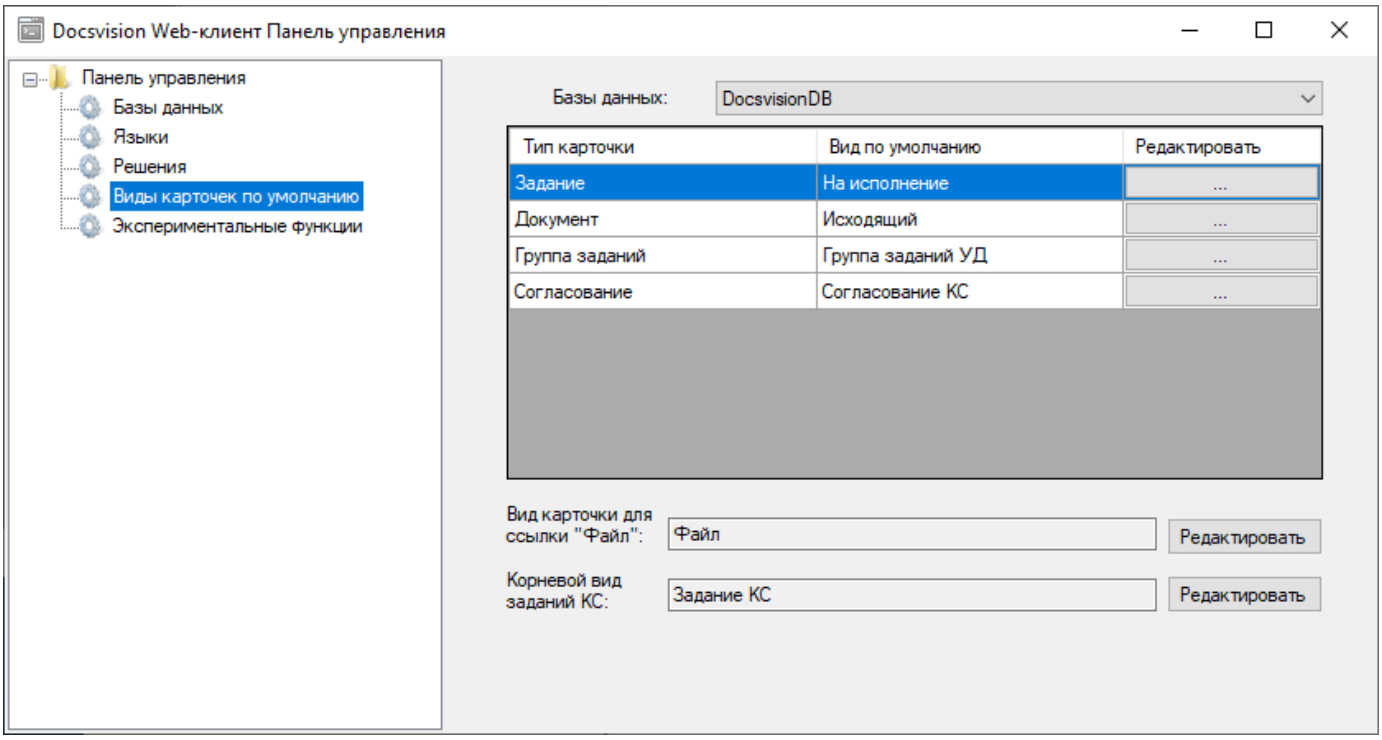


Рисунок 25. Страница настроек "Виды карточек по умолчанию"

Таблица 5. Страница настроек "Виды карточек по умолчанию"

Базы данных	Выбор БД Docsvision, содержащей карточки. Каждая БД может иметь собственный набор *карточек*. Новые БД, можно добавить через *Консоль настройки Docsvision*.
Таблица видов карточек	Список типов карточек Web-клиент и их видов по умолчанию, которые будут использованы при создании новых карточек.
`Вид карточки для ссылки 'Файл'`	Вид карточки, который используется для создания карточек по ссылке Файл.
`Корневой вид заданий КС`	Корневой вид карточек заданий согласования. Для корректной работы согласования, все задания, используемые в согласовании, должны быть дочерними к выбранному в `Корневой вид заданий КС`.

Изменение стандартных видов карточек

Для изменения стандартного вида для соответствующего типа карточек, нажмите на кнопку **Редактировать**. Для изменения списка нажмите кнопку **...**

(три точки) и выберите вид из списка.

Раздел "Экспериментальные функции"

Раздел *Экспериментальные функции* открывает страницу активации функций Web-клиента, находящихся в процессе разработки. Данные функции могут быть включены в основную функциональность Web-клиента, но на данном этапе разработки не могут быть рекомендованы для всех пользователей.

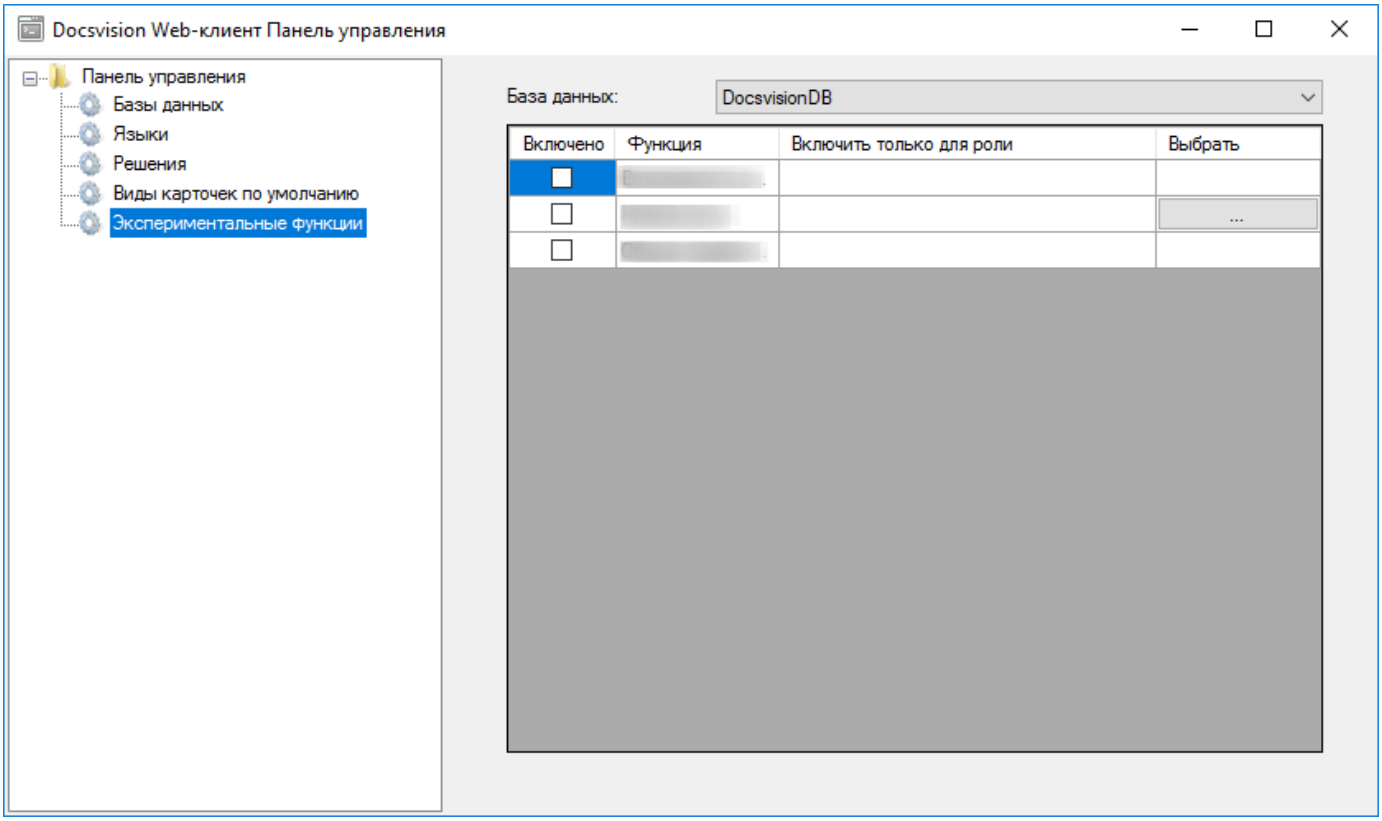


Рисунок 26. Страница настроек "Экспериментальные функции"

Таблица 6. Страница настроек "Экспериментальные функции"

Базы данных	Выбирает БД Docsvision, для которой выполняется настройка функций. Новые БД, можно добавить через *Консоль настройки Docsvision*.
Список доступных экспериментальных функций	См. Экспериментальные функции.

Функции администратора модуля Web-клиент

В разделе приведено описание *функций администратора*. В функции администратора входит предоставление доступа в Web-клиент, настройка дерева папок и дашборда.

Выполнение функций может потребовать определённых навыков и/или использования сторонних инструментов. В рамках документации Web-клиент приводятся инструкции только для настройки Web-клиента и некоторого сопутствующего программного обеспечения. Администратор должен самостоятельно ознакомиться с сопутствующим программным обеспечением и приобрести требуемые навыки для работы с системой Docsvision.

Создать администратора Web-клиента

Администратор Web-клиента — пользователь, которому предоставляются администраторские привилегии при работе с модулем. Когда администратор Web-клиента входит в Web-клиент, под профилем пользователя отображается псевдоним текущего тенанта.



Модуль "Web-клиент" поддерживает следующие варианты создания администратора:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группы пользователей"
- При помощи настройки в конфигурационном файле модуля Платформа, см. подробнее в документации модуля Платформа: "6.1@platform:admin:config-platform.pdf"

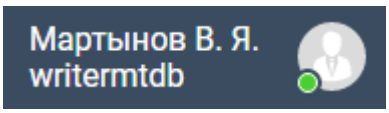


Рисунок 27. Отображение текущего тенанта

Администратор Web-клиента — пользователь:

- Зарегистрированный в *Справочнике сотрудников* системы Docsvision.
- Включённый в группу **Администраторы** в конфигурационном файле Сервера приложений (модуля Платформа) или Консоли управления Docsvision. См. подробнее в документации модуля *Платформа*, раздел "Конфигурация Сервера приложений", и документации Консоли управления Docsvision, раздел "Управление доступом".

Настройка входа в Web-клиент

Начиная, с версии 6.2, сборка 1444 и выше, в Web-клиент могут входить пользователи, добавленные в справочник сотрудников или в группу **Администраторы**.

Сотрудники, входящие в группу **Администраторы** (например, добавленные в группу через Консоль управления Docsvision), могут войти в Web-клиент, даже если их учётная запись отсутствует в справочнике сотрудников.

Если пользователь отсутствует в справочнике сотрудников, но является членом группы **Администраторы**, перед выполнением входа автоматически создается запись о сотруднике в Справочнике сотрудников. Запись о сотруднике будет добавлена в первую по списку организацию, если в справочнике есть хотя бы одна организация. Если в справочнике нет ни одной организации, будет создана организация с именем `DocsVision`, в которую будет добавлен сотрудник. Заполняется только поле *Пользователь*, остальные поля не заполняются

Если пользователь отсутствует в Справочнике сотрудников и не входит в группу **Администраторы**, он не сможет получить доступ в Web-клиент. При попытке входа появляется сообщение об ошибке: `Не удалось войти: Сотрудник с именем пользователя не найден в Справочнике сотрудников и не является администратором`.

Предоставить доступ в Web-клиент

- Чтобы предоставить доступ к Web-клиенту пользователю, добавьте его в *Справочник сотрудников*.
- Пользователи, добавленные в группу **Администраторы**, например через Консоль управления Docsvision, могут войти в Web-клиент, даже если их учётная запись отсутствует в справочнике сотрудников, начиная, с версии 6.2, сборка 1444 и выше.
- Пользователь не получит доступ к Web-клиенту, если отсутствует в *Справочнике сотрудников* и не является администратором.

Настройка способов входа

Чтобы предоставить пользователю возможность авторизоваться с помощью сторонних сервисов, администратор должен:

1. Настроить внешний вид страницы входа по инструкции
2. Выполнить необходимые настройки Docsvision см. подробнее "Расширения авторизации".

3. Выполнить настройку в карточке сотрудника на вкладке "Безопасность".

Настройки капчи

Web-клиент поддерживает механизм под названием капча (Captcha), предназначенный для защиты от попыток автоматического перебора пароля пользователя при входе в систему. Администратор настраивает количество неудачных попыток ввода пароля, после которых срабатывает капча, а также время до сброса счётчика неудачных попыток.

Обратите внимание, капча срабатывает, только если сервер Web-клиент работает по HTTPS, см. "Подключение к Docsvision по HTTPS".

```
{
  "Docsvision": {
    "WebClient": {
        "SettingGroups": {
          "WebClient": {
            "MaxFailedLoginAttempts": 3, ①
            "FailedLoginAttemptsTTL":  300 ②
          }
        }
    }
  }
}
```

① Количество неудачных попыток ввода пароля.

② Время в секундах, после которого счётчик неудачных попыток будет сброшен.

Неудачные попытки входа и срабатывания капчи попадают в журнал Web-клиента. Попытки входа будут записаны в журнал при уровне журналирования как `Error` и серьёзнее, срабатывания капчи будут записаны в журнал при уровне `Warn` и серьёзнее.

Работа в мультитенантном режиме

Web-клиент поддерживает работу в мультитенантном режиме без дополнительных настроек. В мультитенантном режиме имеет значение адрес открытия Web-клиента. Для правильной работы мультитенантного режима необходимо выполнить следующие условия:

☑ Во всех БД, версии модулей должны соответствовать требованиям, если планируется работать в мультитенантном режиме.

☑ Вводимый адрес с указанием псевдонима БД должен настроен на соответствие адресу подключения к серверу Docsvision.

☑ Адрес должен быть введён правильно.

До первой точки указывается псевдоним БД, в которой планируется работа, например, `http://oldDB.company.com:5004/`.

Если все условия выполнены, Web-клиент сможет подключиться к любой указанной БД. При загрузке страницы будет предложено авторизоваться в новой БД. Такая схема позволяет работать одновременно в нескольких БД от имени разных пользователей.

Авторизация в мультитенантном режиме

Чтобы авторизоваться под именем другого пользователя:

1. Откройте Web-клиент в новой вкладке, указав в адресе точку подключения к БД, например, `http://newDB.company.com:5004/`.

Текущая активная БД в мультитенантном режиме указывается рядом с фотографией, под именем текущего пользователя, только если пользователь является Администратором. Текущую БД всегда можно проверить из меню "О программе".

Если указать адрес по умолчанию без псевдонима БД, Web-клиент подключится к БД, заданной по умолчанию в *Консоль настройки Docsvision*.

Если адрес с псевдонимом БД указан корректно, но Web-клиент всё равно не подключается к нужной БД, убедитесь ещё раз, что выполнены все требования.

Если адрес подключения к серверу Docsvision выглядит следующим образом: `http://company.com:5004/`

Адрес подключения Web-клиента к БД с псевдонимом **newDB** будет следующий: `http://newDB.company.com:5004/`.

Псевдоним БД указывается до первой точки в адресе.

Администратор может настроить адрес подключения с помощью файла *hosts* или других инструментов.

Настройка работы с несколькими доменами

Docsvision поддерживает работу в одном или нескольких доменах. Для этого в конфигурационном файле модулей Платформа и Web-клиент поддерживается новая расширенная настройка доменных каталогов — параметр `"Catalogs"`.



Модуль "Платформа" поддерживает следующие настройки работы с несколькими доменами:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 832 и выше: "Настройка доменных каталогов"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже



Возможность настройки со старыми параметрами настройки через параметр `"Domains"` (для модуля Платформа) и параметр `"DefaultDomain"` (для модуля Web-клиент) сохраняется.

В конфигурационных файлах настройка представлена следующим массивом: `"Catalogs": []`. Если в массиве есть хоть одно описание каталога, будут учитываться только настройки из `"Catalogs": []`.

Пример заполнения:

```
{
"Catalogs": [
  {
    "FullDomainName": "example1.com", ①
    "NetBiosDomainName": "example1", ②
    "LdapOptions": { ③
      "LdapServerAddresses": [ "example1.com" ], ④
      "AuthType": "Basic", ⑤
      "Credential": { ⑥
        "UserName": "user@example1.com",
        "Password": "Password"
      }
    }
  },
  {
    "FullDomainName": "example2.com", ⑦
    "ChallengeTo": "example1.com" ⑧
  }
]
```

```
}
```

① `FullDomainName` — полное имя домена указывается в формате `user@example.com` учётной записи после `@`.

② `NetBiosDomainName` — `NetBios`-имя домена указывается в формате `domain\\user` перед `\\`.

③ `LdapOptions` — настройки подключения к LDAP каталогу домена.

④ `LdapServerAddresses` — адреса серверов LDAP данного домена. Если адрес соответствует полному имени домену, параметр можно удалить.

⑤ `AuthType` — тип аутентификации в LDAP.

⑥ `Credential` — логин и пароль пользователя для подключения к LDAP-каталогам текущего домена. Пароль для подключения можно зашифровать, см. подробнее раздел "Защита системной учётной записи" в руководстве по настройке системы.

⑦ `FullDomainName` — полное имя домена указывается в формате `user@example.com` учётной записи после `@`.

⑧ `ChallengeTo` — настройка позволяет переадресовывать запросы от выбранного домена к каталогам указанного домена, см. далее.

Раздел `Catalogs` может содержать массив настроек вида:

```
{
   "FullDomainName": "example.com", ①
   "NetBiosDomainName": "EXAMPLE", ②
   "LdapOptions": { ③
     "LdapServerAddresses": [ "example.com" ], ④
     "AuthType": "Basic", ⑤
     "Port": 389, ⑥
     "Credential": { ⑦
       "UserName": "user@example.com",
       "Password": "Password"
     }
   }
}
```

① `FullDomainName` — полное имя домена указывается в формате `user@example.com` учётной записи после `@`.

② `NetBiosDomainName` — `NetBios`-имя домена указывается в формате `domain\\user` перед `\\`.

③ `LdapOptions` — настройки подключения к LDAP каталогу домена.

④ `LdapServerAddresses` — адреса серверов LDAP данного домена. Если адрес соответствует полному имени домену, параметр можно удалить.

⑤ `AuthType` — тип аутентификации в LDAP.

⑥ `Port` — порт подключения к LDAP каталогам, перечисленным в `LdapServerAddresses`.

⑦ `Credential` — логин и пароль пользователя для подключения к LDAP-каталогам текущего домена. Пароль для подключения можно зашифровать, см. подробнее раздел "Защита системной учётной записи" в руководстве по настройке системы.

Настройка "ChallengeTo"

Для большей гибкости предусмотрена настройка `ChallengeTo`, позволяющая переадресовывать запросы от выбранного домена к каталогам указанного домена. Например, если пользователи в домене `example.com`, имеют учётную запись `user@example1.com`, конфигурационный файл можно изменить следующим образом:

```
{
    "FullDomainName": "example1.com",
    "NetBiosDomainName": "example1",
    "ChallengeTo": "example2.com"
}
```

Осуществляется проверка значения, указанного после `@` (для UPN-формата) или до `\` (для NT-формата), проводится поиск соответствия в `Catalogs`. Если у такого соответствия есть `ChallengeTo`, далее аутентификация пользователя будет проверяться по LDAP серверу того каталога, который указан у домена в `ChallengeTo`.

После выполнения вышеуказанной настройки аутентификация и получение информации о пользователе `user@example1.com` будут осуществляться в каталогах домена `example2.com`.

Настройка `ChallengeTo` применима именно при наличии UPN-суффиксов внутри одного домена, если доменов несколько, рекомендуется добавить столько же записей с указанием отдельных LDAP-серверов для этих доменов.

При такой настройке существует ограничение. Встроенная (сквозная) аутентификация будет работать только для тех пользователей домена, в который введен сервер на Linux.



Сервер Docsvision, установленный на Linux, требует дополнительной настройки для использования технологии единого входа (когда пользователь уже аутентифицирован в другом модуле системы). Подробнее настройка описана в Руководстве по настройке системы, см. "Настройка прозрачной аутентификации".

Кэширование информации о пользователе от LDAP

Следующие данные пользователя могут кэшироваться в ответах от LDAP серверов[1]:

- `SID` — идентификатор безопасности (Security Identifier).
- `SAM` — имя пользователя в формате SAM (Security Account Manager).
- `UPN` — имя пользователя в формате UPN (User Principal Name).
- `WindowsAccountName` — имя пользователя в формате NTAccount (NetBios домен\имя пользователя в формате SAM).
- `DomainName` — полное имя домена пользователя.
- `NetBiosDomainName` — NetBios имя домена пользователя.



Кэширование ответов от LDAP серверов по умолчанию отключено.

Кэширование ответов от LDAP серверов по умолчанию отключено. Чтобы включить кэширование, необходимо добавить параметр `UseCatalogsResponseCache` со значение `true` в конфигурационные файлы модулей Платформа и Web-клиент. Параметр добавляется в корень конфигурационного файла модулей Платформа или Web-клиент:

```
{
  "UseCatalogsResponseCache": true
}
```

Временем кэширования ответов можно управлять, по умолчанию время кэширования составляет 24 часа. Чтобы изменить время, необходимо добавить параметр `CatalogsResponseCacheExpirationInSeconds` и указать в нём нужное

значение в секундах.

Пример включения кэширования ответов на 1 минуту:

```
{
  "UseCatalogsResponseCache": true, ①
  "CatalogsResponseCacheExpirationInSeconds": 60 ②
 }
```

① Кэширование включено.

② Время кэширования — 60 секунд.

Кэширование работает только совместно с расширенной конфигурацией доменов — конфигурация доменов определена в разделе Catalogs.



Обратите внимание, параметры настройки необходимо добавлять в конфигурационные файлы модулей Платформа и Web-клиент.

При включенном журналировании в режиме трассировки предоставление данных из кэша диагностируется сообщением вида:

```
LDAP query data from cache returned for account [имя учётной записи/SID]
```

Расширения авторизации

Web-клиент поддерживает авторизацию с помощью расширений авторизации OpenID: Microsoft Azure, ADFS и ЕСИА. Подробную инструкцию по настройке см. в Руководстве по настройке, раздел "Регистрация расширений авторизации".

Настройка компонентов КриптоПро

Для формирования усиленной подписи (с использованием сертификатов ЭП) необходимо установить компоненты КриптоПро и сертификаты.



Для работы с электронной подписью необходимы компоненты Криптопро. По вопросам их приобретения и лицензирования обращайтесь в компанию ООО "КРИПТО-ПРО".

Формирование усиленной подписи возможно только на персональных компьютерах, соответствующих требованиям:

- Операционная система: Windows, Mac OS или Linux (версии см. в требованиях КриптоПро CSP).
- Веб-браузер: Google Chrome, Яндекс.Браузер или Microsoft Edge на Chromium (поддерживаемые версии приведены в требованиях к модулю).

1. Установите и активируйте расширение для браузера в зависимости от вашей операционной системы и используемого браузера:
   - Windows
   - Linux
   - MacOS

   Если расширение плагина имеется в списке установленных, но отключёно, необходимо самостоятельно включить его в браузере.

   a. Если расширение плагина имеется в списке установленных, но отключёно, необходимо самостоятельно включить его в браузере.

   b. Если расширение отсутствует в списке или недоступно в магазине приложений, вы можете установить расширение, воспользовавшись инструкцией на сайте КриптоПро. Вы также можете продолжить работу с электронной подписью в браузерах Chromium-Gost или Яндекс Браузер с полной поддержкой необходимой функциональности.
2. Установите сертификат пользователя.
   - Установите **неквалифицированный** сертификат в хранилище "Личное" на уровне "Текущий пользователь".
   - Установите **квалифицированный** сертификат на компьютер СКЗИ КриптоПро CSP в хранилище *Личное* на уровне *Текущий пользователь*.
3. Установите корневой сертификат удостоверяющего центра[2] в хранилище *Доверенные корневые центры сертификации* на уровне *Локальный компьютер*.

Настройка использования облачной подписи



Модуль "Web-клиент" поддерживает следующие способы настроить использование облачной подписи:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Облачная подпись"

- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже



Запросы к облачному сервису электронной подписи отправляются через сервер Web-клиента. Это означает, что Web-клиент должен иметь доступ к сервису ЭП для отправки запросов на проверку подписи и для авторизации.

Проверка подлинности подписи может осуществляться локально или через облачный сервис электронной подписи. Способ проверки подписи зависит от настроек в файле конфигурации Web-клиента.

Чтобы изменить способ проверки подлинности подписи из конфигурационного файла:

1. Откройте файл конфигурации Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к параметру `menu:"Docsvision"["WebClient" > "SettingGroups"` и выполните следующие настройки:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "CloudSignature": {
          "Address": "https://адрес-облачного-сервиса-DSS", ①
          "ClientId": null, ②
          "SignServerAppName": null, ③
          "STSAppName": null, ④
          "DocumentStoreAppName": null, ⑤
          "VerifyUrl": "https://dss.cryptopro.ru/verify/rest/api/signatures", ⑥
          "TSPServiceAddress": null, ⑦
          "SignatureType": "BES", ⑧
          "AuthorizeRedirectUrl": "http://адрес-Web-клиента-
Docsvision/api/CryptoProDss/AcceptAuthorization", ⑨
          "TokenStorageMode": "InMemory" ⑩
        }
      }
    }
  }
}
```

① Адрес сервиса облачной подписи.

② Идентификатор клиента. Настраивается в сервисе облачной подписи КриптоПро.

③ Сервис подписи, обрабатывающий создание транзакций и получение результатов, подтвержденной операции.

④ Сервис аутентификации, ответственный за аутентификацию пользователей для возможности обращений к сервису подписи и подтверждение транзакций.

⑤ `SignServerAppName`, `STSAppName` и `DocumentStoreAppName` — имена сайтов на сервере КриптоПро DSS. Настраиваются при установке сервиса электронной подписи КриптоПро DSS.

⑥ URL для проверки подлинности подписи.

⑦ Адрес сервиса штампа времени для XLT1.

⑧ Тип подписи:

- `BES` — будет использоваться CAdES-BES.
- `XLT1` — будет использоваться CAdES-XLT1.
- `T` — будет использоваться CAdES-T.

⑨ URL, на который будет переадресован пользователь после авторизации в сервисе электронной подписи. Настраивается в сервисе электронной подписи.

⑩ Режим хранения токена авторизации.
Возможные значения:

- `Disabled` — авторизация будет запрашиваться каждый раз.
- `InSessionCache` — авторизация будет повторно запрошена по окончании сессии.
- `InMemory` — авторизация будет запрошена при перезапуске службы Сервера Web-клиента.
- `InUserProfileEncrypted` — авторизация будет запрошена по истечении срока действия токена.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они

> считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Проверка облачной подписи

Для проверки подлинности подписи при использовании облачных сертификатов необходимо выполнить дополнительную настройку.

1. Запросите корневой сертификат КриптоПро для облачного сертификата.
2. Установите корневой сертификат на сервере Web-клиента.
3. Убедитесь, что на сервере Web-клиента установлены компоненты КриптоПро CSP версии 5.0 и выше.
4. При необходимости установите также компоненты TSP Client и OCSP Client.
   a. Выполните соответствующие настройки TSP и OCSP.
5. После выполнения настроек станет доступна проверка подписей с использованием облачных сертификатов.

Настройки интерфейса Web-клиента

Web-клиент

Настройки интерфейса Web-клиента

Информация о доступных администратору настройках интерфейса Web-клиента: закрепление главного меню, режим отображений боковой панели, настройка создания карточек и другие настройки.

- Режим закрепления основного меню
- Режим скрытия правой боковой панели
- Настройка меню создания карточек
- Больше подробностей в навигационном меню слева…

Режим закрепления основного меню

По умолчанию Основное меню Web-клиента (с деревом папок) закрепляется/открепляется с помощью кнопок . Администратор может переключить режим закрепления основного меню: основное меню будет закрепляться/открепляться при нажатии кнопки . При этом кнопки

откреплення/закрепления Основного меню будут скрыты.



Модуль "Web-клиент" поддерживает следующие способы изменить режим закрепления основного меню:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Интерфейс""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы изменить режим закрепления основного меню из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `UXMainMenuBehavior`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "UXMainMenuBehavior": "0" ①
        }
      }
    }
  }
}
```

① Возможные значения:

- `0` — для использования режима переключения кнопками /.
- `1` — для использования режима переключения кнопкой .



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они

считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.

Режим скрытия правой боковой панели

На правой боковой панели отображаются параметры отправки документа на согласование, на ознакомление и пр. По умолчанию панель скрывается при нажатии на ней кнопок выполнения: **ОК**, **Отмена** и пр. Администратор может включить режим, при котором панель также будет скрываться по щелчку в любой области страницы Web-клиента вне боковой панели.



Модуль "Web-клиент" поддерживает следующие способы изменить режим скрытия правой боковой панели:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Интерфейс""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы изменить режим скрытия панели из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" > "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `HideSidebarOnPageClick`:

   Пример включения режима, при котором панель будет скрываться нажатием кнопки **ОК**/**Отмена** и пр., а также при щелчке вне боковой панели:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "HideSidebarOnPageClick": "True" ①
        }
      }
    }
```

```
  }
}
```

① Возможные значения:

- `False` (значение по умолчанию) — панель должна скрываться только при нажатии кнопки **ОК**/**Отмена** и пр.
- `True` — панель также должна скрываться при щелчке вне боковой панели.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.

Настройка меню создания карточек

Администратор Web-клиента может изменить настройки меню создания карточек: добавить или удалить типы карточек, изменить последовательность отображения типов.



Модуль "Web-клиент" поддерживает следующие варианты настройки меню создания карточек:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Интерфейс""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы изменить настройки меню создания карточек из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к элементу `"Docsvision" › "WebClient" › CardTypes`.

3. Измените параметр `CardTypeId`:



Web-клиент поддерживает работу только с карточками, созданными на основе *Базовых объектов*.

```
{
  "Docsvision": {
    "WebClient": {
      "CardTypes": [
        {
          "CardTypeId": "b9f7bfd7-7429-455e-a3f1-94ffb569c794", ①
          "CssClass": "document", ②
          "SaveInFolder": "true" ③
        },
        {
          "CardTypeId": "c7b36f33-cdd4-4da9-8444-600fe14111e4",
          "CssClass": "task",
          "SaveInFolder": "true"
        }
      ]
    }
  }
}
```

① "CardTypeId": "GUID-идентификатор типа карточки".
Чтобы изменить порядок кнопок создания типов карточек, измените порядок следования GUID-идентификаторов (`CardTypeId`) в `CardTypes`.

② Чтобы изменить цвет строки типа карточки в меню создания карточек, добавьте в элемент `CardType` параметр `CssClass` с названием класса CSS в значении.

③ Чтобы новые карточки выбранного типа сохранялись в текущей открытой папке, добавьте в элемент `CardType` параметр `SaveInFolder` со значением `true`. Если `SaveInFolder` имеет значение `false` (или отсутствует), ярлыки карточек не будут добавляться в текущую папку.

4. Сохраните изменения конфигурационного файла.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

> Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Чтобы изменение цвета строки типа карточки в меню создания работало корректно, к Web-клиенту должен быть подключен файл стилей с CSS-классом `.Значение-атрибута-CssClass-color`:

```
.document-color {
    background-color: #639;
}
```

Отключить создание карточек определённого типа

В случае необходимости, администратор системы может отключить возможность создания из Web-клиента карточек определённого типа. Когда функция отключена, соответствующий тип карточек будет скрыт из списка доступных для создания типов карточки вместе с подвидами:

- При нажатии кнопки **+ Создать** на панели управления.
- При создании карточки с добавлением ссылки из карточек *Документ* и *Задание*.



> *Модуль "Web-клиент" поддерживает следующие способы отключить создание карточек определённого типа:*
>
> - Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Внешний вид"
> - При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы скрыть тип карточки из меню создания через конфигурационный файл:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к элементу **"Docsvision" › "WebClient" › CardTypes**.

   В элементе `CardTypes` перечислены типы карточек, с которыми работает Web-

клиент:

```
{
  "Docsvision": {
    "WebClient": {
      "CardTypes": [
        {
          "CardTypeId": "b9f7bfd7-7429-455e-a3f1-94ffb569c794", ①
          "CssClass": "document",
          "SaveInFolder": "true"
        },
        {
          "CardTypeId": "c7b36f33-cdd4-4da9-8444-600fe14111e4", ②
          "CssClass": "task",
          "SaveInFolder": "true"
        },
        {
          "CardTypeId": "c7b36f33-cdd4-4da9-8444-600fe14111e4", ③
          "CssClass": "task",
          "SaveInFolder": "true",
          "DisableCreateFlags": "0" ④
        }
      ]
    }
  }
}
```

① Документ.

② Задание.

③ Группа заданий.

④ Создайте или измените параметр `DisableCreateFlags` в элементе нужного типа карточек.
Значение параметра `DisableCreateFlags` является суммой цифр, каждая из которых определяет один из вариантов скрытия типа карточки:

- `0` — не скрывать.
- `1` — скрывать тип из списка типов при создании карточки при нажатии кнопки **+ Создать** на панели управления.
- `2` — скрывать тип из списка типов при создании карточки с добавлением ссылки из карточек *Документ* и *Задание*, а также при создании карточки в качестве отчёта к *Заданию*.

- 4 — скрывать тип из списка типов при создании связанного задания карточки *Документ*.
- 8 — скрывать тип из списка типов при создании подчиненного задания карточки *Задание*.

3. Сохраните изменения конфигурационного файла.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Чтобы отключить любую возможность создания карточки *Группа заданий* из Web-клиента, укажите значение ***15*** (= 1+2+4+8). Чтобы скрыть тип карточки *Документ* из списка типов при создании новой карточки, укажите значение 1.



Параметр `DisableCreateFlags` не действует, когда меню создания карточки вызывается из настраиваемой разметки.

Порядок кнопок "ОК/Отмена" по умолчанию



Модуль "Web-клиент" поддерживает следующие варианты настройки порядка кнопок:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Интерфейс""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Порядок кнопок **ОК**/**Сохранить** и **Отмена**, отображаемых в диалоговых окнах и разметках Web-клиента, может быть настроен. Пользователь может настроить схему расположения в меню *Профиль*. Администратор может установить порядок расположения по умолчанию, который будет действовать до выбора нового

значения пользователем.

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" > "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `DefaultButtonPosition`:

   Включение режима "Отмена/ОК"

   ```
   {
     "Docsvision": {
       "WebClient": {
         "SettingGroups": {
           "WebClient": {
             "DefaultButtonPosition": "Auto" ①
           }
         }
       }
     }
   }
   ```

   ① Возможные значения:

   - `Auto` — значение по умолчанию — порядок отображения кнопок определяется типом ОС пользователя.
     Для ОС MacOS: **Отмена**/**ОК**.
     Для ОС Windows, Linux, Android и др. — **ОК**/**Отмена**
   - `OkCancel` — **ОК**/**Отмена**
   - `CancelOk` — **Отмена**/**ОК**

4. Сохраните изменения конфигурационного файла.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Анимация на странице аутентификации

На странице аутентификации пользователя раньше использовалась анимация фонового изображения.

Анимация регулировалась настройкой `EnableLoginScreenAnimation` в конфигурационном файле Web-клиента. Эта настройка перестала быть актуальна и была упразднена.

Изменить путь к пользовательской документации

Если требуется, администратор может разместить на собственном сайте документацию Docsvision, ссылка на которую приведена в Web-клиенте.

По умолчанию ссылка ведёт к документации на сайте компании "ДоксВижн". Инструкция по переносу документации приведена в документации по администрированию модуля *Платформа*, см. раздел Изменить расположение пользовательской документации.

Шаблоны для определения типа клиентского устройства



Модуль "Web-клиент" поддерживает следующие способы настройки определения типа клиентского устройства:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Сайт""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Пользовательский интерфейс Web-клиента адаптируется под тип клиентского устройства, который определяется по User Agent, передаваемому веб-браузером клиента. Если User Agent не сопоставляется или некорректно сопоставляется с типом устройства, нужно изменить шаблон, по которому выполняется сопоставление:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к элементу **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `TabletRegExpString`:

Определять устройство с User Agent "NEWPAD" как планшет:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "TabletRegExpString": "SM-T355|MediaPad|NEWPAD" ①
        }
      }
    }
  }
}
```

① В значении параметра через символ | нужно указать User Agent соответствующего устройства.

- `MobileRegExpString` — шаблон определения мобильных устройств нового поколения.
- `MobileRegExpPart2String` — шаблон определения старых мобильных устройств.
- `TabletRegExpString` — шаблон определения планшетов.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.



Адаптивные свойства разметок могут быть настроены в программе Конструктор Web-разметок.

Настройки агрегации

Новый грид позволяет настроить агрегацию строк. Агрегация — сбор суммарных данных — позволяет получать численные результаты по данным, содержащимся

в строках таблицы. Например, в представлении может отображаться количество документов определённого вида или количество документов, отправленных определённому пользователю.

Инструкцию по установке, настройке и требования к системе для установки Elasticsearch см. в документации модуля Платформа: Установка, Настройка, Требования.

Доступность агрегации

Для работы с агрегацией требуется иметь опцию *Docsvision Модуль Интеллектуальный поиск* лицензии Docsvision. Если опция отключена, Elasticsearch работает как полнотекстовый поиск без функции агрегации.

Агрегация доступна только для пользователей в следующих системных группах или группах в Справочнике сотрудников:

- *Использование агрегатов* (**DocsVision Aggregation Users**).
- *Администраторы* (**DocsVision Administrators**).
- *Продвинутые пользователи* (**DocsVision Power Users**).
- *Администраторы безопасности* (**DocsVision Security Administrators**).

> В Консоли управления Docsvision поддерживается возможность добавить пользователей в группы без необходимости работы с форматом `.json`: "Консоли управления Docsvision"

Сотрудники, включённые в эти группы, получают доступ к использованию агрегации с Elasticsearch. Для пользователей, которым не разрешена работа с агрегацией, не отображается дополнительная область в интерфейсе Web-клиента. При использовании Elasticsearch для таких пользователей будут выводиться только результаты поиска.

Настройки включения в группы применятся только в новой сессии пользователя. Чтобы настройки применились, необходимо повторно выполнить вход в Web-клиент.

Изменить количество вариантов агрегата

По умолчанию количество выводимых вариантов в агрегате равно 100.

Чтобы изменить это значение, в таблицу БД **dvsys_settings** добавьте настройку `FullTextIndexing_AggResultCount`, максимально возможная настройка `65,536` (65536

вариантов).

Задержка обновления агрегации



Модуль "Web-клиент" поддерживает следующие способы настройки агрегации:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Интерфейс""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Администратор может настроить задержку обновления списка карточек и доступных агрегатов при использовании Elasticsearch.

Чтобы настроить автоматическое применение по таймеру из конфигурационного файла:

1. Откройте файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. В секции **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"** измените значение параметра `Grid`:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "Grid": {
          "ApplyAggregatesTimeout": "1000", ①
          "ApplyAggregatesButton": "True" ②
        }
      }
    }
  }
}
```

① По истечении указанного количества миллисекунд будет автоматически обновлён список карточек и доступные агрегаты.

② Флаг определяет, будет ли список карточек обновляться автоматически или перестраиваться по нажатию кнопки **Применить**.
Доступные агрегаты обновляются согласно настройке `ApplyAggregatesTimeout`.



В конфигурации по умолчанию параметры отсутствуют.

Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Включить диаграммы на главной странице

На главной странице Web-клиента возможно отображение диаграмм. По умолчанию диаграммы не отображаются, их нужно включить.

Чтобы включить диаграммы на главной странице:

1. Откройте *Конструктор Web-разметок*.
2. Перейдите в локацию **WebFrame** › **Главная страница**.
3. Настройте условия выбора для активной разметки.
4. Добавьте условие использования для разметки DV.Dashboard_DocumentManagement_Diagram.
5. При необходимости измените свойства элементов управления для диаграмм: `incomingDiagram`, `outgoingDiagram`, `incomingDynamicsDiagram`.



- При изменении свойств элемента управления рекомендуется очистка кэша браузера.
- ЭУ не работает в браузере Internet Explorer.
- ЭУ не предназначен для работы с большим количеством карточек.
- ЭУ не работает с представлением *Дайджест*.

Настройки работы с карточками

Web-клиент

Настройки работы с карточками

Информация о настройках количества карточек в папке, размера файлов в карточках и списка разрешённых карточек.

- Ограничить количество отображаемых карточек в папке
- Ограничить размер прикладываемых файлов
- Изменить список разрешённых для добавления файлов
- Больше подробностей в навигационном меню слева…

Ограничить количество отображаемых карточек в папке

Настройка не поддерживается в новом гриде.

Ограничить размер прикладываемых файлов

В Web-клиенте по умолчанию действует несколько ограничений, влияющих на возможность прикладывания файлов большого размера:

- Максимальный размер файла, прикладываемого при создании карточки (параметр `MaxUploadFileItemSize`). Значение по умолчанию: 50 Мб.
- Ранее действовала настройка для указания максимального размера тела запроса (параметр `MaxRequestBodySize`), в новейших версиях Web-клиента настройка отключена.

Администратор системы может изменить действующее ограничение.



Модуль "Web-клиент" поддерживает следующие способы ограничить размер прикладываемых файлов:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Работа с файлами""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

1. Чтобы изменить ограничение для максимального размера единичного файла, в конфигурационном файле `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows перейдите к параметру **"Docsvision" › "WebClient" › View › MaxUploadFileItemSize"**.

```
{
  "Docsvision": {
    "SettingGroups": {
```

```
      "WebClient": {
        "View": {
          "MaxUploadFileItemSize": "0" ①
        },
      },
    },
  },
}
```

① Укажите новое ограничение для максимального размера единичного файла в байтах, не превышающее 2147483647.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

2. Сохраните изменения конфигурационного файла.

Ограничить максимальный размер файла для предпросмотра

В Web-клиенте по умолчанию действует ограничение предпросмотра файлов больше 100 Мб.

Администратор системы может изменить действующее ограничение.



Модуль "Web-клиент" поддерживает следующие способы ограничить рмаксимальный размер файла для предпросмотра:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Работа с файлами""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы изменить ограничение для максимального размера файла для предпросмотра из конфигурационного файла:

1. Откройте конифгурационный файл Web-клиента: `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.

2. В категории настроек **"Docsvision" > "WebClient" > "SettingGroups" > System** измените параметр `MaxPreviewFileSize`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "MaxPreviewFileSize": "0" ①
        }
      }
    }
  }
}
```

① Укажите значение в байтах.
Увеличение значения повысит нагрузку на сервер и время открытия файла на предпросмотр.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

3. Сохраните изменения конфигурационного файла.

Изменить список разрешённых для добавления файлов

По умолчанию пользователь может добавлять в карточки файлы со следующими расширениями

`rtf`, `.dot`, `.doc`, `.dotx`, `.ods`, `.docx`, `.odt`, `.csv`, `.xlt`, `.xls`, `.xltx`, `.xlsx`, `.pdf` (в т.ч `PDF/A-1a`), `.jpg`, `.jpeg`, `.gif`, `.png`, `.bmp`, `.tiff`, `.tif`, `.emf`, `.wmf`, `.xslt`, `.xsl`, `.xml`, `.htm`, `.html`, `.txt`, `.log`, `.cs`, `.asmx`, `.ascx`, `.aspx`, `.js`, `.cpp`, `.h`, `.css`, `.idl`, `.ini`, `.java`, `.rc`, `.rc2`, `.shtm`, `.shtml`, `.sql`, `.zip`, `.rar`, `.7z`, `.vsd`, `.vsdx`, `.msg`, `.one`, `.pptx`, `.ppt`, `.avi`, `.wav`, `.docm`, `.xltm`, `.dotm`, `.xlsm`, `.p7s`, `.sign`, `.sig`, `.sgn`.

Администратор может добавить в список новые расширения файлов или удалить существующие.



Модуль "Web-клиент" поддерживает следующие способы изменить список разрешённых для добавления файлов::

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Работа с файлами""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы изменить список разрешённых для добавления файлов из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `AcceptableFileExtensions`. Новые расширения файлов нужно добавлять через символ `|` в виде регулярного выражения.

   Список разрешённых файлов и добавление `.dwg` в допустимые расширения:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "AcceptableFileExtensions": "(
\\b)(rtf|dot|doc|dotx|ods|docx|odt|csv|xlt|xls|xltx|xlsx|pdf|jpg|jpeg|gif|png|bmp|tiff
|tif|emf|wmf|xslt|xsl|xml|htm|html|txt|log|cs|asmx|ascx|aspx|js|cpp|h|css|idl|ini|java
|rc|rc2|shtm|shtml|sql|zip|rar|7z|vsd|vsdx|msg|one|pptx|ppt|avi|wav|docm|xltm|dotm|xls
m|p7s|sign|sig|sgn|dwg)$"
        }
      }
    }
  }
}
```



Обратите внимание, двойной обратный слэш необходим для экранирования, если указывается значение в конфигурационном файле Web-клиента. При указании значения в Консоли управления Docsvision используется одиночный обратный слэш.



В конфигурации по умолчанию параметры отсутствуют.

> Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.
>
> Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.

Изменить логотип для штампа ЭП

По умолчанию в печатной версии подписанного файла, в штампе ЭП размещается логотип "Щит". Администратор может изменить изображение.



> В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.
>
> Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.



> *Модуль "Web-клиент" поддерживает следующие варианты настройки логотипа штампа ЭП:*
>
> - Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Штамп ЭП"
> - При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы изменить логотип для штампа ЭП из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" > "WebClient" > "SettingGroups" > "DefaultESImageGenerator"**.
3. Измените значение параметра `LogoImagePath`, указав путь к PNG файлу нового логотипа (относительно каталога `/lib/docsvision/webclient/` или абсолютный).

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "DefaultESImageGenerator": {
          "LogoImagePath": "Content/logo.png"
        }
      }
    }
  }
}
```

Файл логотипа должен быть доступен процессу сервера Web-клиента. Размер стандартного изображения: 127x131 пискелей. Изображение будет сжато в соответствии с фактическим размером штампа.

4. Сохраните изменения конфигурационного файла.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Предусмотрена возможность реализации собственной функции формирования штампа. Пример реализации см. в репозитории "ДоксВижн" на GitHub.

Подписание без скачивания файла

Активация настройки позволяет не скачивать подписываемый файл в браузер для последующего использования, вместо этого скачиваться будет только хэш подписываемого файла.



Модуль "Web-клиент" поддерживает следующие способы изменить режим скачивания подписи:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Электронная подпись""

- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Настройка по умолчанию активирована (установлена в значение `true`) для Web-клиент версии 6.1, сборка 909 и выше. Для работы настройки потребуется наличие библиотеки КриптоПро .NET с соответствующей лицензией на сервере Web-клиента, см. подробнее "Необходимое программное обеспечение". При отсутствии библиотеки работает прежнее поведение — будет скачиваться файл.

Библиотека КриптоПро .NET ПО ускоряет формирование подписи, особенно при подписании файлов большого размера. При необходимости администратор может отключить оптимизацию при помощи параметра `CalculateSignatureHashOnServer` в конфигурационном файле Web-клиента.



- Если алгоритм подписания не поддерживается, действует стандартная логика подписания со скачиванием файла в браузер.
- В данный момент подписание без скачивания файла не действует с облачной подписью.

Чтобы отключить поведение по-умолчанию:

1. Откройте конфигурационный файл, расположенный по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к разделу **"Docsvision" > "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `CalculateSignatureHashOnServer` с `true` на `false`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "CalculateSignatureHashOnServer": true ①
        }
      }
    }
  }
}
```

① Измените значение:

- `false` — будет скачиваться полностью подписываемый файл (устаревшее поведение, библиотека не требуется).
- `true` — скачивается только хэш подписываемого файла (значение по умолчанию, требуется библиотека КриптоПро .NET).

Изменить способ удаления карточек и папок

По умолчанию карточки из Web-клиента удаляются в корзину, папки удаляются временно, но в корзину не попадают. Администратор может включить режим удаления без возможности восстановления.



Восстановить удалённые карточки и папки можно только в Windows-клиенте.



Модуль "Web-клиент" поддерживает следующие варианты настройки удаления карточек и папок:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Интерфейс""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `DeletePermanently`:

   Удалённые карточки будут помещаться в корзину, папки будут удаляться временно

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "DeletePermanently": "false" ①
        }
      }
    }
```

```
  }
}
```

① Возможные значения:

- `True` — отключает использование корзины, папки и карточки удаляются безвозвратно.
- `False` — удалённые карточки помещаются в корзину, папки удаляются временно (значение по умолчанию).



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.

Настройки кэширования

Web-клиент

Настройки кэширования

Информация о настройках клиентского кэширования, настройках очистки данных пользовательской сессии, кэшировании разметок и прочих параметрах кэширования.

- Настройки клиентского кэширования
- Отключить кэширование разметок фильтров данных
- Больше подробностей в навигационном меню слева…

Настройки клиентского кэширования

Web-клиент по умолчанию кэширует некоторые данные на стороне клиента: настройки папок, сохранённые поисковые запросы, настройки разметок.

После изменения настроек папок или запросов, пользователь должен нажать на кнопку  на панели с деревом папок, чтобы обновить данные в кэше.

Администратор может отключить кэширование настроек у всех пользователей. При этом клиент будет каждый раз заново получать актуальные данные с сервера Web-клиента. Отключение кэширования отрицательно повлияет на скорость работы модуля.



Модуль "Web-клиент" поддерживает следующие варианты изменения настроек клиентского кэширования:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Кэширование"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы отключить кэширование настроек:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › "DirectoryCache"**.
3. Измените значение параметров:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "DirectoryCache": {
          "AdvancedLayouts": "False", ①
          "Folders": "True", ②
          "SearchCard": "True" ③
        }
      }
    }
  }
}
```

① `AdvancedLayouts` — отключает кэширование разметок.

② `Folders` — отключает кэширование настроек папок.

③ `SearchCard` — отключает кэширование сохранённых поисковых запросов.



Параметр `AdvancedLayouts` не используется, рекомендуется использовать актуальную настройку ***SharedDictionaryRefreshPeriod***, см. подробнее в разделе "Функциональность "Общие справочники"".

Параметр `Folders` является дублирующим для параметра `ShortCacheRefreshPeriod`, см. подробнее в разделе "Кэширование обращений к справочникам и карточкам".

4. Сохраните изменения конфигурационного файла.

Отключить кэширование разметок фильтров данных

По умолчанию Web-клиент использует серверный кэш для хранения разметок фильтров данных в представлении. Кэш обновляется при перезапуске сервера Web-клиент, а также автоматически при изменении настроек представления.

Администратору нужно отключить кэширование разметок фильтров, если в Web-клиенте планируется использовать две или более разметки представлений. Если кэширование не отключено, пользователи могут получать разметки фильтра, не соответствующие условиям выбора разметок, указанным в программе Конструктор Web-разметок.



Модуль "Web-клиент" поддерживает следующие способы отключить кэширование разметок фильтров данных:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Кэширование"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы отключить кэширование разметок из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"**.

3. Измените значение параметра `GridFilterCache` в `false`:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "GridFilterCache": "False" ①
        }
      }
    }
  }
}
```

① Возможные значения:

- `False` — кэширование разметок фильтров будет отключено.
- `True` — значение по умолчанию, кэширование включено.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.

Настройки кэширования разметок

Web-клиент использует серверный кэш для хранения подготовленных разметок карточек, основного меню, дашборда и др.

Для каждой локации разметок (см. Руководство пользователя программы Конструктор Web-разметок) используется собственный кэш, для которого может быть установлен требуемый режим кэширования:

Disabled	Кэширование отключено.
PerUser	Для каждого пользователя используется собственный кэш.

AllUsers	Используется общий кэш для всех пользователей.

Кэширование по умолчанию настроено следующим образом:

AllUsers	• Локация "Создание дочернего задания согласования". • Локация "Панель поиска по столбцам представления". • Локация "Диалог поисковых параметров". • Локация "Корень WebFrame".
PerUser	• Локация "Главная страница". • Локация "Основное меню". • Локация "Диалог профиля пользователя". • Локация "Групповые операции с карточками".
Disabled	• Остальные локации.

Администратор может изменить режим кэширования. Для выполнения следующих действий нужна программа *Docsvision Explorer* (входит в *Resource Kit*).

1. Откройте *Docsvision Explorer*.
2. Подключитесь к серверу и БД Docsvision, в которой выполняется изменение.
3. В разделе *Cards* выберите карточку *WebClientSettingsDirectory*.

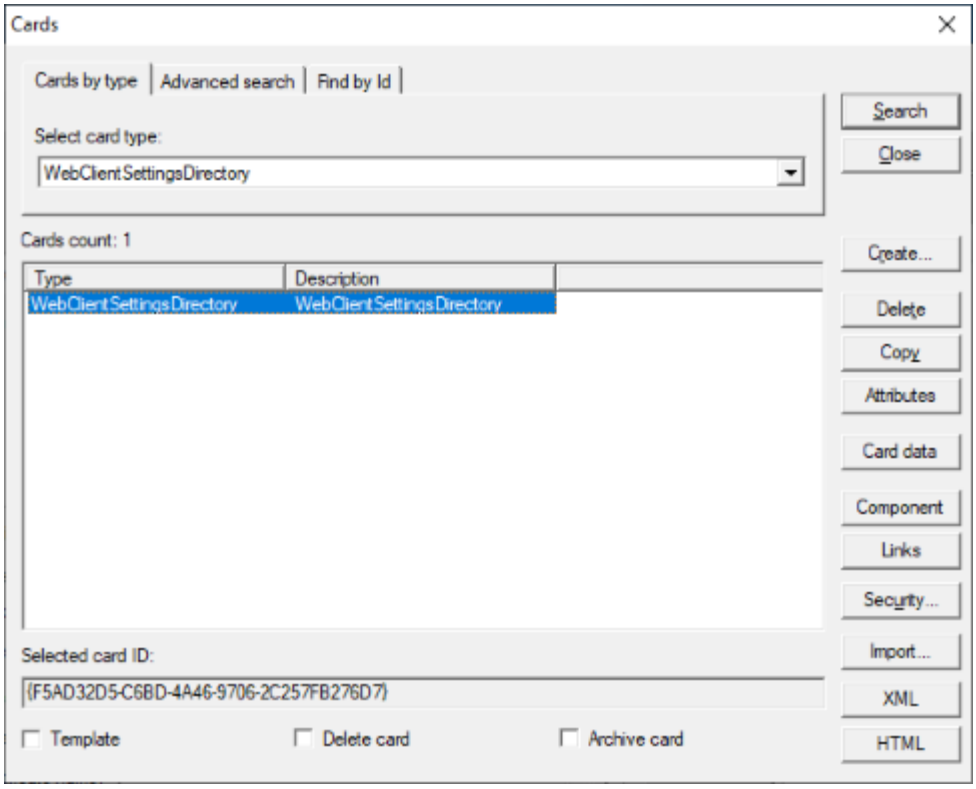


Рисунок 28. Раздел "Cards" в Docsvision Explorer

4. Нажмите кнопку **Card data** на панели команд. Будет открыт редактор данных карточки *WebClientSettingsDirectory*.
5. Перейдите к секции *Локации* карточки.

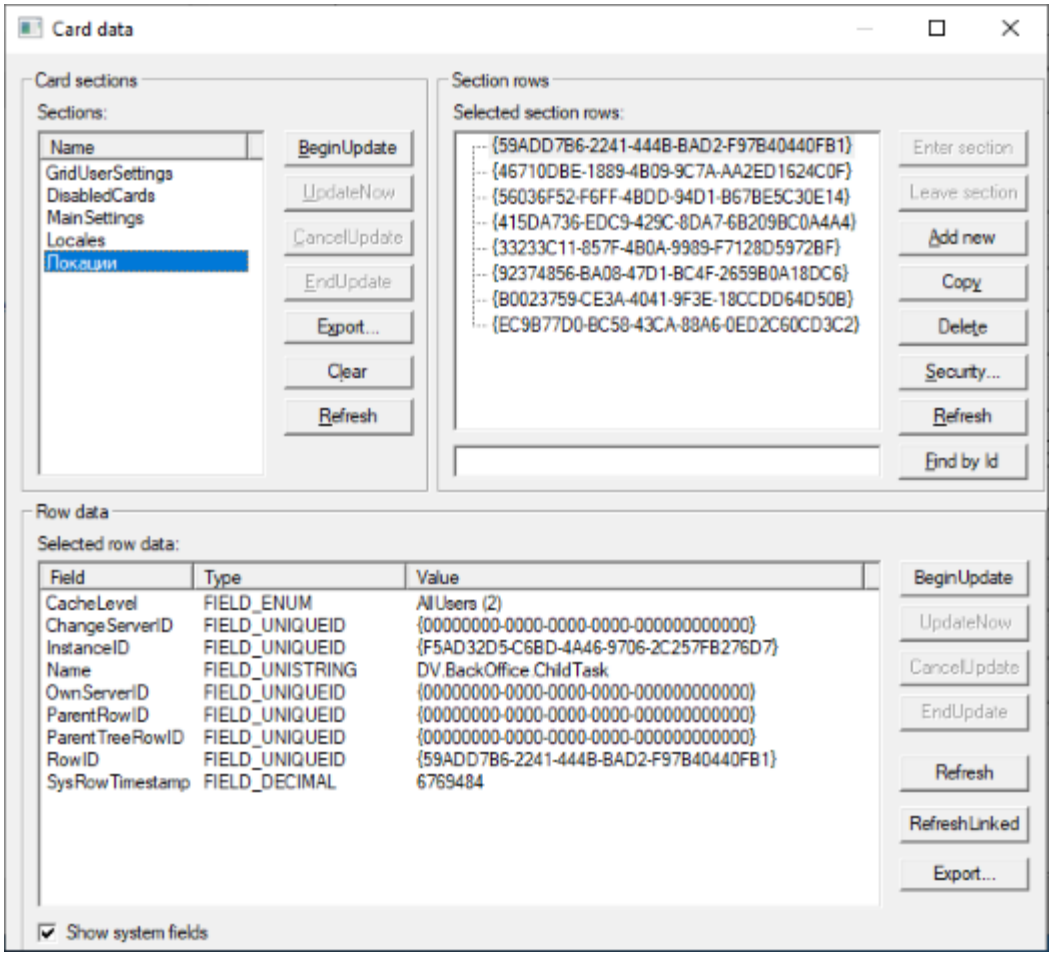


Рисунок 29. Данные секции "Локации" карточки "WebClientSettingsDirectory"

6. В *Section rows* выберите строку секции с настройками требуемой локации. Название локации указано в поле *Name* строки секции.
7. Измените значение поля *CacheLevel* (дважды щелкните по полю для открытия редактора) на требуемый режим кэширования:
   - ***0 — Disabled***.
   - ***1 — PerUser***.
   - ***2 — AllUsers***.
8. Закройте *Docsvision Explorer*.
9. Перезапустите сервер Web-клиента.

Изменить время обновления кэша настроек представлений

Web-клиент кэширует настройки представлений в серверном кэше и обновляет его с периодичностью в одну минуту.

На сервере Web-клиента, предназначенном для тестирования, период обновления может быть уменьшен, например, до одной секунды (для

своевременного получения изменений разметок на клиенте).



Модуль "Web-клиент" поддерживает следующие варианты изменения времени обновления кэша настроек представлений:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Кэширование"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы изменить период обновления кэша из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › System**.
3. Измените значение параметра `MinCheckDirectoryTimestampsInterval`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "MinCheckDirectoryTimestampsInterval": "1000" ①
        }
      }
    }
  }
}
```

① Время ожидания задаётся в миллисекундах.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.

Функциональность "Общие справочники"

Данная функциональность включает особый режим работы со справочниками, данные которых используются только в режиме чтения.

Функциональность по умолчанию активна для Web-клиента, начиная с версии 17.

Использование функциональности *Общие справочники* призвано сократить потребление памяти процессом сервера Docsvision.



Модуль "Web-клиент" поддерживает следующие варианты настройки функциональности "Общие справочники":

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Кэширование"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Изменить способ обновления справочников в режиме "только чтение"

Web-клиент использует справочники, данные которых только читаются и не изменяются, в режиме "только чтение". В данном режиме один справочник используется всеми сессиями и обновляется в соответствии со стратегией обновления, установленной в настройках Web-клиент.

Администратор может изменить режим обновления таких справочников в зависимости от требований к актуальности их данных. Увеличение частоты обновления ведёт к увеличению числа обращений с северу Docsvision, а уменьшение — к возможности получения устаревших данных.

Чтобы изменить способ обновления справочников из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › "System"**.
3. Измените значение параметра `SharedDictionaryRefreshMode`. Параметр определяет способ обновления справочников.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "SharedDictionaryRefreshMode": "0", ①
          "SharedDictionaryRefreshPeriod": "120" ②
        }
      }
    }
  }
}
```

① .Возможные значения:

- `0` — обновление выполняется с интервалом, указанным в параметре `SharedDictionaryRefreshPeriod`.
- `1` — если в одной из сессий данные справочника в режиме только чтение были изменены, обновление будет выполнено при следующем обращении.
- `2` — если в одной из сессий данные справочника в режиме только чтение были изменены, обновление будет выполнено сразу после изменения (значение по умолчанию).

② Задать период обновления справочника равный 2 минутам. Значение указывается в секундах, по умолчанию — 600 секунд.

4. Настройки обновления справочников влияют также на обновление разметок. **На период разработки разметок** рекомендуется установить меньший период обновления, а также уменьшить значения параметров `ShortCacheRefreshPeriod` и `LongCacheRefreshPeriod`, см. подробнее "Кэширование обращений к справочникам и карточкам".



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

5. Сохраните изменения конфигурационного файла.

Кэширование обращений к справочникам и карточкам

При обращении к справочникам и карточкам Web-клиент может использовать два режима обновления объектов:

- `ShortCacheRefreshPeriod` — короткий период обновления кэша.
- `LongCacheRefreshPeriod` — длительный период обновления кэша.

Короткий период обновления позволяет повысить производительность при обращении к справочникам и карточкам. Параметры также влияют на обновление данных в разметке профиля пользователя.



Модуль "Web-клиент" поддерживает следующие варианты настройки кэширования обращений к справочникам и карточкам:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Кэширование"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы настроить период обновления из конфигурационного файла, укажите значения в секундах:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "ShortCacheRefreshPeriod": "600", ①
          "LongCacheRefreshPeriod": "14400" ②
        }
      }
    }
  }
}
```

① Короткий период обновления кэша и данных в разметке профиля пользователя.

② Длительный период обновления кэша и данных в разметке профиля пользователя.

Настройка `ShortCacheRefreshPeriod` *используется для:*

- Карточки папок
- Справочника видов
- Конструктора справочников
- Справочника сотрудников
- Справочника контрагентов
- Справочника категорий

Настройка `LongCacheRefreshPeriod` *используется для:*

- Всего остального, включая справочник разметок



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Управление кэшем папок

Web-клиент кэширует настройки папки с сохранением числа непрочитанных карточек в папке. Если настройки папки были изменены, счётчики непрочитанных карточек можно обновить вручную.

Чтобы обновить счётчик непрочитанных, достаточно нажать на кнопку **Обновить** в дереве папок.

Автоматически счётчик обновляется раз в 10 минут или после перезапуска службы Сервера Web-клиента.

Такая настройка позволяет оптимизировать загрузку главного меню папок в Web-клиенте.



Вместо подключения большого числа папок точечно в меню через разметку или через пользовательские папки рекомендуется подключать одну корневую папку, в которой нужные папки являются дочерними.

Отключить счётчик непрочитанных карточек



Модуль "Web-клиент" поддерживает следующие варианты отключения счётчика непрочитанных карточек:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Интерфейс""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы настроить проверку непрочитанных карточек для группы пользователей в конфигурационном файле:

1. Создайте новую группу в справочнике сотрудников.
2. Скопируйте RowID группы в утилите DVExplorer.
3. Добавьте в конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows настройку `DisableUnreadCountersGroup`:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "DisableUnreadCountersGroup": "RowID" ①
        }
      }
    }
  }
}
```

① RowID группы, скопированный в утилите DVExplorer.

4. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они

> считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Настройка запроса на перенос срока исполнения заданий

В Web-клиенте версии 6.1, сборка 1528 и выше исполнитель задания может создать запрос на перенос срока исполнения после предварительной настройки.

Ниже настройка приведена на примере заданий **Задание УД › На исполнение** и **Задание УД › На ознакомление**.

Чтобы сотрудники могли работать с запросами на перенос срока исполнения:

1. Откройте *Конструктор состояний*. Выберите *Библиотеку карточек* "Базовые объекты" и *Вид карточки* **Задание УД › На исполнение** и **Задание УД › На ознакомление** на панели конструктора.
2. Сделайте доступной операцию редактирования *Запрос изменения срока* и, при необходимости, *Изменение срока исполнения* в требуемых состояниях.
3. Откройте *Конструктор ролей*. Выберите *Библиотеку карточек* "Базовые объекты" и *Вид карточки* **Задание УД › На исполнение** и **Задание УД › На ознакомление** на панели конструктора.
4. На вкладке *Роли* добавьте роли, для которых должна быть доступна возможность запросить перенос сроков исполнения. Также настройте условия, при которых будут доступна операция.
5. На вкладке *Матрица доступа* выполните настройки для:
   - Создания и отзыва запроса.
   - Принятия и отклонения запроса.
6. Добавьте на разметки заданий элемент управления "Запрос изменения срока исполнения".

Протоколирование событий

1. При остановке службы Сервера Web-клиента в файл журнала Web-клиента добавляется запись о завершении Web-клиента, например:

```
[2022-10-26
17:38:08.1840][Error][DocsVision.Platform.WebClient.Diagnostics.Trace.TraceError]
Application end: HostingEnvironment
```

2. При старте службы Сервера Web-клиента создаётся новый файл журнала Web-клиента.
3. К названию файла журнала добавляется ID процесса и дата запуска Web-клиента:

   ```
   WebClient${event-properties:item=Tenant} PID $\{processId} $\{shortdate}.log
   ```

   Например: `WebClient PID 4592 2022-10-26.log`

4. В созданный файл добавляется сообщение о старте Web-клиента, например:

   ```
   [2022-10-26
   17:11:56.8774][Error][DocsVision.Platform.WebClient.Diagnostics.Trace.TraceError]
   Application start
   ```

Протоколирование запросов сервера Web-клиента

Протоколирование запросов полезно для анализа входящих запросов сервера Web-клиента и ответов на них, заголовков и т.д.

Журнал запросов содержит следующую информацию:

- HTTP запросы и ответы.
- Дата и время запросов и ответов.
- Заголовки

Протоколирование запросов сервера Web-клиента настраивается в конфигурационном файле модуля по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows, за настройку отвечает параметр `W3CLogging`:

```
{
"W3CLogging": {
    "IsEnabled": true, ①
    "LogDirectory": "/var/log/docsvision/webclient/", ②
    "LoggingFields":
"Date,Time,ServerName,Method,UriStem,UriQuery,ProtocolStatus,TimeTaken,ProtocolVersion,Ho
st,UserAgent,Referer,ConnectionInfoFields", ③
    "FileSizeLimit": 10485760, ④
    "RetainedFileCountLimit": 4, ⑤
    "FileName": "webclient-w3c-" ⑥
```

```
}
}
```

① Включение/отключение протоколирования событий. Значение по умолчанию: `false`, в стандартной конфигурации остальные настройки отсутствуют.

② Путь к журналу.

③ Журналируемые поля. По умолчанию указываются запросы, ответы, заголовки, дата/время (записывается в формате UTC) и имя сервера. Со всеми допустимыми значениями можно ознакомиться в документации Microsoft.

④ Максимальный размер файла журнала, после которого будет создаваться новый. Значение в байтах, по умолчанию 10485760 байт (10 Мб). Цифра `0` означает отсутствие ограничения по размеру.

⑤ Количество сохраняемых файлов журнала. Допустимые значения от `1` до `10000`, по умолчанию используется `4`.

⑥ Префикс имени файла журнала.

Запросы будут записаны в файл `/var/log/docsvision/webclient/webclient-w3c-{YYYYMMDD.X}.txt` для Linux и `C:\ProgramData\docsvision\webclient\webclient-w3c-{YYYYMMDD.X}.txt` для Windows. Настройка альтернативного расширения поддерживается только при протоколировании с NLog, см. подробнее ниже.



Не удаляйте папку с журналом модуля, т.к. у службы сервера Web-клиента отсутствуют права на создание новой папки.

Протоколирование запросов с помощью NLog

Если требуется отображать в журналах локальное время или расширить список журналируемой информации, есть возможность подключения Nlog. При этом стандартное журналирование "W3CLogging" будет отключено (вне независимости от значения параметра **W3CLogging › IsEnabled**.

Протоколирование запросов с использованием NLog имеет ряд преимуществ перед журналированием W3CLogging. NLog предлагает больше параметров логирования (38 против 17),указывает в журнале местное время с точностью до миллисекунды, позволяет задать свои форматы для отображаемых данных и даёт гибкость в управлении файлами журнала.

Чтобы активировать протоколирование запросов с помощью NLog, измените параметры журналирования в параметре: **NLog › targets**:

```
{
"NLog": {
   "extensions": [
    { "assembly": "DocsVision.WebClient" } ①
   ],
   "targets": {
    "w3cFile": {
      "type": "File",
      "fileName": "${gdc:baseLogFolder}/docsvision/webclient/webclient-w3c-
${date:format=yyyyMMdd}.${processId}.log",
      "layout": {
        "type": "W3CLoggingLayout",
        "columns": [ ②
        { "column": "date" },
        { "column": "time", "layout": "${longdate}" }, ③
        { "column": "c-ip" },
        ]
      }
    }
   },
   "rules": [
    {
      "logger": "*",
      "minLevel": "Debug",
      "writeTo": "w3cFile"
    }
   ]
}
}
```

① Обязательный параметр, если он не указан, NLog не будет записывать обычные сообщения в журнал.

② Секция задаёт собственный набор колонок. Если список колонок не указан, в журнал будут включены только колонки по умолчанию `date`, `time`, `c-ip`, `s-ip`, `cs-username`, `s-computername`, `cs-method`, `cs-uri-stem`, `cs-uri-query`, `sc-statuscode`, `sc-bytes`, `cs-bytes`, `time-taken`, `cs-host`, `cs(User-Agent)`.

③ Формат представления данных колонки может быть изменён. Если формат не указан, будут использоваться значения по умолчанию. В `layout` можно использовать любые переменные NLog, подробнее см. в документации NLog.

▼ *Полный список колонок и допустимых значений для журналирования:*

- `date` — Дата окончания обработки запроса. Значение по умолчанию: `${date:format=yyyy-MM-dd}`.
- `time` — Время окончания обработки запроса. Значение по умолчанию: `${date:format=HH:mm:ss.fff}`.
- `webclient-version` — Версия Web-клиента. Например `6.1.725+a4c9b271d7`. Значение по умолчанию: `${webclient-version}`. Для протоколирования запросов сервера приложений аналогичная колонка называется `appserver-version`.
- `c-ip` — IP-адрес клиента. Значение по умолчанию: `${aspnet-request-ip}`.
- `cs(Auth-Type)` — Тип используемой аутентификации (например, Cookie). Значение по умолчанию: `${aspnet-user-authtype}`.
- `cs(Certificate)` — Сертификат клиента текущего запроса. Значение по умолчанию: `${aspnet-request-client-certificate}`.
- `cs(Content-Type)` — Тип контента: `text/html`, `multipart/form-data` и т.д.. Значение по умолчанию: `${aspnet-request-contenttype}`.
- `cs(Cookie)` — Cookie, подробнее о допустимых параметрах в документации NLog. Значение по умолчанию: `${aspnet-request-cookie}`.
- `cs(Referer)` — Значение поля `Referer` из заголовка запроса. Значение по умолчанию: `${aspnet-request-referrer}`.
- `cs(User-Agent)` — User agent. Значение по умолчанию: `${aspnet-request-useragent}`.
- `cs-bytes` — Получено байтов. Значение по умолчанию: `${aspnet-request-contentlength}`.
- `cs-connectionid` — Идентификатор соединения, см. в документации Microsoft. Значение по умолчанию: `${aspnet-request-connection-id}`.
- `cs-form` — Данные формы из запроса. Подробнее о параметрах в документации NLog. Значение по умолчанию: `${aspnet-request-form}`.
- `cs-headers` — Заголовок запроса. Если нужно взять только определенные заголовки, можно воспользоваться параметрами, подробнее см. в документации NLog. Значение по умолчанию: `${aspnet-request-headers}`.
- `cs-host` — Значение `Host` из заголовка запроса. Значение по умолчанию: `${aspnet-request-url:includeScheme=false:includeHost=true:includePath=false}`.
- `cs-fullhostname` — Полное имя хоста (с портом). Значение по умолчанию:

`${aspnet-request-host}`.

- `cs-isauthenticated` — Состояние аутентификации (значения `0` или `1`). Значение по умолчанию: `${aspnet-user-isAuthenticated}`.
- `cs-method` — Метод (`get`/`post` и т.п.). Значение по умолчанию: `${aspnet-request-method}`.
- `cs-mvc-action` — Вызываемое действие. Значение по умолчанию: `${aspnet-mvc-action}`.
- `cs-mvc-controller` — Вызываемый контроллер. Значение по умолчанию: `${aspnet-mvc-controller}`.
- `cs-posted-body` — Значение по умолчанию `${aspnet-request-posted-body}`.

  Значения:

  - `1`, если:
    - Это запрос HTTP/1.x с ненулевым значением Content-Length или заголовком "Transfer-Encoding: chunked".
    - Это запрос HTTP/2, который не установил флаг END_STREAM на исходном фрейме заголовка.
  - `0`, если:
    - Это запрос HTTP/1.x без заголовка Content-Length или "Transfer-Encoding: chunked", или Content-Length равен 0.
    - Это запрос HTTP/1.x с Connection: Upgrade (например, WebSockets). Для этих запросов не существует тела HTTP-запроса, и никакие данные не должны быть получены до тех пор, пока не будет выполнено обновление.
    - Это запрос HTTP/2, который устанавливает END_STREAM на начальном фрейме заголовка.
- `cs-tracking-consent` — Согласие на отслеживание (значения `0` или `1`). Значение по умолчанию: `${aspnet-request-tracking-consent}`.

  Возможные параметры:

  - `property`.

    Допустимые значения:

    - `CanTrack` — Указывает, было ли дано согласие, или если согласие не требуется (1 или 0);

    - `HasConsent` — было ли дано согласие;
    - `IsConsentNeeded` — требуется ли согласие для данного запроса.

  Пример вызова в собственном формате: `${aspnet-request-tracking-consent:property=CanTrack}`.

- `cs-uri-query` — Значение по умолчанию: `${aspnet-request-url:includeScheme=false:includeHost=false:includePath=false:includeQueryString=true}`. Запрос URI.
- `cs-uri-stem` — Ресурс URI. Значение по умолчанию: `${aspnet-request-url:includeScheme=false:includeHost=false}`.
- `cs-username` — Имя пользователя. Значение по умолчанию: `${aspnet-user-identity}`.
- `is-websocket` — Является ли запрос запросом на установку WebSocket (`0` или `1`). Значение по умолчанию: `${aspnet-request-is-web-socket}`.
- `s-basepath` — Значение `ContentRootPath`. Значение по умолчанию: `${aspnet-appbasepath}`.
- `s-computername` --Имя сервера. Значение по умолчанию: `${machinename}`.
- `s-environment` — Значение `ASPNETCORE_ENVIRONMENT`. Значение по умолчанию: `${aspnet-environment}`.
- `s-ip` — IP-адрес сервера. Значение по умолчанию: `${aspnet-request-local-ip}`.
- `s-port` — Порт. Значение по умолчанию: `${aspnet-request-local-port}`.
- `s-webrootpath` — `WebRootPath`. Значение по умолчанию: `${aspnet-webrootpath}`.
- `sc(Content-Type)` — Тип контента ответа. Значение по умолчанию: `${aspnet-response-contenttype}`.
- `sc-bytes` — Отправлено байт. Значение по умолчанию: `${aspnet-response-contentlength}`.
- `sc-response-has-started` — Были ли отправлены заголовки ответа, т.е. начался ли уже ответ или нет. Значения `0` или `1`. Значение по умолчанию: `${aspnet-response-has-started}`.
- `sc-status` — Код статуса ответа (`200`, `404` и т.д.). Значение по умолчанию: `${aspnet-response-statuscode}`.
- `time-taken` — Сколько времени заняла обработка запроса (в миллисекундах). Значение по умолчанию: `${aspnet-request-duration}`.

Настройка журналирования

События Web-клиента протоколируются с помощью компонента NLog (http://nlog-project.org).



Для каждой используемой БД в режиме мультитенантности ведётся свой журнал, например: `WebClient_DataBaseName_ГГГГ-ММ-ДД`.

Расширенные настройки журналирования осуществляются в конфигурационном файле Web-клиента.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Измените параметры журналирования в параметрах `targets` и `rules`.

```
{
 "NLog": {
     "throwConfigExceptions": false,
     "extensions": [
         { "assembly": "NLog.Extensions.Logging" },
         { "assembly": "NLog.Web.AspNetCore" }
     ],
     "targets": {
         "async": true,
         "logFile": {
             "type": "File",
             "fileName": "${basedir}/../Logs/WebClient${event-properties:item=Tenant}
PID ${processId} ${shortdate}.log", ①
             "layout": "[${longdate}][${level}][${callsite}] ${message}
${onexception:${newline}${exception:maxInnerExceptionLevel=10:format=shortType,message
,stacktrace:separator=*:innerExceptionSeparator=&#xD;&#xA;&#x9;}}"
         },
         "logConsole": {
```

```
            "type": "Console"
        }
    },
    "rules": [ ②
        {
            "logger": "Microsoft.*",
            "maxLevel": "Info",
            "final": true
        },
        {
            "logger": "*",
            "minLevel": "Error",
            "writeTo": "logFile" ③
        },
        {
            "logger": "*",
            "minLevel": "Trace",
            "writeTo": "logConsole" ④
        }
    ]
 }
}
```

① Измените значение параметра `fileName`, чтобы настроить путь для сохранения журнала работы:
В `fileName` можно использовать допустимые для NLog переменные, например: `\Logs\${level}\WebClient_${shortdate}.log`.

② Измените значение параметров `minLevel` и `maxLevel`, например, на `trace`, чтобы включить протоколирование всех типов событий.
Допустимые уровни протоколирования приведены на странице NLog в GitHub.
Для получения дополнительной информации о других настройках NLog обратитесь к документации по данной платформе.

③ Настройки журнала Web-клиента.

④ Настройки журнала Панель управления Web-клиентом.

3. Сохраните изменения конфигурационного файла.



Уровень журналирования может быть изменён методом `/Log/SetLogLevel`, если пользователь, который вызывает метод, состоит в группе **DocsVision Administrators**. Методом можно задать следующие уровни: ***Trace = 0***, ***Debug = 1***, ***Info = 2***,

> ***Warn = 3***, ***Error = 4*** (любое другое значение будет распознано как ***Error***). Передать уровень журналирования можно, например, при помощи GET-параметра `minLevel`:
>
> ```
> Log/SetLogLevel?minLevel=0
> ```
>
> После перезапуска службы Сервера Web-клиента уровень журналирования будет возвращён в исходное состояние в соответствии с конфигурационным файлом.

Расширенное журналирование

Web-клиент поддерживает расширенное журналирование параметров производительности. Сюда входит точное логирование времени выполнения запросов Web-клиента, время выполнения запросов к службе Сервера Docsvision с привязкой к запросу Web-клиента.

Для настройки доступно три фильтра журнала: По пользователю, По конкретному адресу, По времени запроса.

После включения диагностического логирования можно делать запросы. Если в параметре будет слеш (`\` или `/`), его необходимо переводить в URL код. Иными словами, заменять `example\usr01` на `example%5Cusr01`.

Для отдельного запроса Web-клиента можно посмотреть:

- Какие запросы к службе Сервера Docsvision выполнялись.
- Время, потраченное на выполнение каждого из них.
- Время, проведённое запросами на стороне службы Сервера Docsvision и на стороне Web-клиента.
- Параметры запросов.

Эта информация позволяет:

- Устанавливать причины аномально быстрого или медленного выполнения запроса Web-клиента.
- Выявлять ошибки и неоптимальные места в работе Web-клиента и Object Manager.
- Обнаруживать блокировки Web-клиента и Object Manager.
- Выявлять проблемные запросы на сервере.

Чтобы включить диагностическое логирование, нужно в конфигурационном файле Web-клиента для параметра `EnableDiagnosticsLogger` в разделе **"Docsvision" › "WebClient" > "SettingGroups" › "System"** указать значение `1`:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "EnableDiagnosticsLogger": "1" ①
        }
      }
    }
  }
}
```

① Включить расширенное журналирование.

Настроить параметры журналирования можно в конфигурационном файле следующим образом:

```
{
  "targets": {
    "async": true,
    "logFile": {
        "type": "File",
        "fileName": "${gdc:baseLogFolder}/docsvision/webclient/webclient.log",
        "layout": "[${longdate}][${level}][${callsite}] ${message}
${onexception:${newline}${exception:maxInnerExceptionLevel=10:format=shortType,message,st
acktrace:separator=*:innerExceptionSeparator=&#xD;&#xA;&#x9;}}"
    },
    "logConsole": {
        "type": "Console"
    },
    "diagnosticsLogWebClient": {
        "type": "File",
        "fileName":
"${gdc:baseLogFolder}/docsvision/webclient/DiagnosticsLogWebClient${event-
properties:item=Tenant} PID ${processId} ${shortdate}.tsv" ①
    },
    "diagnosticLogStorageServer": {
        "type": "File",
        "fileName":
"${gdc:baseLogFolder}/docsvision/webclient/DiagnosticLogStorageServer${event-
```

```
properties:item=Tenant} PID ${processId} ${shortdate}.tsv" ②
    }
  },
  "rules": [ ③
    {
      "logger": "DiagnosticsLogWebClient",
      "minLevel": "Trace",
      "writeTo": "diagnosticsLogWebClient"
    },
    {
      "logger": "DiagnosticLogStorageServer",
      "minLevel": "Trace",
      "writeTo": "diagnosticLogStorageServer"
    },
  ]
}
```

① Адрес хранения журнала Web-клиента. В параметре важно добавить в имя файла `${processId}`, например, `WebCDiagnostics PID${processId}.log`, а также указать `layout="${message}"`.

② Адрес хранения журнала службы Сервера Docsvision. В параметре важно добавить в имя файла `${processId}`, например, `WebCDiagnostics PID${processId}.log`, а также указать `layout="${message}"`.

③ Обязательные разделы для работы расширенного журналирования.

Журнал ведётся в формате `.tsv` и в дальнейшем может быть проанализирован через приложение для работы с журналами, например Log Parser Studio. Расширенный журнал по умолчанию сохраняется в стандартный каталог журналов Web-клиента, но при помощи параметров конфигурации NLog можно записать в иной каталог.

Чтобы включить ведение расширенного журнала с требуемым фильтром, нужно выполнить в браузере запрос. Примеры запросов приведены далее.

Примеры запросов

1. Запрос включения журнала для всех пользователей:

   ```
   http://адрес-сервера-Web-клиента:ПОРТ/DiagnosticsLog/AddLogUser?user=*
   ```

   Журналируются все запросы `example\usr01`:

```
http://адрес-сервера-Web-клиента:ПОРТ/DiagnosticsLog/AddLogUser?user=example%5Cusr01
```

2. Запрос включения журнала по адресу:

```
http://адрес-сервера-Web-
клиента:ПОРТ/DiagnosticsLog/AddLogAddress?address=Grid%2FGetGridDataEx
```

Журналируются все запросы, в имени которых содержится подстрока `Grid/GetGridDataEx`. При необходимости можно журналировать все запросы конкретного контроллера:

```
http://адрес-сервера-Web-клиента:ПОРТ/DiagnosticsLog/AddLogAddress?address=Grid
http://адрес-сервера-Web-клиента:ПОРТ/DiagnosticsLog/AddLogAddress?address=*
```

3. Запрос включения журнала по времени выполнения запросов:

```
http://адрес-сервера-Web-
клиента:ПОРТ/DiagnosticsLog/AddLogRequestTime?milliseconds=500
```

4. Запрос установки таймера журналирования с указанием времени работы 10 минут:

```
http://адрес-сервера-Web-клиента:ПОРТ/DiagnosticsLog/SetIntervalTimer?minutes=10
```

▼ *Принцип работы запроса :*

Вызов запроса `SetIntervalTimer` не включает журналирование. Для журналирования должен быть указан конкретный фильтр (`AddLogUser`, `AddLogAddress` или `AddLogRequestTime`), который будет вызван до или после запроса `SetIntervalTimer`. Запрос `SetIntervalTimer` может быть выполнен в любой момент до или после установки фильтров. При его выполнении таймер перезапускается. Т.е. если нужно продлить время журналирования, достаточно выполнить этот запрос ещё раз.

Указание таймера в параметре `SetIntervalTimer` не является обязательным, в таком случае журналирование всех запросов по умолчанию будет длиться 1 час. Даже после сброса с помощью `Reset` запроса. Пользовательское значение таймера сбрасывается на значение по умолчанию после перезапуска службы Сервера Web-клиента (т.е. значение времени таймера

хранится в кэше Web-клиента).

Время работы таймера отсчитывается с момента последнего запроса к API журналирования.

5. Фильтры можно комбинировать. Ниже приведён пример включения фильтра для пользователя `example\usr01`, адреса `Grid/GetGridDataEx`, времени выполнения запросов `50` мс:

```
http://адрес-сервера-Web-клиента:ПОРТ/DiagnosticsLog/AddLogUser?user=example%5Cusr01
http://адрес-сервера-Web-
клиента:ПОРТ/DiagnosticsLog/AddLogAddress?address=Grid%2FGetGridDataEx
http://адрес-сервера-Web-клиента:ПОРТ/DiagnosticsLog/AddLogRequestTime?milliseconds=50
```

6. Чтобы отключить ведение расширенного журнала, нужно выполнить запрос в браузере:

```
http://адрес-сервера-Web-клиента:ПОРТ/DiagnosticsLog/Reset
```

Включить журнал событий навигации

Администратор может включить ведение журнала для записи информации о событиях и последующего анализа ошибок. В лог браузера записывается информация о специальных навигационных обработчиках, передаваемым им параметрах и результатах их работы.



Модуль "Web-клиент" поддерживает следующие способы включить журнал событий навигации:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Журналирование""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы включить протоколирование из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции menu:"Docsvision"["WebClient" > "SettingGroups" > "WebClient".

3. Измените значение параметра `EnableRouterLogging` на ***True***:

Включение протоколирования событий навигации

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "EnableRouterLogging": "True" ①
        }
      }
    }
  }
}
```

① Возможные значения:

- `True` — протоколирование событий навигации включено.
- `False` — значение по умолчанию, протоколирование событий навигации отключено.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.

Включить счетчики производительности



Функциональность счётчиков производительности — встроенный механизм ОС Windows и не работает в ОС Linux.

Система отслеживания работоспособности счётчиков



Функциональность счётчиков производительности — встроенный механизм ОС Windows и не работает в ОС Linux.

Управление сессиями пользователей

Web-клиент

Управление сессиями пользователей

Информация об управлении временем активности сессий пользователей.

- Автоматически закрывать неактивные сессии пользователей
- Изменить время ожидания пользовательской сессии
- Управление временем жизни пользовательских cookie

Автоматически закрывать неактивные сессии пользователей

Web-клиент автоматически переводит в офлайн-режим сессии (подключение к серверу Docsvision) неактивных пользователей. При этом у пользователей появится предупреждение о переводе сессии в офлайн. Впоследствии неактивные сессии будут закрыты. Таймер закрытия сессии запускается одновременно с переходом в офлайн-режим.



После перевода пользовательской сессии в офлайн-режим или её закрытия будет остановлено автоматическое обновление счетчиков непрочитанных карточек и содержимого папок.



Модуль "Web-клиент" поддерживает следующие варианты настройки закрытия неактивных сессий пользователей:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Пользовательские сессии"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Чтобы изменить стандартные настройки сессии пользователей из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции menu:"Docsvision"["WebClient" > "SettingGroups" > "WebClient".

3. Измените значение параметра:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "SessionSuspendTimeout": 10, ①
          "SessionCloseTimeout": 300, ②
          "SessionInactiveMessageHide": true ③
        }
      }
    }
  }
}
```

① `SessionSuspendTimeout` — время неактивности пользователя в минутах, через которое сессия будет переведена в офлайн-режим. Значение по умолчанию — 10 минут.

② `SessionCloseTimeout` — время неактивности пользователя в минутах, через которое сессия будет закрыта. Значение по умолчанию — 300 минут.

③ `SessionInactiveMessageHide` — отключает показ сообщения *"Подключение остановлено в связи с отсутствием активности"*.

- Значение ***true*** Сообщение не отображается, сессия восстанавливается при нажатии курсором мыши в любой области окна Web-клиент.
- Значение ***false*** Сообщение отображается, сессия восстанавливается при нажатии курсором мыши на сообщение или в любой области окна Web-клиент (значение по умолчанию).

4. Чтобы отключить перевод неактивной сессии в офлайн-режим, установите ***SessionSuspendTimeout*** в значение ***0***.

   Сервер Web-клиента проверяет неактивность пользователей через промежуток времени ***SessionSuspendTimeout * 0.5***, поэтому перевод пользовательской сессии в офлайн-режим может быть выполнен в промежутке времени до ***SessionSuspendTimeout * 1.5***.

5. Чтобы отключить закрытие неактивных сессий, установите ***SessionCloseTimeout*** в значение ***0***.



В конфигурации по умолчанию параметры отсутствуют.

> Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.
>
> Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

6. Сохраните изменения конфигурационного файла.

Изменить время ожидания пользовательской сессии



> *Модуль "Web-клиент" поддерживает следующие способы изменить время ожидания пользовательской сессии:*
>
> - Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Подключение к серверу приложений""
> - При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Администратор может изменить максимальное время ожидания ответа (таймаут) от сервера Docsvision. Время ожидания устанавливается в секундах.

Чтобы изменить таймаут пользовательской сессии из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › System**.
3. Измените значение параметра `ConnectionTimeout`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "ConnectionTimeout": 3600000 ①
        }
      }
    }
```

```
  }
}
```

① Время ожидания пользовательской сессии равное 3600000, указывается в секундах.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.

Управление временем жизни пользовательских cookie



Модуль "Web-клиент" поддерживает следующие варианты управления временем жизни пользовательских cookie:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Пользовательские cookie"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "AuthCookieLifetime": 120, ①
          "MobileAuthCookieLifetime": 120 ②
        }
      }
    }
  }
}
```

① `AuthCookieLifetime` — период действия пользовательской сессии.

② `MobileAuthCookieLifetime` — период действия пользовательской сессии на

мобильных устройствах.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Принцип действия настроек:

1. Настройка `AuthCookieLifetime` устанавливает период действия пользовательской сессии. Значение по умолчанию: `120` часов (5 дней).
2. Настройка `MobileAuthCookieLifetime` отдельно задаёт период действия пользовательской сессии на мобильных устройствах. Значение по умолчанию: `120` часов (5 дней).

Настройки можно комбинировать. Например, задать для `AuthCookieLifetime` значение `240` часов, чтобы все пользователи получали cookie на 10 дней. Для настройки `MobileAuthCookieLifetime` можно указать период в `720` часов (30 дней). Когда пользователь откроет Web-клиент по прошествии трёх дней с предыдущей авторизации, cookie автоматически будут продлены.

Настройки DVWebTool и DVSupService

DVWebTool и DVSupService необходимы для работы пользователя в Web-клиенте. DVWebTool и DVSupService предоставляют пользователю возможность открывать файлы в офисном редакторе с правами **текущего пользователя**. Если пользователь использует ОС Windows, будет использоваться DVWebTool, если сотрудник работает в Linux, будет использоваться DVSupService.

Web-клиент

Настройки DVWebTool и DVSupService

Информация о настройках DVWebTool и DVSupService, включая настройку порта подключения, автоматического запуска и закрытия, а также сканирования документов.

- Изменить порт подключения к DVWebTool
- Настройки запуска инструментов
- Настройки сканирования с помощью DVSupService
- Больше подробностей в навигационном меню слева…

Установка DVWebTool и DVSupService

DVWebTool и DVSupService предоставляются пользователям возможность открывать приложенные к карточкам файлы. DVWebTool устанавливается на компьютеры пользователей в ОС Windows из инсталлятора, собранного при помощи WebToolBuilder (входит в комплект утилит `Docsvision web-client toolkit.msi`), DVSupService устанавливается в ОС Linux стандартными инструментами системы.

Данный раздел описывает:

- Установку DVWebTool с помощью WebToolBuilder.
- Установку DVSupService системными средствами.

Установка DVWebTool при помощи WebToolBuilder

Программа *WebToolBuilder* предназначается для сборки *DVWebTool*, минуя Панель управления Web-клиентом. *WebToolBuilder* входит в комплект утилит `Docsvision web-client toolkit.msi`.

WebToolBuilder необходимо запускать для сборки DVWebTool в следующих случаях:

1. Когда устанавливается версия Web-клиента без программы Панель управления Web-клиентом
2. Когда изменились файлы DVWebTool или вышла новая версия
3. Когда изменился адрес или порт сервера Web-клиента
4. Когда изменились внутренние расширения DVWebTool

Программа "WebToolBuilder" устанавливается:

- Из инсталлятора комплекта утилит `Docsvision web-client toolkit.msi`. Docsvision Web-клиент Toolkit содержит WebToolBuilder и Конструктор Web-разметок, пакет утилит устанавливается в ОС Windows. Программа WebToolBuilder может быть запущена с минимальным пользовательским интерфейсом или как консольное приложение.

- Файлы утилиты будут расположены по пути: `C:\Program Files (x86)\Docsvision\WebClientToolkit\WebToolBuilder\`
- Файлы DVWebTool будут расположены по пути (в манифесте прописан `localhost`): `C:\Program Files (x86)\Docsvision\WebClientToolkit\WebToolBuilder\DVWebTool\`

Новая версия WebToolBuilder выпускается при каждом исправлении в Web-клиенте. Для распространения WebToolBuilder могут использоваться групповые политики.

Запуск в режиме окна

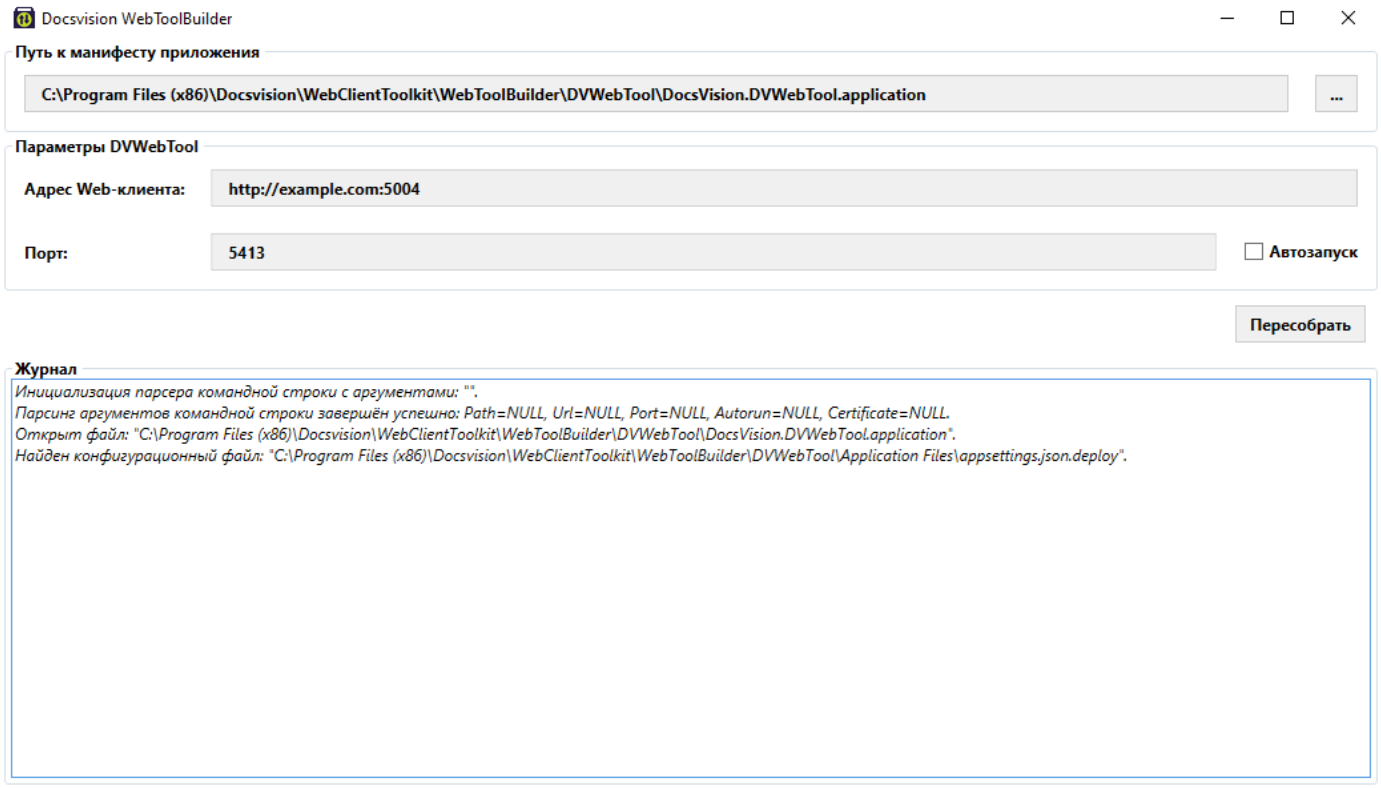


Рисунок 30. Окно программы "WebToolBuilder"

В окне приложения отображается следующий набор элементов:

- Путь к манифесту приложения — поле для выбора файла манифеста `DocsVision.DVWebTool.application`
- `...` — кнопка для выбора манифеста приложения из файловой системы
- Адрес Web-клиента — полный адрес сервера Web-клиента
- Порт — порт для подключения к DVWebTool
- Флаг `Автозапуск` — отвечает за автоматический запуск DVWebTool при загрузке системы

- Кнопка **Пересобрать** неактивна, пока не выбран файл манифеста и не заполнены поля *Адрес Web-клиента* и *Порт*
- Поле *Журнал*, в котором записываются действия при работе с программой

Запуск в консольном режиме

1. Перейдите в каталог с WebToolBuilder (по умолчанию `C:\Program Files (x86)\Docsvision\WebClientToolkit\WebToolBuilder\WebToolBuilder.exe`) и запустите утилиту командой с указанием пути к файлу манифеста:

   ```
   ./WebToolBuilder C:\Program Files (x86)\Docsvision\WebClientToolkit\WebToolBuilder\DVWebTool\DocsVision.DVWebTool.application
   ```

2. Чтобы получить справку по работе с утилитой в консольном режиме, введите `-h` при нахождении в папке утилиты:

   Доступные опции:

   - `-u`, `--url` — URL развёртывания манифеста приложения на сервере (обязательная опция)
   - `-p`, `--port` — Порт для работы DVWebTool (записывается в конфигурационный файл DVWebTool)
   - `-a`, `--autorun` — Флаг, отвечающий за автоматический запуск DVWebTool при загрузке системы
   - `-c`, `--certificate` — Путь к файлу сертификата для подписания манифеста приложения и манифеста развёртывания
   - `--version` — Узнать версию приложения
   - `-?`, `-h`, `--help` — Показать справку
   - `--whatIf` — Отображает список параметров, которые будут использоваться при пересборке DVWebTool

Сборка DVWebTool через WebToolBuilder

Программа *DVWebTool* необходима для выполнения сканирования документов и открытия файлов в офисном редакторе в ОС Microsoft Windows.



Программа *DVWebTool* работает только в ОС Microsoft Windows. Для работы требуется установить ПО .NET Desktop Runtime (x86/x64) версии 8.0 и ASP.NET Core Runtime версии 8.0. ПО можно загрузить

с сайта Microsoft.

Чтобы установить "DVWebTool":

1. Чтобы установить *DVWebTool*, необходимо создать образ утилиты. Для этого запустите *WebToolBuilder* на Windows.
2. В поле *Путь к манифесту приложения* выберите файл манифеста `C:\Program Files (x86)\Docsvision\WebClientToolkit\WebToolBuilder\DVWebTool\DocsVision.DVWebTool.application`. После выбора файла автоматически будут заполнены другие поля.
   - В поле *Адрес Web-клиента* указывается адрес сервера Web-клиента
   - В поле *Порт* указывается порт для подключения к DVWebTool (по умолчанию `5413`)
3. При необходимости установите флаг `Автозапуск`, чтобы *DVWebTool* запускалась автоматически при загрузке системы.
4. Нажмите кнопку **Пересобрать**, чтобы открыть каталог.

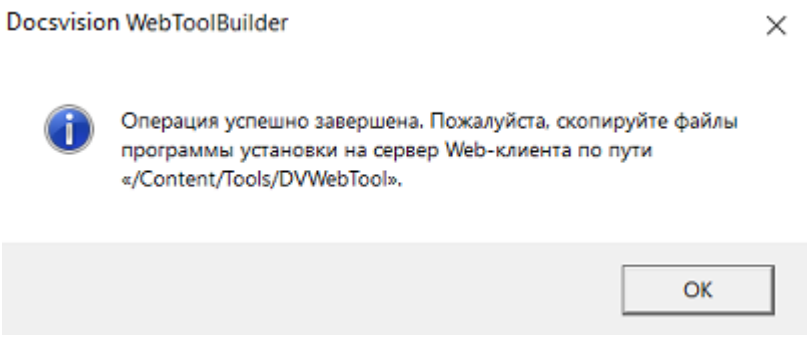


Рисунок 31. Успешное завершение сборки

После завершения сборки появится окно об успешном завершении операции. При нажатии на кнопку **ОК** будет открыта папка в файловой системе с собранными файлами DVWebTool.

5. Скопируйте открытую папку *DVWebTool* на сервер Docsvision по пути `путь-установки-Web-клиента/Content/Tools` (если папка отсутствует, создайте её самостоятельно).
6. Перезапустите службу Web-клиента.
7. *DVWebTool* будет установлена при первом открытии файла из карточки в ОС Windows. Программу также можно установить вручную из **Профиля пользователя › Установки › Установить DVWebTool**.

   После завершения установки программа *DVWebTool* будет запущена

автоматически, а в области уведомлений Windows появится значок программы ().

Программа устанавливается только для текущего пользователя. Если на компьютере работает несколько пользователей (несколько профилей Windows), программу нужно установить каждому пользователю.

Установка и запуск службы DVSupService

Служба *DVSupService* необходима для открытия файлов в офисном редакторе в ОС Linux.



Служба *DVSupService* работает только в ОС Linux. Для работы требуется установить ASP.NET Core Runtime версии 8.0. ПО можно загрузить с сайта Microsoft.

Чтобы установить "DVSupService":

1. Выполните следующую команду:

   Astra Linux

   ```
   $ sudo apt install docsvision-dvsupservice
   ```

   РЕД ОС

   ```
   $ sudo dnf install docsvision-dvsupservice
   ```

   Служба будет установлена в каталог `/lib/docsvision/dvsupservice` от имени суперпользователя. Если на компьютере работает несколько пользователей, службу нужно установить один раз.

   В конфигурационном файле службы по адресу `/usr/lib/docsvision/dvsupservice/appsettings.json` для ОС Linux можно указать настройку `WebClientAddress` (по умолчанию null). Если для этой настройки указать адрес сервера (например `http://docsvision.server.com:5004`), служба будет принимать запросы на открытие только с указанного адреса. При необходимости можно указать несколько адресов.

2. Запустите службу следующей командой:

   ```
   $ sudo systemctl start dvsupservice
   ```

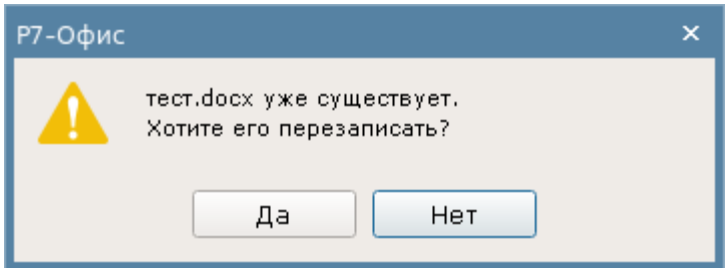


Рисунок 32. Диалог сохранения в Р7-Офис

При работе *DVSupService* с Р7-Офис может появляться диалог сохранения. В таком случае необходимо, не изменяя директорию, нажать **Сохранить** › **Да**.

3. Посмотреть журнал работы можно по адресу `/var/log/docsvision/dvsupservice/$ДД-ММ-ГГГГ_DvSupService.log`.
   Путь по умолчанию может быть изменён в конфигурационном файле службы:

```
{
"fileName": "/var/log/docsvision/dvsupservice/${shortdate}_DvSupService.log"
}
```

4. Чтобы просмотреть информацию о службе, выполните команду:

Astra Linux

```
$ apt list docsvision-dvsupservice
```

РЕД ОС

```
$ dnf list docsvision-dvsupservice
```

5. Обновить службу можно, выполнив следующие команды:

Astra Linux

```
sudo apt-get update
sudo apt-get install docsvision-dvsupservice
```

РЕД ОС

```
sudo dnf makecache
```

```
sudo dnf update docsvision-dvsupservice
```

6. Удалить службу можно с помощью следующей команды:

Astra Linux

```
$ sudo apt-get purge --autoremove docsvision-dvsupservice
```

РЕД ОС

```
$ sudo dnf remove docsvision-dvsupservice
```

Сменить пользователя DVSupService

По умолчанию служба *DVSupService* запускается от имени пользователя ROOT.

1. Чтобы запустить службу от имени другого пользователя, необходимо выполнить следующую команду:

```
sudo systemctl edit dvsupservice
```

2. Записать следующее содержимое в появившемся окне:

```
[Service]
User=user
Group=user
```

В примере выше `user` — пользователь, от имени которого необходимо запускать *DVSupService*.

Изменить порт подключения к DVWebTool



Программа *DVWebTool* устанавливается на клиентскую машину только в ОС Microsoft Windows. Для ОС Linux используется *DVSupService*, см. подробнее Установка и запуск службы DVSupService.



Модуль "Web-клиент" поддерживает следующие способы изменения порта подключения к программе DVWebTool:

- При помощи WebToolBuilder без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше, см. подробнее в разделе "Работа с программой WebToolBuilder"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Веб-браузер пользователя подключается к программе *DVWebTool* по адресу `http://localhost:5413`. Администратор может изменить стандартный порт подключения.

Чтобы изменить стандартный порт подключения из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › System**.
3. Измените значение параметра ***WebServicesPort***, указав новый номер порта подключения к программе *DVWebTool*.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "WebServicesPort": 5413 ①
        }
      }
    }
  }
}
```

① Порт подключения к утилите DVWebTool, значение в диапазоне от `1` до `65535`.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над

настройками в Консоли управления Docsvision.

4. Сохраните изменения конфигурационного файла.
5. Перезапустите программу *DVWebTool*.

Настройки запуска инструментов

Настройки запуска программы DVWebTool

Поведение по умолчанию: программа запускается при загрузке системы и не закрывается после выполнения действия. Данное поведение можно изменить.



Модуль "Web-клиент" поддерживает следующие способы настройки запуска программы DVWebTool:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Закрывать DVWebTool после выполнения действия"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

Программа *DVWebTool* необходима для выполнения сканирования документов и открытия файлов в офисном редакторе в ОС Microsoft Windows.



Программа *DVWebTool* работает только в ОС Microsoft Windows. Для работы требуется установить ПО .NET Desktop Runtime (x86/x64) версии 8.0 и ASP.NET Core Runtime версии 8.0. ПО можно загрузить с сайта Microsoft.

В конфигурационном файле Web-клиента поведение DVWebTool при запуске настраивается:

- `DvWebToolAppShutDown` — закрывать DVWebTool после выполнения действия. Значение по умолчанию: `false` — не закрывать.
- `DvWebToolStartUpLaunch` — отвечает за автоматический запуск DVWebTool. Значение по умолчанию: `true` — утилита запускается автоматически при загрузке системы.

Программа WebToolBuilder позволяет собрать DVWebTool со следующими параметрами:

- *Адрес сервера Web-клиента*: укажите адрес сервера Web-клиента.
- *Порт DVWebTool*: укажите порт DVWebTool. По умолчанию `5413`.

- **Автозапуск**: параметр имеет приоритет над параметром DvWebToolStartUpLaunch в конфигурационном файле Web-клиента. Если в конфигурационном файле Web-клиента параметр true, но утилита собрана со снятым флагом **Автозапуск**, автозапуск не сработает.

Чтобы настройки вступили в действие после изменения, необходимо повторно собрать инсталлятор DVWebTool по инструкции "Установка DVWebTool и DVSupService".

Запуск программы DVWebTool



Сборка *DVWebTool* делегирована утилите WebToolBuilder, см. подробнее в разделе "Установка DVWebTool и DVSupService".

1. DVWebTool автоматически запускается при входе пользователя в ОС, если при сборке был установлен флаг **Автозапуск**. Программу можно запустить вручную из меню **Пуск** › **Программы** › **Docsvision** › **DVWebTool**.

   В одной ОС утилита *DVWebTool* может быть запущена одновременно только от одного пользователя.

2. Для получения информации о программе, выберите пункт *About…* в контекстном меню программы. Будет открыто окно *About*.

   Окно About содержит:

   - Название и версию программы.
   - Список установленных расширений.

3. Для просмотра журнала работы программы, выберите пункт *View log* в контекстном меню программы.
   Журнал будет открыт в стандартном текстовом редакторе.

4. После установки новой версии Web-клиента программу *DVWebTool* потребуется заново собрать по инструкции в разделе "Установка DVWebTool и DVSupService".

Настройки запуска DVSupService

1. Чтобы запустить службу DVSupService, выполните следующую команду:

   ```
   $ sudo systemctl start dvsupservice
   ```

Служба *DVSupService* необходима для открытия файлов в офисном редакторе в ОС Linux. По умолчанию служба *DVSupService* запускается от имени пользователя ROOT.

2. Чтобы запустить службу от имени другого пользователя, необходимо выполнить следующую команду:

```
sudo systemctl edit dvsupservice
```

3. Записать следующее содержимое в появившемся окне:

```
[Service]
User=user
Group=user
```

В примере выше `user` — пользователь, от имени которого необходимо запускать *DVSupService*.

Настроить автоматический запуск и иное поведение можно при помощи системных средств, см. подробнее руководство по работе с программным пакетом `systemd` в ОС Linux.

Настройки сканирования с помощью DVSupService

По умолчанию сканирование с помощью ЭУ "Сканирование" работает только в ОС Windows с помощью программы DVWebtool, чтобы иметь возможность сканировать файлы в ОС Linux, необходимо выполнить дополнительные действия по настройке DVSupService для работы со сканерами с поддержкой SANE. Установка DVSupService описана в разделе "Установка и запуск службы DVSupService".

Выбор провайдера сканирования

Параметр `ScanProviders` позволяет задать провайдеры сканирования. Настройка выполняется в конфигурационном файле утилиты по адресу `/usr/lib/docsvision/dvsupservice/appsettings.json` для ОС Linux `C:\Program Files\Docsvision\DvSupService\appsettings.json` для ОС Windows.

```
{
"ScanProviders": [
    "Naps2Provider" ①
```

```
  ],
}
```

① Провайдер сканирования. Возможные значения: `"Naps2Provider"`, `"ScanImageProvider"`.

Если подключено несколько провайдеров, рядом с именем сканера в интерфейсе сканирования карточки будет отображаться название провайдера, чтобы избежать дублирования пунктов.

Настройка сканирования



Модуль "Web-клиент" поддерживает следующие способы настройки порта подключения к DVSupService:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Группа настроек "Интерфейс""
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

В конфигурационном файле утилиты по адресу `/usr/lib/docsvision/dvsupservice/appsettings.json` для ОС Linux `C:\Program Files\Docsvision\DvSupService\appsettings.json` для ОС Windows параметр `ServicesPort` позволяет задать порт подключения к DVSupService для сканирования:

```
{
"ServicesPort": 5413, ①
}
```

① Порт для подключения к DVSupService. Значение должно быть в диапазоне от `1` до `65535`, по умолчанию `5413`.

На машине с Linux и подключенным сканером необходимо установить компоненты SANE, выполнив в командной строке:

Astra Linux

```
$ sudo apt install sane
```

РЕД ОС

```
$ sudo dnf install sane
```

Если планируется использовать только провайдер `Naps2Provider`, дополнительные компоненты устанавливать не требуется. Если планируется использовать провайдер `ScanImageProvider`, необходимо установить пакет `sane-utils` командой:

Astra Linux

```
$ sudo apt install sane sane-utils
```

РЕД ОС

```
$ sudo dnf install sane sane-utils
```

Изменить транспорт для работы с сервером Docsvision

Возможность изменить адрес подключения к серверу Docsvision и, соответственно, транспорт в Web-клиенте упразднена, начиная с версии Web-клиент 17. Даже если в конфигурационном файле транспорт был изменён на COM, будет использоваться REST.

В зависимости от установленной версии модуля *Платформа* используемая точка подключения может не соответствовать указанной в адресе подключения.

- Web-клиент в связке с модулем *Платформа* версии 5.5.1 и ниже по умолчанию использует SOAP.
- Web-клиент в связке с модулем *Платформа* версии 5.5.2 и выше по умолчанию использует REST.

Включить аутентификацию Kerberos

Прозрачная аутентификация Kerberos позволяет выполнять вход от имени УЗ текущего пользователя (если он зарегистрирован в справочнике сотрудников). Необходимые настройки описаны в руководстве по настройке системы, см. раздел "Настройка прозрачной аутентификации".

При удалённом подключении (по RDP) к машине на ОС Windows, введённой в домен ALD Pro, сквозная аутентификация в сервисах Docsvision может не работать, т.к. *официально* ALD Pro не поддерживает ввод машин Windows в домен.

Настройка "Р7-Офис. Сервер документов"

Модуль "Web-клиент" поддерживает следующие способы настройки Р7-Офис:

- Из интерфейса Консоли управления Docsvision без необходимости работы с форматом `.json`, начиная с версии 6.1 сборки 1265 и выше: "Настройка "Р7-Офис" в Консоли управления Docsvision"
- При помощи настройки в конфигурационном файле модуля, см. подробнее ниже

- Р7-Офис. Сервер документов должен быть установлен на машину с сервером Web-клиента.
  - Если версия Web-клиента **выше** версии 6.1, сборка 735 и выше, Р7-Офис можно устанавливать отдельно от сервера Web-клиента.
  - Если версия Web-клиента **ниже** версии 6.1, сборка 735 и выше, Р7-Офис необходимо устанавливать на сервер Web-клиента. Если Р7 будет установлен отдельно от серверных компонентов Docsvision, функции консолидации и объединения версий не будут работать. Также могут возникать проблемы с редактированием файла. Аналогичное поведение будет наблюдаться, если в конфигурационном файле указать адрес Р7 офиса, установленного на другой машине Linux.
- Для установки Р7-Офис в ОС Astra Linux требуется минимальная версия Astra Linux 1.7.3.

"Р7-Офис. Сервер документов" или "OnlyOffice" используется для предварительного просмотра файлов в карточках и для консолидации версий документов при согласовании. Данный раздел описывает установку и настройку Р7-Офис. Сервер документов, а также подготовку Web-клиента для использования функций предварительного просмотра и консолидации.

Предварительный просмотр при помощи "Р7-Офис. Сервер документов" или OnlyOffice позволяет отображать предпросмотр не печатной версии документа, а непосредственно файла. Например, для документа Excel появляется возможность скопировать данные из файла с сохранением структуры и форматирования. Возможность предварительного просмотра сохраняется, но функциональность ограничена набором возможностей Р7, например, новый компонент не поддерживает функцию поворота на 90 градусов.



Р7-Офис сервер документов может быть установлен и настроен самостоятельно по инструкции ниже или можно сразу подключиться к активному серверу Р7. Если планируется подключаться к активному серверу, перейдите сразу к указанию адреса Р7 Сервера документов в конфигурационном файле Web-клиента.

Содержимое файла в области предпросмотра иногда может отображаться с некорректным форматированием. Например, отсутствующие на сервере Р7 шрифты могут незначительно отличаться от исходного документа, некоторые сложные элементы могут выглядеть иначе.

Установка Р7-Офис. Сервер документов

Чтобы настроить работу Р7-Офис для включения предварительного просмотра и работы консолидации:

1. Убедитесь, что в Windows-клиенте указан *Адрес сервера Web-клиента*, см. подробнее в разделе "Системные настройки модуля Windows-клиент".

2. Установите "Р7-Офис. Сервер документов" согласно официальной инструкции на сайте Р7-Офис. Потребуется лицензия на Р7-Офис. В случае установки лицензии в ручном режиме, дополнительно может потребоваться назначить права на файл `license.lic`:

   ```
   chmod 644 /var/www/r7-office/Data/license.lic
   ```

3. Если все действия выполнены правильно, при открытии в браузере адреса `http://localhost:port` появится сообщение "Сервер документов запущен".

4. Создайте файл конфигурации для Р7-Офис `/etc/nginx/includes/ds-docsvision.conf`.

5. Выполните команду, чтобы новый конфигурационный файл прочитался:

```
systemctl reload nginx
```

6. В конфигурационный файл добавьте следующие строки:

```
location ~* ^(\/upload.*)(\/.*) {
  alias /tmp$1$2;
  add_header Content-Disposition "attachment; filename*=UTF-8''$arg_filename";
}
```

7. В файле `/etc/r7-office/documentserver/local.json` задайте для параметров `inbox`, `outbox` и `browser` значение `false`.

   Затем перезапустите службу Сервера Docsvision.

   Если Р7-Офис и Web-клиент находятся на одной машине и для связи между собой используют один частный IP-адрес, для корректного отображения предварительного просмотра необходимо изменить значение следующих параметров в файле `/etc/r7-office/documentserver/default.json`:

   ```
   {
     "request-filtering-agent" : {
       "allowPrivateIPAddress": false,
       "allowMetaIPAddress": false
     },
   }
   ```

   По умолчанию значение параметров установлено в `false`, необходимо изменить значение на `true`.

   После изменения параметров необходимо перезапустить службы командами:

   ```
   $ sudo service nginx restart && sudo supervisorctl restart all
   ```

8. В целях повышения безопасности настоятельно рекомендуется перевести сервер документов на использование HTTPS. Если переводить сервер документов на HTTPS не планируется, перейдите к пункту Конфигурация

Web-клиента. Для этого на сервере с установленным Р7 необходимо установить самоподписанный сертификат.

9. Создать сертификат можно при помощи утилиты OpenSSL. Запустите Win64 OpenSSL Command Prompt от имени администратора, перейдите в директорию, в которой располагается файл `r7-cert.pfx`
10. Выполните команду:

```
openssl req -x509 -newkey rsa:2048 -keyout test.key -nodes -out test.crt -subj
"/CN=r7.domain.com" ①
```

① Вместо `r7.domain.com` укажите имя сервера с установленным Р7.

11. Установите сертификат в системе на сервере Web-клиент, поместив его в папку `/usr/local/share/ca-certificates/`, после чего необходимо будет выполнить команду:

```
sudo update-ca-certificates -v -f.
```

12. После установки сертификата переключите Р7 на использование HTTPS с созданным сертификатом согласно https официальной инструкции на сайте Р7.
13. Настройте подключение к Web-клиенту по HTTPS по инструкции "Подключение к Docsvision по HTTPS".

Конфигурация Web-клиента

1. Откройте конфигурационный файл Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows в любом текстовом редакторе. Укажите адрес сервера документов Р7 и определите режим предварительного просмотра:



Для корректной работы предварительного просмотра убедитесь, что URL сервера документов Р7-Офис/OnlyOffice доступен с клиентских рабочих мест.

```
{
  "Docsvision": {
    "WebClient": {
```

```
      "SettingGroups": {
        "WebClient": {
          "ServerR7ConnectionAddress": "https://dvserver.preview.com", ①
          "ServerR7UploadDirectory": "/tmp/upload", ②
          "FilePreviewMode": "2", ③
        }
      }
    }
  }
}
```

① `ServerR7ConnectionAddress` — URL сервера документов Р7-Офис/OnlyOffice. Настройка обязательна для заполнения. Если URL сервера документов Р7 не указан, остальные настройки будет проигнорированы.

② `ServerR7UploadDirectory` — каталог загрузки консолидируемых файлов.

③ `FilePreviewMode` — определяет, какой компонент используется для предпросмотра. Значения: `0` — предпросмотр отключен, `2` — предпросмотр с помощью Р7.

2. Откройте в браузере адрес Р7-Офис сервер документов, например, `https://dvserver.preview.com`.
3. Откройте карточку с приложенным файлом. Предпросмотр должен отображаться.

Авторизация с помощью токена

Работа с Р7-Офис сервер документов для консолидации и предварительного просмотра поддерживает авторизацию с помощью токена, что является более безопасным вариантом стандартного режима работы.

Чтобы настроить авторизацию с помощью токена:

1. В конфигурационном файле Р7-Офис по адресу `/etc/r7-office/documentserver/local.json` установите параметры `inbox`, `outbox` и `browser` в значение `true` для использования токенов.
2. В том же конфигурационном файле `local.json` укажите секретное слово для параметров `inbox`, `outbox`, `session`, используя цифры и буквы латинского алфавита, фрагмент конфигурационного файла см. ниже.

```
{
  "services": {
    "CoAuthoring": {
```

```
      "sql": {
        "type": "postgres",
        "dbHost": "localhost",
        "dbPort": "5432",
        "dbName": "r7office",
        "dbUser": "r7office",
        "dbPass": "r7office"
      },
      "redis": {
        "host": "localhost"
      },
      "token": {
        "enable": {
          "request": {
            "inbox": true,
            "outbox": true
          },
          "browser": true
        },
        "inbox": {
          "header": "Authorization"
        },
        "outbox": {
          "header": "Authorization"
        }
      },
      "secret": {
        "inbox": {
          "string": "verysecretstring" ①
        },
        "outbox": {
          "string": "verysecretstring" ②
        },
        "session": {
          "string": "verysecretstring" ③
        }
      }
    }
  }
}
```

① Секретное слово для параметра `inbox`.

② Секретное слово для параметра `outbox`.

③ Секретное слово для параметра `session`.

3. Перезапустите сервер документов Р7.
4. В конфигурационном файле Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows в разделе **`"Docsvision" > "WebClient" > "SettingGroups" > "WebClient"`** укажите параметр `ServerR7SecretToken` со значением секретного слова.
5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
6. Если предыдущие шаги настройки выполнены корректно, Р7-Офис сервер документов должен работать без ошибок. Если при предварительном просмотре или консолидации возникают ошибки с упоминанием токена в тексте ошибки, настройка выполнена некорректно, проверьте правильность выполнения описанных выше шагов.

Предпросмотр больших файлов

Предпросмотр больших файлов может вызывать в окне предпросмотра сообщения о невозможности загрузить ресурсы. В таких случаях приложенный в карточку большой файл (100 Мб и больше) может не открыться.

Поведение корректируется настройками в конфигурационном файле Р7 Сервер документов. В `default.json` укажите следующие настройки:

Фрагмент конфигурационного файла:

```
"FileConverter": {
    "converter": {
        "maxDownloadBytes": 1048576000, ①
        "downloadTimeout": {
            "connectionAndInactivity": "60m", ②
            "wholeCycle": "60m" ③
        },
    }
}
```

① `maxDownloadBytes` — определяет максимальный размер запрашиваемого файла в байтах. Значение по умолчанию — `104857600`.

② `connectionAndInactivity` — определяет время ожидания отправки сервером заголовков ответа перед прерыванием запроса и таймаут для сокета после таймаута неактивности, значение по умолчанию — `2` минуты.

③ `wholeCycle` — определяет таймаут всего цикла запроса, значение по умолчанию

— 2 минуты.

После указания описанных настроек предпросмотр больших файлов будет открываться без ошибок.

Изменить форматы для редактирования

По умолчанию Web-клиент предлагает возможность открывать в стороннем ПО файлы следующих форматов: `.doc`, `.docx`, `.docm`, `.xls`, `.xlsx`, `.xlsm`, `.ppt`, `.pptx`, `.odt` и `.ods`. Данная функциональность доступна только, если установлено соответствующее ПО, позволяющее открывать файлы на редактирование, а также если пользователь установил и запустил *DVWebTool* или *DVSupService*.

При необходимости администратор может расширить список поддерживаемых форматов, которые можно будет открывать на редактирование при условии, если необходимое ПО установлено на компьютере пользователя.



С добавленными форматами функциональность открытия на редактирование может работать некорректно.

Если нужно предоставить возможность открывать файлы определённых форматов в карточки, выполните следующую настройку:

1. Откройте конфигурационный файл Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. В параметре **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient" › "AcceptableFileExtensionsForEditing"** измените список доступных файлов. Новые расширения файлов нужно добавлять через символ `|` в виде регулярного выражения:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "AcceptableFileExtensionsForEditing": "(
\\b)(doc|docx|docm|xls|xlsx|xlsm|ppt|pptx|odt|ods|xodt|xods|xodp)$" ①
        }
      }
    }
  }
```

```
}
```

① Перечислите поддерживаемые форматы в значении параметра.



Обратите внимание, двойной обратный слэш необходим для экранирования, если указывается значение в конфигурационном файле Web-клиента. При указании значения в Консоли управления Docsvision используется одиночный обратный слэш.

3. Если параметр `"AcceptableFileExtensionsForEditing"` не указан, по умолчанию на редактирование открываются только файлы форматов `.doc`, `.docx`, `.docm`, `.xls`, `.xlsx`, `.xlsm`, `.ppt`, `.pptx`, `.odt` и `.ods`.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Особенности настройки кластера Web-клиент

С целью повышения отказоустойчивости и распределения нагрузки, администратор может объединить несколько экземпляров модуля Web-клиент в кластер.

Обязательные условия для работы кластера

- Каждый экземпляр Web-клиента должен иметь уникальный адрес подключения.
- Балансировщик нагрузки должен передавать заголовок X-Forwarded-Proto.

Подготовка узлов кластера

Чтобы подготовить узлы кластера Web-клиента к работе:

1. Установите и настройте модуль Web-клиент на первом компьютере кластера Linux.
2. Установите и настройте Web-клиент на последующих узлах кластера Linux.

3. Настройте балансировщик нагрузки.

Сгенерируйте отпечаток

1. Если Web-клиент использует HTTPS, создайте сертификат с помощью OpenSSL. Пример использования утилиты приведён в разделе администрирования модуля Консоль управления Docsvision: "Создание доверенного сертификата для подключения к БД".
2. Установите и настройте Web-клиент на втором компьютере кластера (далее — **УЗЕЛ 2**).



Экземпляр Web-клиента, установленный на втором и последующих узлах кластера, должен быть настроен на работу с теми же сервером и базой данных Docsvision, что и УЗЕЛ 1.

3. В конфигурационном файле `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows на УЗЛЕ 1 укажите путь к полученному сертификату в элементе `"Https":` (`"Http":`, если используется незащищённое соединение). Фрагмент конфигурационного файла приведён ниже:

```
{
"Https": {
  "Url": "https://*:5005",
  "Certificate": {
    "Path": "/home/user/certificate.pfx", ①
    "Password": "password" ②
    }
  }
}
```

① Путь к сертификату

② Пароль от сертификата



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сгенерируйте сертификат на втором узле.
5. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows на УЗЛЕ 2 и укажите путь к полученному сертификату.
6. Повторите шаги инструкции для третьего и последующих узлов кластера Web-клиента.

Настройка nginx

При использовании прокси-сервиса nginx, установленного на отдельной машине от Web-клиента необходимо в корень конфигурационного файла Web-клиент добавить настройку `ForwardedHeaders/KnownProxies`, разрешающую принимать от прокси-сервиса заголовки. Если Web-клиент и прокси установлены на одном ПК, ничего добавлять не нужно.

```
{
  "ForwardedHeaders":
  {
    "KnownProxies": {
      "proxy1": "{123.456.789}" ①
    }
  }
}
```

① Укажите IP-адрес сервера nginx. Вместо `proxy1` можно использовать любое уникальное название. В значении параметра указывается только IP-адрес. При необходимости можно указывать несколько прокси-сервисов.

На стороне прокси-сервиса должна быть настроена передача заголовков: `X-Forwarded-Proto`, `X-Forwarded-Host`, `X-Forwarded-Port` (если порт не соответствует схеме), `X-Forwarded-For`.

Пример для nginx:

```
proxy_set_header X-Forwarded-Proto $scheme;
proxy_set_header X-Forwarded-Host $host;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Port $server_port;
```

Если за прокси-сервисом скрыто несколько экземпляров Web-клиента, нужно установить настройку

`Docsvision/WebClient/SettingGroups/WebClient/ServerR7JwtAccessToWebClientEnable` в значение `true`. Если настройка отсутствует, добавьте её самостоятельно.

Пример конфигурации:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "ServerR7JwtAccessToWebClientEnable": true
        }
      }
    }
  }
}
```

См. также **Настройка утилиты DVWebTool для работы в кластере**.

Настройка утилиты DVWebTool для работы в кластере

Руководство по настройке утилиты DVWebTool для работы в кластере недействительно для текущей версии модуля.

Особенности работы сессии в Web-клиенте

Начиная с Web-клиент 17 был отключен стандартный механизм сессий ASP.NET. Благодаря такой реализации работы сессии повышается быстродействие Web-клиента и долгие запросы не блокируют работу пользователя.

Если в решении использовался метод `HttpContext.Current.Session`, возникнет ошибка `Session = null`. В случае появления ошибки можно изменить значение параметра `mode` в конфигурационном файле Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows следующим образом:

```
{
  "system.web": {
    "sessionState": {
      "mode": "InProc"
    }
  }
}
```

В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

Действия после изменения учётной записи пользователя

После изменения учётной записи пользователя в *Справочнике сотрудников* необходимо перезапустить службу Сервера Web-клиента, в противном случае аутентификация пользователя в Web-клиенте завершится с ошибкой *Карточка с указанным идентификатором не существует*.

Экспериментальные функции

Отдельные функции Web-клиента, представленные в версии, носят экспериментальный характер. Экспериментальные функции предоставляют новые возможности или улучшения, но их разработка ещё не завершена, а использование может привести к побочным эффектам.

Использование экспериментальных функций может привести к снижению производительности, ошибкам, отсутствию ожидаемых функций и т. п.

Экспериментальные функции могут быть включены в основную функциональность Web-клиента в следующих версиях, но на данном этапе разработки не могут быть рекомендованы для всех пользователей.

Экспериментальные функции по умолчанию отключены.

Включить экспериментальную функцию

Чтобы включить экспериментальные функции:

1. Откройте Панель управления Web-клиентом и перейдите в раздел настроек "Экспериментальные функции".
2. Установите флаг Включено в строке включаемой функции.
3. Нажмите кнопку ... в поле Выбрать и выберите роль сотрудников, которым предоставляется доступ к экспериментальной функции.



Рекомендуется создать для этого отдельную роль в Справочнике сотрудников. Если роль не выбрана, данная экспериментальная функция не предоставляется.

Если функциональность не может быть ограничена определённой группой пользователей, кнопка Выбрать будет недоступна. В этом случае экспериментальная функция включается для всех пользователей Web-клиента.

4. Закройте Панель управления Web-клиентом.
5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Для корректной работы функции может потребоваться выполнить дополнительные настройки. Детальнее ознакомьтесь с описанием интересующей функции, чтобы узнать подробности.

Отключить экспериментальную функцию

Чтобы отключить экспериментальную функцию:

1. Напротив функции снимите флаг `Включено`.
2. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Параметры конфигурационного файла Web-клиента

Конфигурация модуля может выполняться при помощи конфигурационного файла Web-клиента, расположенного по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows, однако следует помнить, что настройки конфигурационного файла не сохраняются при обновлении версии Web-клиента.



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над

настройками в Консоли управления Docsvision.



Рекомендуемый способ конфигурации Web-клиента — при помощи Консоли управления Docsvision, дополнительные сведения приведены в разделах соответствующих параметров по ссылкам.

Настройки в конфигурационном файле не сохраняются при обновлении версии Web-клиента.

Общие настройки

Параметры меню создания карточек

Чтобы изменить настройки меню создания карточек из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к элементу **`"Docsvision" › "WebClient" › CardTypes`**.
3. Измените параметр `CardTypeId`:



Web-клиент поддерживает работу только с карточками, созданными на основе *Базовых объектов*.

```
{
  "Docsvision": {
    "WebClient": {
      "CardTypes": [
        {
          "CardTypeId": "b9f7bfd7-7429-455e-a3f1-94ffb569c794", ①
          "CssClass": "document", ②
          "SaveInFolder": "true" ③
        },
        {
          "CardTypeId": "c7b36f33-cdd4-4da9-8444-600fe14111e4",
          "CssClass": "task",
          "SaveInFolder": "true"
        }
      ]
    }
  }
}
```

① "CardTypeId": "GUID-идентификатор типа карточки".
Чтобы изменить порядок кнопок создания типов карточек, измените порядок следования GUID-идентификаторов (`CardTypeId`) в `CardTypes`.

② Чтобы изменить цвет строки типа карточки в меню создания карточек, добавьте в элемент `CardType` параметр `CssClass` с названием класса CSS в значении.

③ Чтобы новые карточки выбранного типа сохранялись в текущей открытой папке, добавьте в элемент `CardType` параметр `SaveInFolder` со значением `true`. Если `SaveInFolder` имеет значение `false` (или отсутствует), ярлыки карточек не будут добавляться в текущую папку.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Интерфейс"".
- Конфигурационный файл — "Настройка меню создания карточек".

Отключить создание типа карточек

Чтобы скрыть тип карточки из меню создания через конфигурационный файл:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к элементу **"Docsvision" › "WebClient" › CardTypes**.

   В элементе `CardTypes` перечислены типы карточек, с которыми работает Web-клиент:

```
{
  "Docsvision": {
    "WebClient": {
      "CardTypes": [
        {
          "CardTypeId": "b9f7bfd7-7429-455e-a3f1-94ffb569c794", ①
          "CssClass": "document",
          "SaveInFolder": "true"
        },
        {
          "CardTypeId": "c7b36f33-cdd4-4da9-8444-600fe14111e4", ②
          "CssClass": "task",
          "SaveInFolder": "true"
        },
        {
```

```
          "CardTypeId": "c7b36f33-cdd4-4da9-8444-600fe14111e4", ③
          "CssClass": "task",
          "SaveInFolder": "true",
          "DisableCreateFlags": "0" ④
        }
      ]
    }
  }
}
```

① Документ.

② Задание.

③ Группа заданий.

④ Создайте или измените параметр `DisableCreateFlags` в элементе нужного типа карточек.
Значение параметра `DisableCreateFlags` является суммой цифр, каждая из которых определяет один из вариантов скрытия типа карточки:

- `0` — не скрывать.
- `1` — скрывать тип из списка типов при создании карточки при нажатии кнопки **+ Создать** на панели управления.
- `2` — скрывать тип из списка типов при создании карточки с добавлением ссылки из карточек *Документ* и *Задание*, а также при создании карточки в качестве отчёта к *Заданию*.
- `4` — скрывать тип из списка типов при создании связанного задания карточки *Документ*.
- `8` — скрывать тип из списка типов при создании подчиненного задания карточки *Задание*.

Подробнее см.:

- Консоль управления Docsvision — "Внешний вид".
- Конфигурационный файл — "Отключить создание карточек определённого типа".

Удаление карточек и папок в корзину или навсегда

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.

2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `DeletePermanently`:

   Удалённые карточки будут помещаться в корзину, папки будут удаляться временно

   ```
   {
     "Docsvision": {
       "WebClient": {
         "SettingGroups": {
           "WebClient": {
             "DeletePermanently": "false" ①
           }
         }
       }
     }
   }
   ```

   ① Возможные значения:

   - `True` — отключает использование корзины, папки и карточки удаляются безвозвратно.
   - `False` — удалённые карточки помещаются в корзину, папки удаляются временно (значение по умолчанию).

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Интерфейс"".
- Конфигурационный файл — "Изменить способ удаления карточек и папок".

Разрешённые для добавления файлы

Чтобы изменить список разрешённых для добавления файлов из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `AcceptableFileExtensions`. Новые расширения файлов нужно добавлять через символ `|` в виде регулярного выражения.

   Список разрешённых файлов и добавление `.dwg` в допустимые расширения:

   ```
   {
   ```

```
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "AcceptableFileExtensions": "(
\\b)(rtf|dot|doc|dotx|ods|docx|odt|csv|xlt|xls|xltx|xlsx|pdf|jpg|jpeg|gif|png|bmp|tiff
|tif|emf|wmf|xslt|xsl|xml|htm|html|txt|log|cs|asmx|ascx|aspx|js|cpp|h|css|idl|ini|java
|rc|rc2|shtm|shtml|sql|zip|rar|7z|vsd|vsdx|msg|one|pptx|ppt|avi|wav|docm|xltm|dotm|xls
m|p7s|sign|sig|sgn|dwg)$"
        }
      }
    }
  }
}
```



Обратите внимание, двойной обратный слэш необходим для экранирования, если указывается значение в конфигурационном файле Web-клиента. При указании значения в Консоли управления Docsvision используется одиночный обратный слэш.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Работа с файлами"".
- Конфигурационный файл — "Изменить список разрешённых для добавления файлов".

Максимальный размер прикладываемых файлов

1. Чтобы изменить ограничение для максимального размера единичного файла, в конфигурационном файле `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows перейдите к параметру **"Docsvision" › "WebClient" › View › MaxUploadFileItemSize"**.

```
{
  "Docsvision": {
    "SettingGroups": {
      "WebClient": {
        "View": {
          "MaxUploadFileItemSize": "0" ①
        },
      },
    },
  },
```

```
}
```

① Укажите новое ограничение для максимального размера единичного файла в байтах, не превышающее `2147483647`.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Работа с файлами"".
- Конфигурационный файл — "Ограничить размер прикладываемых файлов".

Максимальный размер файлов для предпросмотра

Чтобы изменить ограничение для максимального размера файла для предпросмотра из конфигурационного файла:

1. Откройте конифгурационный файл Web-клиента: `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. В категории настроек **"Docsvision" › "WebClient" > "SettingGroups" › System** измените параметр `MaxPreviewFileSize`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "MaxPreviewFileSize": "0" ①
        }
      }
    }
  }
}
```

① Укажите значение в байтах.
Увеличение значения повысит нагрузку на сервер и время открытия файла на предпросмотр.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Работа с файлами"".
- Конфигурационный файл — "Ограничить максимальный размер файла для предпросмотра".

Открытие файлов на редактирование

Если нужно предоставить возможность открывать файлы определённых форматов в карточки, выполните следующую настройку:

1. Откройте конфигурационный файл Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. В параметре **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient" › "AcceptableFileExtensionsForEditing"** измените список доступных файлов. Новые расширения файлов нужно добавлять через символ `|` в виде регулярного выражения:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "AcceptableFileExtensionsForEditing": "(
\\b)(doc|docx|docm|xls|xlsx|xlsm|ppt|pptx|odt|ods|xodt|xods|xodp)$" ①
        }
      }
    }
  }
}
```

① Перечислите поддерживаемые форматы в значении параметра.



Обратите внимание, двойной обратный слэш необходим для экранирования, если указывается значение в конфигурационном файле Web-клиента. При указании значения в Консоли управления Docsvision используется одиночный обратный слэш.

3. Если параметр `"AcceptableFileExtensionsForEditing"` не указан, по умолчанию на редактирование открываются только файлы форматов `.doc`, `.docx`, `.docm`, `.xls`, `.xlsx`, `.xlsm`, `.ppt`, `.pptx`, `.odt` и `.ods`.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Работа с файлами"".
- Конфигурационный файл — "Изменить форматы для редактирования".

Настройки внешнего вида

Стандартный порядок кнопок "ОК/Отмена"

Порядок кнопок **ОК**/**Сохранить** и **Отмена**, отображаемых в диалоговых окнах и разметках Web-клиента, может быть настроен. Пользователь может настроить схему расположения в меню *Профиль*. Администратор может установить порядок расположения по умолчанию, который будет действовать до выбора нового значения пользователем.

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" > "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `DefaultButtonPosition`:

   Включение режима "Отмена/ОК"

   ```
   {
     "Docsvision": {
       "WebClient": {
         "SettingGroups": {
           "WebClient": {
             "DefaultButtonPosition": "Auto" ①
           }
         }
       }
     }
   }
   ```

   ① Возможные значения:

   - `Auto` — значение по умолчанию — порядок отображения кнопок определяется типом ОС пользователя.
     Для ОС MacOS: **Отмена**/**ОК**.
     Для ОС Windows, Linux, Android и др. — **ОК**/**Отмена**
   - `OkCancel` — **ОК**/**Отмена**
   - `CancelOk` — **Отмена**/**ОК**

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Интерфейс"".

- Конфигурационный файл — "Порядок кнопок "ОК/Отмена" по умолчанию".

Закрепление основного меню

Чтобы изменить режим закрепления основного меню из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision"** › **"WebClient"** > **"SettingGroups"** > **"WebClient"**.
3. Измените значение параметра `UXMainMenuBehavior`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "UXMainMenuBehavior": "0" ①
        }
      }
    }
  }
}
```

① Возможные значения:

- `0` — для использования режима переключения кнопками .
- `1` — для использования режима переключения кнопкой .

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Интерфейс"".
- Конфигурационный файл — "Режим закрепления основного меню".

Скрытие боковой панели в карточках

Чтобы изменить режим скрытия панели из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision"** › **"WebClient"** > **"SettingGroups"** > **"WebClient"**.

3. Измените значение параметра `HideSidebarOnPageClick`:

   Пример включения режима, при котором панель будет скрываться нажатием кнопки **ОК**/**Отмена** и пр., а также при щелчке вне боковой панели:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "HideSidebarOnPageClick": "True" ①
        }
      }
    }
  }
}
```

   ① Возможные значения:

   - `False` (значение по умолчанию) — панель должна скрываться только при нажатии кнопки **ОК**/**Отмена** и пр.
   - `True` — панель также должна скрываться при щелчке вне боковой панели.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Интерфейс"".
- Конфигурационный файл — "Режим скрытия правой боковой панели".

Настройки инструментов

Поведение DVWebtool

В конфигурационном файле Web-клиента поведение DVWebTool при запуске настраивается:

- `DvWebToolAppShutDown` — закрывать DVWebTool после выполнения действия. Значение по умолчанию: `false` — не закрывать.
- `DvWebToolStartUpLaunch` — отвечает за автоматический запуск DVWebTool. Значение по умолчанию: `true` — утилита запускается автоматически при загрузке системы.

Подробнее см.:

- WebToolBuilder — "Установка DVWebTool и DVSupService".
- Конфигурационный файл — "Настройки запуска инструментов".

Порт подключения к DVWebTool

Веб-браузер пользователя подключается к программе *DVWebTool* по адресу `http://localhost:5413`. Администратор может изменить стандартный порт подключения.

Чтобы изменить стандартный порт подключения из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › System**.
3. Измените значение параметра ***WebServicesPort***, указав новый номер порта подключения к программе *DVWebTool*.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "WebServicesPort": 5413 ①
        }
      }
    }
  }
}
```

① Порт подключения к утилите DVWebTool, значение в диапазоне от 1 до 65535.

Подробнее см.:

- Консоль управления Docsvision — "Работа с программой WebToolBuilder".
- Конфигурационный файл — "Изменить порт подключения к DVWebTool".

Настройки аутентификации

Web-клиент в сети без Active Directory

В конфигурационном файле Web-клиента укажите следующие параметры:

```
{
  "SystemUserAccount": "account@domain.com", ①
  "SystemUserPassword": "password", ②
```

```
  "Docsvision": {
    "WebClient": {
      "Server": {
        "SessionManagerConnection": {
          "ConnectionAddress": "http://dvserver.domain.com:5000", ③
          "BaseName": "alias" ④
        }
      }
    }
  }
}
```

① Учётная запись из каталога ALD Pro для подключения к серверу Docsvision.

② Пароль УЗ из каталога ALD Pro для подключения к серверу Docsvision.

③ Адрес сервера Docsvision.

④ Псевдоним БД, от имени которой будет запускаться программа Панель управления Web-клиентом.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Расширения аутентификации"".
- Конфигурационный файл — "Настройка Web-клиента в сети без Active Directory".

Использовать капчу

Администратор настраивает количество неудачных попыток ввода пароля, после которых срабатывает капча, а также время до сброса счётчика неудачных попыток.

Обратите внимание, капча срабатывает, только если сервер Web-клиент работает по HTTPS, см. "Подключение к Docsvision по HTTPS".

```
{
  "Docsvision": {
    "WebClient": {
        "SettingGroups": {
          "WebClient": {
            "MaxFailedLoginAttempts": 3, ①
            "FailedLoginAttemptsTTL":  300 ②
          }
        }
    }
```

```
  }
}
```

① Количество неудачных попыток ввода пароля.

② Время в секундах, после которого счётчик неудачных попыток будет сброшен.

Неудачные попытки входа и срабатывания капчи попадают в журнал Web-клиента. Попытки входа будут записаны в журнал при уровне журналирования как `Error` и серьёзнее, срабатывания капчи будут записаны в журнал при уровне `Warn` и серьёзнее.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Безопасность"".
- Конфигурационный файл — "Предоставить доступ в Web-клиент".

Настройки сессии

Таймаут пользовательской сессии

Администратор может изменить максимальное время ожидания ответа (таймаут) от сервера Docsvision. Время ожидания устанавливается в секундах.

Чтобы изменить таймаут пользовательской сессии из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › System**.
3. Измените значение параметра `ConnectionTimeout`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "ConnectionTimeout": 3600000 ①
        }
      }
    }
  }
}
```

① Время ожидания пользовательской сессии равное 3600000, указывается в секундах.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Интерфейс"".
- Конфигурационный файл — "Изменить время ожидания пользовательской сессии".

Закрытие неактивных сессии

Чтобы изменить стандартные настройки сессии пользователей из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции menu:"Docsvision"["WebClient" > "SettingGroups" > "WebClient".
3. Измените значение параметра:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "SessionSuspendTimeout": 10, ①
          "SessionCloseTimeout": 300, ②
          "SessionInactiveMessageHide": true ③
        }
      }
    }
  }
}
```

① `SessionSuspendTimeout` — время неактивности пользователя в минутах, через которое сессия будет переведена в офлайн-режим. Значение по умолчанию — 10 минут.

② `SessionCloseTimeout` — время неактивности пользователя в минутах, через которое сессия будет закрыта. Значение по умолчанию — 300 минут.

③ `SessionInactiveMessageHide` — отключает показ сообщения *"Подключение остановлено в связи с отсутствием активности"*.

- Значение ***true*** Сообщение не отображается, сессия восстанавливается

при нажатии курсором мыши в любой области окна Web-клиент.

- Значение ***false*** Сообщение отображается, сессия восстанавливается при нажатии курсором мыши на сообщение или в любой области окна Web-клиент (значение по умолчанию).

4. Чтобы отключить перевод неактивной сессии в офлайн-режим, установите ***SessionSuspendTimeout*** в значение ***0***.

   Сервер Web-клиента проверяет неактивность пользователей через промежуток времени ***SessionSuspendTimeout * 0.5***, поэтому перевод пользовательской сессии в офлайн-режим может быть выполнен в промежутке времени до ***SessionSuspendTimeout * 1.5***.

5. Чтобы отключить закрытие неактивных сессий, установите ***SessionCloseTimeout*** в значение ***0***.

Подробнее см.:

- Консоль управления Docsvision — "Пользовательские сессии".
- Конфигурационный файл — "Автоматически закрывать неактивные сессии пользователей".

Время жизни пользовательских cookie

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "AuthCookieLifetime": 120, ①
          "MobileAuthCookieLifetime": 120 ②
        }
      }
    }
  }
}
```

① `AuthCookieLifetime` — период действия пользовательской сессии.

② `MobileAuthCookieLifetime` — период действия пользовательской сессии на мобильных устройствах.

Подробнее см.:

- Консоль управления Docsvision — "Пользовательские cookie".
- Конфигурационный файл — "Управление временем жизни пользовательских cookie".

Настройки электронной подписи

Логотип для штампа ЭП

Чтобы изменить логотип для штампа ЭП из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › "DefaultESImageGenerator"**.
3. Измените значение параметра `LogoImagePath`, указав путь к PNG файлу нового логотипа (относительно каталога `/lib/docsvision/webclient/` или абсолютный).

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "DefaultESImageGenerator": {
          "LogoImagePath": "Content/logo.png"
        }
      }
    }
  }
}
```

Подробнее см.:

- Консоль управления Docsvision — "Штамп ЭП".
- Конфигурационный файл — "Изменить логотип для штампа ЭП".

Включение облачной подписи



Запросы к облачному сервису электронной подписи отправляются через сервер Web-клиента. Это означает, что Web-клиент должен иметь доступ к сервису ЭП для отправки запросов на проверку подписи и для авторизации.

Проверка подлинности подписи может осуществляться локально или через

облачный сервис электронной подписи. Способ проверки подписи зависит от настроек в файле конфигурации Web-клиента.

Чтобы изменить способ проверки подлинности подписи из конфигурационного файла:

1. Откройте файл конфигурации Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к параметру `menu:"Docsvision"["WebClient" > "SettingGroups"` и выполните следующие настройки:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "CloudSignature": {
          "Address": "https://адрес-облачного-сервиса-DSS", ①
          "ClientId": null, ②
          "SignServerAppName": null, ③
          "STSAppName": null, ④
          "DocumentStoreAppName": null, ⑤
          "VerifyUrl": "https://dss.cryptopro.ru/verify/rest/api/signatures", ⑥
          "TSPServiceAddress": null, ⑦
          "SignatureType": "BES", ⑧
          "AuthorizeRedirectUrl": "http://адрес-Web-клиента-
Docsvision/api/CryptoProDss/AcceptAuthorization", ⑨
          "TokenStorageMode": "InMemory" ⑩
        }
      }
    }
  }
}
```

① Адрес сервиса облачной подписи.

② Идентификатор клиента. Настраивается в сервисе облачной подписи КриптоПро.

③ Сервис подписи, обрабатывающий создание транзакций и получение результатов, подтвержденной операции.

④ Сервис аутентификации, ответственный за аутентификацию пользователей для возможности обращений к сервису подписи и подтверждение транзакций.

⑤ `SignServerAppName`, `STSAppName` и `DocumentStoreAppName` — имена сайтов на сервере

КриптоПро DSS. Настраиваются при установке сервиса электронной подписи КриптоПро DSS.

⑥ URL для проверки подлинности подписи.

⑦ Адрес сервиса штампа времени для XLT1.

⑧ Тип подписи:

- `BES` — будет использоваться CAdES-BES.
- `XLT1` — будет использоваться CAdES-XLT1.
- `T` — будет использоваться CAdES-T.

⑨ URL, на который будет переадресован пользователь после авторизации в сервисе электронной подписи. Настраивается в сервисе электронной подписи.

⑩ Режим хранения токена авторизации.
Возможные значения:

- `Disabled` — авторизация будет запрашиваться каждый раз.
- `InSessionCache` — авторизация будет повторно запрошена по окончании сессии.
- `InMemory` — авторизация будет запрошена при перезапуске службы Сервера Web-клиента.
- `InUserProfileEncrypted` — авторизация будет запрошена по истечении срока действия токена.

Подробнее см.:

- Консоль управления Docsvision — "Облачная подпись".
- Конфигурационный файл — "Настройка компонентов КриптоПро".

Скачивание файла в браузер

Чтобы отключить поведение по-умолчанию:

1. Откройте конфигурационный файл, расположенный по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к разделу **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `CalculateSignatureHashOnServer` с `true` на `false`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "CalculateSignatureHashOnServer": true ①
        }
      }
    }
  }
}
```

① Измените значение:

- `false` — будет скачиваться полностью подписываемый файл (устаревшее поведение, библиотека не требуется).
- `true` — скачивается только хэш подписываемого файла (значение по умолчанию, требуется библиотека КриптоПро .NET).

Подробнее см.:

- Консоль управления Docsvision — "Облачная подпись".
- Конфигурационный файл — "Подписание без скачивания файла".

Настройки представлений

Интервал обновления агрегации

Администратор может настроить задержку обновления списка карточек и доступных агрегатов при использовании Elasticsearch.

Чтобы настроить автоматическое применение по таймеру из конфигурационного файла:

1. Откройте файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. В секции **"Docsvision" › "WebClient" > "SettingGroups" > "WebClient"** измените значение параметра `Grid`:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "Grid": {
          "ApplyAggregatesTimeout": "1000", ①
```

```
          "ApplyAggregatesButton": "True" ②
        }
      }
    }
  }
}
```

① По истечении указанного количества миллисекунд будет автоматически обновлён список карточек и доступные агрегаты.

② Флаг определяет, будет ли список карточек обновляться автоматически или перестраиваться по нажатию кнопки **Применить**.
Доступные агрегаты обновляются согласно настройке `ApplyAggregatesTimeout`.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Интерфейс"".
- Конфигурационный файл — "Настройки агрегации".

Настройки кэширования

Клиентское кэширование

Чтобы отключить кэширование настроек:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision"** › **"WebClient"** > **"SettingGroups"** › **"DirectoryCache"**.
3. Измените значение параметров:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "DirectoryCache": {
          "AdvancedLayouts": "False", ①
          "Folders": "True", ②
          "SearchCard": "True" ③
        }
      }
    }
  }
}
```

① `AdvancedLayouts` — отключает кэширование разметок.

② `Folders` — отключает кэширование настроек папок.

③ `SearchCard` — отключает кэширование сохранённых поисковых запросов.

> Параметр `AdvancedLayouts` не используется, рекомендуется использовать актуальную настройку ***SharedDictionaryRefreshPeriod***, см. подробнее в разделе "Функциональность "Общие справочники"".
>
> Параметр `Folders` является дублирующим для параметра `ShortCacheRefreshPeriod`, см. подробнее в разделе "Кэширование обращений к справочникам и карточкам".

Подробнее см.:

- Консоль управления Docsvision — "Кэширование".
- Конфигурационный файл — "Настройки клиентского кэширования".

Время обновления кэша представлений

Чтобы изменить период обновления кэша из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › System**.
3. Измените значение параметра `MinCheckDirectoryTimestampsInterval`.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "MinCheckDirectoryTimestampsInterval": "1000" ①
        }
      }
    }
  }
}
```

① Время ожидания задаётся в миллисекундах.

Подробнее см.:

- Консоль управления Docsvision — "Кэширование".
- Конфигурационный файл — "Изменить время обновления кэша настроек представлений".

Кэширование разметок фильтров данных

Чтобы отключить кэширование разметок из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" > "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `GridFilterCache` в `false`:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "GridFilterCache": "False" ①
        }
      }
    }
  }
}
```

① Возможные значения:

- `False` — кэширование разметок фильтров будет отключено.
- `True` — значение по умолчанию, кэширование включено.

Подробнее см.:

- Консоль управления Docsvision — "Кэширование".
- Конфигурационный файл — "Отключить кэширование разметок фильтров данных".

Управление кэшем папок

Чтобы настроить проверку непрочитанных карточек для группы пользователей в конфигурационном файле:

1. Создайте новую группу в справочнике сотрудников.
2. Скопируйте RowID группы в утилите DVExplorer.
3. Добавьте в конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows настройку `DisableUnreadCountersGroup`:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "DisableUnreadCountersGroup": "RowID" ①
        }
      }
    }
  }
}
```

① RowID группы, скопированный в утилите DVExplorer.

4. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Интерфейс".
- Конфигурационный файл — "Управление кэшем папок".

Кэширование обращений к объектам

Чтобы настроить период обновления из конфигурационного файла, укажите значения в секундах:

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "ShortCacheRefreshPeriod": "600", ①
          "LongCacheRefreshPeriod": "14400" ②
        }
      }
    }
  }
```

```
}
```

① Короткий период обновления кэша и данных в разметке профиля пользователя.

② Длительный период обновления кэша и данных в разметке профиля пользователя.

Настройка `ShortCacheRefreshPeriod` используется для:

- Карточки папок
- Справочника видов
- Конструктора справочников
- Справочника сотрудников
- Справочника контрагентов
- Справочника категорий

Настройка `LongCacheRefreshPeriod` используется для:

- Всего остального, включая справочник разметок

Подробнее см.:

- Консоль управления Docsvision — "Кэширование".
- Конфигурационный файл — "Кэширование обращений к справочникам и карточкам".

Обновление справочников в режиме "только чтение"

Чтобы изменить способ обновления справочников из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции **"Docsvision" › "WebClient" > "SettingGroups" › "System"**.
3. Измените значение параметра `SharedDictionaryRefreshMode`. Параметр определяет способ обновления справочников.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
```

```
        "System": {
          "SharedDictionaryRefreshMode": "0", ①
          "SharedDictionaryRefreshPeriod": "120" ②
        }
      }
    }
  }
}
```

① .Возможные значения:

- `0` — обновление выполняется с интервалом, указанным в параметре `SharedDictionaryRefreshPeriod`.
- `1` — если в одной из сессий данные справочника в режиме только чтение были изменены, обновление будет выполнено при следующем обращении.
- `2` — если в одной из сессий данные справочника в режиме только чтение были изменены, обновление будет выполнено сразу после изменения (значение по умолчанию).

② Задать период обновления справочника равный 2 минутам. Значение указывается в секундах, по умолчанию — 600 секунд.

4. Настройки обновления справочников влияют также на обновление разметок. **На период разработки разметок** рекомендуется установить меньший период обновления, а также уменьшить значения параметров `ShortCacheRefreshPeriod` и `LongCacheRefreshPeriod`, см. подробнее "Кэширование обращений к справочникам и карточкам".

Подробнее см.:

- Консоль управления Docsvision — "Кэширование".
- Конфигурационный файл — "Функциональность "Общие справочники"".

Настройки журналирования

Настройка уровня журналирования

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Измените параметры журналирования в параметрах `targets` и `rules`.

```
{
 "NLog": {
     "throwConfigExceptions": false,
     "extensions": [
         { "assembly": "NLog.Extensions.Logging" },
         { "assembly": "NLog.Web.AspNetCore" }
     ],
     "targets": {
         "async": true,
         "logFile": {
             "type": "File",
             "fileName": "${basedir}/../Logs/WebClient${event-properties:item=Tenant}
PID ${processId} ${shortdate}.log", ①
             "layout": "[${longdate}][${level}][${callsite}] ${message}
${onexception:${newline}${exception:maxInnerExceptionLevel=10:format=shortType,message
,stacktrace:separator=*:innerExceptionSeparator=&#xD;&#xA;&#x9;}}"
         },
         "logConsole": {
             "type": "Console"
         }
     },
     "rules": [ ②
         {
             "logger": "Microsoft.*",
             "maxLevel": "Info",
             "final": true
         },
         {
             "logger": "*",
             "minLevel": "Error",
             "writeTo": "logFile" ③
         },
         {
             "logger": "*",
             "minLevel": "Trace",
             "writeTo": "logConsole" ④
         }
     ]
 }
}
```

① Измените значение параметра `fileName`, чтобы настроить путь для сохранения журнала работы:
В `fileName` можно использовать допустимые для NLog переменные, например: `\Logs\${level}\WebClient_${shortdate}.log`.

② Измените значение параметров `minLevel` и `maxLevel`, например, на `trace`, чтобы включить протоколирование всех типов событий.
Допустимые уровни протоколирования приведены на странице NLog в GitHub.
Для получения дополнительной информации о других настройках NLog обратитесь к документации по данной платформе.

③ Настройки журнала Web-клиента.

④ Настройки журнала Панель управления Web-клиентом.

Подробнее см.:

- В Консоли управления Docsvision настройки могут быть заданы на странице "Дополнительные настройки Web-клиента".
- Конфигурационный файл — "Настройка журналирования".

Включить журнал событий навигации

Чтобы включить протоколирование из конфигурационного файла:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к секции menu:"Docsvision"["WebClient" > "SettingGroups" > "WebClient".
3. Измените значение параметра `EnableRouterLogging` на ***True***:

   Включение протоколирования событий навигации

   ```
   {
     "Docsvision": {
       "WebClient": {
         "SettingGroups": {
           "WebClient": {
             "EnableRouterLogging": "True" ①
           }
         }
       }
     }
   }
   ```

   ① Возможные значения:
   - `True` — протоколирование событий навигации включено.

- `False` — значение по умолчанию, протоколирование событий навигации отключено.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Журналирование"".
- Конфигурационный файл — "Включить журнал событий навигации".

Включить протоколирование запросов сервера Web-клиента

Протоколирование запросов сервера Web-клиента настраивается в конфигурационном файле модуля по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows, за настройку отвечает параметр `W3CLogging`:

```
{
"W3CLogging": {
    "IsEnabled": true, ①
    "LogDirectory": "/var/log/docsvision/webclient/", ②
    "LoggingFields":
"Date,Time,ServerName,Method,UriStem,UriQuery,ProtocolStatus,TimeTaken,ProtocolVersion,Ho
st,UserAgent,Referer,ConnectionInfoFields", ③
    "FileSizeLimit": 10485760, ④
    "RetainedFileCountLimit": 4, ⑤
    "FileName": "webclient-w3c-" ⑥
}
}
```

① Включение/отключение протоколирования событий. Значение по умолчанию: `false`, в стандартной конфигурации остальные настройки отсутствуют.

② Путь к журналу.

③ Журналируемые поля. По умолчанию указываются запросы, ответы, заголовки, дата/время (записывается в формате UTC) и имя сервера. Со всеми допустимыми значениями можно ознакомиться в документации Microsoft.

④ Максимальный размер файла журнала, после которого будет создаваться новый. Значение в байтах, по умолчанию 10485760 байт (10 Мб). Цифра `0` означает отсутствие ограничения по размеру.

⑤ Количество сохраняемых файлов журнала. Допустимые значения от `1` до `10000`, по умолчанию используется `4`.

⑥ Префикс имени файла журнала.

Запросы будут записаны в файл `/var/log/docsvision/webclient/webclient-w3c-{YYYYMMDD.X}.txt` для Linux и `C:\ProgramData\docsvision\webclient\webclient-w3c-{YYYYMMDD.X}.txt` для Windows. Настройка альтернативного расширения поддерживается только при протоколировании с NLog, см. подробнее ниже.



Не удаляйте папку с журналом модуля, т.к. у службы сервера Web-клиента отсутствуют права на создание новой папки.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Журналирование"".
- Конфигурационный файл — "Протоколирование запросов сервера Web-клиента".

Протоколирование запросов с помощью NLog

Чтобы активировать протоколирование запросов с помощью NLog, измените параметры журналирования в параметре: **NLog › targets***:*

```
{
"NLog": {
   "extensions": [
    { "assembly": "DocsVision.WebClient" } ①
   ],
   "targets": {
    "w3cFile": {
      "type": "File",
      "fileName": "${gdc:baseLogFolder}/docsvision/webclient/webclient-w3c-
${date:format=yyyyMMdd}.${processId}.log",
      "layout": {
        "type": "W3CLoggingLayout",
        "columns": [ ②
        { "column": "date" },
        { "column": "time", "layout": "${longdate}" }, ③
        { "column": "c-ip" },
        ]
      }
    }
   },
   "rules": [
    {
      "logger": "*",
      "minLevel": "Debug",
      "writeTo": "w3cFile"
    }
   ]
```

```
}
}
```

① Обязательный параметр, если он не указан, NLog не будет записывать обычные сообщения в журнал.

② Секция задаёт собственный набор колонок. Если список колонок не указан, в журнал будут включены только колонки по умолчанию `date`, `time`, `c-ip`, `s-ip`, `cs-username`, `s-computername`, `cs-method`, `cs-uri-stem`, `cs-uri-query`, `sc-statuscode`, `sc-bytes`, `cs-bytes`, `time-taken`, `cs-host`, `cs(User-Agent)`.

③ Формат представления данных колонки может быть изменён. Если формат не указан, будут использоваться значения по умолчанию. В `layout` можно использовать любые переменные NLog, подробнее см. в документации NLog.

- В Консоли управления Docsvision настройки могут быть заданы на странице "Дополнительные настройки Web-клиента".
- Конфигурационный файл — "Протоколирование запросов с помощью NLog".

Прочие настройки

Настройка отпечатка сертификата кластера Web-клиент

1. Если Web-клиент использует HTTPS, создайте сертификат с помощью OpenSSL. Пример использования утилиты приведён в разделе администрирования модуля Консоль управления Docsvision: "Создание доверенного сертификата для подключения к БД".
2. Установите и настройте Web-клиент на втором компьютере кластера (далее — **УЗЕЛ 2**).



Экземпляр Web-клиента, установленный на втором и последующих узлах кластера, должен быть настроен на работу с теми же сервером и базой данных Docsvision, что и УЗЕЛ 1.

3. В конфигурационном файле `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows на УЗЛЕ 1 укажите путь к полученному сертификату в элементе `"Https":` (`"Http":`, если используется незащищённое соединение). Фрагмент конфигурационного файла приведён ниже:

```
{
"Https": {
  "Url": "https://*:5005",
```

```
  "Certificate": {
    "Path": "/home/user/certificate.pfx", ①
    "Password": "password" ②
    }
  }
}
```

① Путь к сертификату

② Пароль от сертификата



В конфигурации по умолчанию параметры отсутствуют. Предпочтительным способом настройки является Консоль управления Docsvision, добавляйте параметры в конфигурационный файл только при крайней необходимости.

Если настройки указаны в конфигурационном файле, они считываются в первую очередь и имеют приоритет над настройками в Консоли управления Docsvision.

4. Сгенерируйте сертификат на втором узле.
5. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows на УЗЛЕ 2 и укажите путь к полученному сертификату.
6. Повторите шаги инструкции для третьего и последующих узлов кластера Web-клиента.

Подробнее см.:

- В Консоли управления Docsvision настройки могут быть заданы на странице "Дополнительные настройки Web-клиента".
- Конфигурационный файл — "Особенности настройки кластера Web-клиент".

Определение типа клиентского устройства

Пользовательский интерфейс Web-клиента адаптируется под тип клиентского устройства, который определяется по User Agent, передаваемому веб-браузером клиента. Если User Agent не сопоставляется или некорректно сопоставляется с типом устройства, нужно изменить шаблон, по которому выполняется сопоставление:

1. Откройте конфигурационный файл `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows.
2. Перейдите к элементу **"Docsvision" > "WebClient" > "SettingGroups" > "WebClient"**.
3. Измените значение параметра `TabletRegExpString`:

   Определять устройство с User Agent "NEWPAD" как планшет:

   ```
   {
     "Docsvision": {
       "WebClient": {
         "SettingGroups": {
           "WebClient": {
             "TabletRegExpString": "SM-T355|MediaPad|NEWPAD" ①
           }
         }
       }
     }
   }
   ```

   ① В значении параметра через символ `|` нужно указать User Agent соответствующего устройства.

   - `MobileRegExpString` — шаблон определения мобильных устройств нового поколения.
   - `MobileRegExpPart2String` — шаблон определения старых мобильных устройств.
   - `TabletRegExpString` — шаблон определения планшетов.

Подробнее см.:

- Консоль управления Docsvision — "Группа настроек "Сайт"".
- Конфигурационный файл — "Шаблоны для определения типа клиентского устройства".

Предпросмотр с помощью Р7-Офис/OnlyOffice

"Р7-Офис. Сервер документов"/OnlyOffice расширяет стандартную функциональность, позволяя отображать документ напрямую и копировать данные непосредственно из файла.

1. Откройте конфигурационный файл Web-клиента по адресу

`/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows в любом текстовом редакторе. Укажите адрес сервера документов Р7 и определите режим предварительного просмотра:



Для корректной работы предварительного просмотра убедитесь, что URL сервера документов Р7-Офис/OnlyOffice доступен с клиентских рабочих мест.

```
{
  "Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "WebClient": {
          "ServerR7ConnectionAddress": "https://dvserver.preview.com", ①
          "ServerR7UploadDirectory": "/tmp/upload", ②
          "FilePreviewMode": "2", ③
        }
      }
    }
  }
}
```

① `ServerR7ConnectionAddress` — URL сервера документов Р7-Офис/OnlyOffice. Настройка обязательна для заполнения. Если URL сервера документов Р7 не указан, остальные настройки будет проигнорированы.

② `ServerR7UploadDirectory` — каталог загрузки консолидируемых файлов.

③ `FilePreviewMode` — определяет, какой компонент используется для предпросмотра. Значения: `0` — предпросмотр отключен, `2` — предпросмотр с помощью Р7.

Подробнее см.:

- Консоль управления Docsvision — "Настройка "Р7-Офис" в Консоли управления Docsvision".
- Конфигурационный файл — "Настройка "Р7-Офис. Сервер документов"".

Локализация параметров поисковых запросов

В Web-клиенте названия параметров атрибутивных поисковых запросов будут отображаться на языке пользователя, если выполнена соответствующая настройка.

В Windows-клиенте локализованные названия будут отображаться по-прежнему даже после выполненных настроек.

Чтобы настроить локализацию параметров поисковых запросов:

1. Откройте РМА, нажмите кнопку 🔍 и перейдите на вкладку *XML*.
2. Установите флаг **Редактировать XML запрос**, чтобы начать редактировать XML запроса.

   > ❗ Пока ручное редактирование XML запроса активно, запрос не может быть изменён на вкладке *Атрибутивный*.

3. Задайте локализованные названия для каждого из желаемых языков.

   Пример фрагмента XML:

   ```
      <Condition Alias="alias9" ParameterAlias="paramalias16" Name="Дата регистрации с"
   Type="datetime"
             Flags="1" Value="" ContextWord="Today">
        <LocalizedName>
          <LocalizedString Language="ru">Дата регистрации с</LocalizedString> ①
          <LocalizedString Language="en">Registration date from</LocalizedString> ②
        </LocalizedName>
      </Condition>
   ```

   ① Локализованное значение параметра "Дата регистрации с" для русского языка.

   Если локализованные значения не указаны, наименование параметра будет браться из общего имени параметра, указанного в `Condition`.

   ② Локализованное значение параметра "Дата регистрации с" для английского языка.

Правила указания локализованных значений

- Не меняйте значения `ParameterAlias` или `Name`, указывайте локализованное значение и код языка (например, `ru`, `en`) в элементе `Language`.
- Указывайте только одно значение для одной локали.
- Пустые значения не используются системой.

Сопутствующие настройки

Web-клиент

Сопутствующие настройки

Информация о дополнительных настройках Web-клиента, не подходящих ни под одну другую категорию.

- Настроить работу Web-клиента по HTTPS

Настроить работу Web-клиента по HTTPS

См. "Подключение к Docsvision по HTTPS" в документации по настройке Docsvision.

Настройка аутентификации в Active Directory по протоколу LDAPS

При явной аутентификации пользователя в Web-клиенте, со вводом логина и пароля по умолчанию используется протокол LDAP для подключения к Active Directory и проверки указанных учётных данных. При необходимости можно использовать протокол LDAPS, если его поддержка включена на стороне сервера Active Directory.[3]



Для использования LDAPS с Active Directory требуется модуль Платформа версии 6.1 сборки 546 и выше и модуль Web-клиент версии 6.1 сборки 1110 и выше.

Чтобы включить использование LDAPS с Active Directory :

1. В конфигурационном файле укажите настройки "Catalogs":

```
{
  "Catalogs": [{
     "FullDomainName": "example.com", ①
     "NetBiosDomainName": "example", ②
     "LdapOptions": {
       "LdapServerAddresses": [ ③
         "subdomain.example.com"
       ],
       "AuthType": "Basic", ④
       "Port": 636, ⑤
       "EnableLDAPS": true, ⑥
       "Credential": { ⑦
         "UserName": "EXAMPLE\\user01", ⑧
```

```
        "Password": "user001" ⑨
      }
    }
  }],
}
```

① `FullDomainName` — полное имя домена.

② `NetBiosDomainName` — сокращённое имя домена.

③ `LdapServerAddresses` — адреса серверов LDAP. См. подробнее "Настройка работы с несколькими доменами".

④ `AuthType` — тип аутентификации: `Basic` — аутентификация через LDAP (подробнее см. в документации модуля Платформа "Настройка работы с несколькими доменами"), `Oauth` — аутентификация через ЕСИА или Azure.

⑤ `Port` — порт подключения к серверу LDAP.

⑥ `EnableLDAPS` — параметр определяет, будет ли использоваться LDAP с SSL/TLS (LDAPS) для безопасного обмена данными. Задайте для параметра значение `true`. По умолчанию значение `false`.

⑦ `Credential` — имя и пароль учётной записи, под которой процесс обращается к ActiveDirectory.

⑧ `UserName` — логин пользователя модуля для аутентификации LDAP.

⑨ `Password` — пароль учётной записи пользователя для аутентификации LDAP. Пароль можно указывать в открытом или в зашифрованном виде, см. подробнее о шифровании в документации по установке системы, раздел "Защита системной учётной записи"

2. Настройки `"Catalogs"` преобладают над настройками `"Ldap"`, поэтому блок настроек `"Ldap"` можно удалить:

```
{
  "Ldap": {
    "AuthType": "Basic",
    "LookupTimeout": 2000,
    "Port": 389,
    "Credential": {
      "UserName": null,
      "Password": null
    }
  }
```

```
}
```

3. Создайте самоподписанный сертификат, например, с помощью утилиты openssl или используйте уже имеющийся:

```
$ openssl s_client -connect example.com:636
```

4. Получившийся файл поместите на сервер Linux в папку: `/usr/local/share/ca-certificates`.
5. Выполните команду:

```
$ sudo update-ca-certificates -v
```

6. Установите сертификат на сервере Active Directory.
7. Перезапустите службу **dvappserver** для Linux и **Docsvision Platform** для Windows.

Установка минимального количества потоков

Иногда при работе Web-клиента может наблюдаться снижение производительности даже при условии соблюдения всех системных требований. Это может выражаться в виде периодического замедления в начале рабочего дня или после рассылки писем со ссылкой на Web-клиент, в таких случаях замедление обычно продолжается не более 1-2 минут. Время отклика страницы может увеличиваться до минуты.

Причина такого поведения заключается в слишком долгой адаптации сервера Web-клиента к повысившейся активности пользователей.

Решением ситуации является установка минимального числа потоков, которое соответствовало бы операционному состоянию сервера Web-клиента. Значение устанавливается в конфигурационном файле по адресу `/usr/lib/docsvision/webclient/DocsVision.WebClient.runtimeconfig.json`.

Минимальное число потоков определяется в соответствии с числом активных пользователей (максимальному числу активных-сессий). Более точное значение можно определить, проанализировав показатели счетчика производительности в течение рабочего дня:

1. Установите средства dotnet-counter и dotnet-stack для Linux.
2. Запустите средство `dotnet-counter` и проанализируйте с его помощью количество потоков:

```
./dotnet-counters  collect --counters "System.Runtime[threadpool-thread-count]" -p
XXXXX ①
```

① `-p XXXXX` замените на PID процесса ВК

3. В результате в каталоге средства появится файл `counters.csv` со значением счетчика в разное время.
4. Среднее значение счетчика под нагрузкой можно использовать как минимальное число потоков.

   Укажите его в конфигурационном файле `/usr/lib/docsvision/webclient/DocsVision.WebClient.runtimeconfig.json` в параметре `runtimeOptions.configProperties.System.Threading.ThreadPool.MinThreads`:

```
{
  "runtimeOptions": {
    "tfm": "net6.0",
    "frameworks": [
      {
        "name": "Microsoft.NETCore.App",
        "version": "6.0.0"
      },
      {
        "name": "Microsoft.AspNetCore.App",
        "version": "6.0.0"
      }
    ],
    "configProperties": {
      "System.GC.Server": true,
      "System.Reflection.Metadata.MetadataUpdater.IsSupported": false,
      "System.Runtime.Serialization.EnableUnsafeBinaryFormatterSerialization": true,
      "System.Threading.ThreadPool.MinThreads": 100 ①
    }
  }
}
```

① `"System.Threading.ThreadPool.MinThreads"` — минимальное количество потоков.



Обратите внимание! Конфигурационный файл будет

> перезаписан при установке обновления. Рекомендуется изменять настройку в случае, если наблюдаются замедления в работе Web-клиента во время резкого увеличения числа активных пользователей (например, в начале рабочего дня или после рассылки писем со ссылкой на Web-клиент). Можно также превентивно планировать эту конфигурацию при нормальной работе, если ожидается большое число одновременных пользователей.

5. Перезапустите службу Web-клиента.

Изменить порт подключения Web-клиента

По умолчанию Web-клиент работает по протоколу HTTP, использующему порт `5004`. Администратор может установить новый порт для доступа к Web-клиент.

Также администратору доступна возможность настроить подключение по протоколу HTTPS, см. "Подключение к Docsvision по HTTPS" в руководстве по настройке системы.

Чтобы изменить порт доступа к Web-клиенту:

1. Откройте конфигурационный файл Web-клиента по адресу `/usr/lib/docsvision/webclient/appsettings.json` в Linux и `C:\Program Files\Docsvision\WebClient\appsettings.json` в Windows при помощи любого текстового редактора.
2. Укажите порт доступа к Web-клиенту в параметре **Kestrel** › **Endpoints** › **Http** › **Url**.

```
{
  "Kestrel": {
    "Endpoints": {
      "Http": {
        "Url": "http://*:5004" ①
      }
    }
  }
}
```

① Укажите адрес и порт подключения к Web-клиенту.
Не забудьте открыть порт и создать правила, открывающие новые TCP порты для входящих и исходящих подключений.

Чтобы запустить Web-клиент в ОС Linux на порту 443 (или любом ином, не превышающем 1024), должно соблюдаться хотя бы одно из условий:

1. Сервис должен запускаться с правами суперпользователя (ROOT).
2. Должно запускаться ядро, позволяющее непривилегированным пользователям связывать порты:

```
net.ipv4.ip_unprivileged_port_start=1
```

3. Должны быть изменены параметры запуска сервиса:

```
[Service]
AmbientCapabilities=CAP_NET_BIND_SERVICE
```

3. Сохраните изменения конфигурационного файла.
4. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
5. После изменения порта подключения распространите новые настройки на все клиентские машины.

[0] Разметка основного окна Web-клиента.

[1] Кэширование ответов от LDAP поддерживается в модуле Платформа начиная с версии 6.1. сборка 380 и выше, а также в модуле Web-клиент, начиная с версии 6.1. сборка 735 и выше.

[2] Например, корневой сертификат КриптоПро

[3] Активация протокола LDAPS на стороне сервера Active Directory выходит за рамки документации Docsvision.

Руководство пользователя

Запуск Web-клиента

1. Откройте в веб-браузере страницу Web-клиента `http://WebC-server.domain.com:5004/`.

> В мультитенантном режиме имеет значение адрес открытия Web-клиента. В адресной строке до первой точки указывается псевдоним БД, в которой планируется работа. Например: `http://oldDB.company:5004/`. Если вводится адрес по умолчанию без указания псевдонима БД, работа будет происходить с БД, заданной по умолчанию.

2. При открытии Web-клиента будет запрошена аутентификация.
3. Если сервер Web-клиента был запущен недавно и его (пере)запуск ещё не завершён, отображается сообщение `Идёт инициализация сервера, пожалуйста, подождите`. После окончания инициализации сервера будет выполнена переадресация на предыдущую страницу.

Оповещение о смене статуса сотрудника

Если учётная запись пользователя в неактивном состоянии, после открытия Web-клиента будет предложено перевести её в активное состояние.

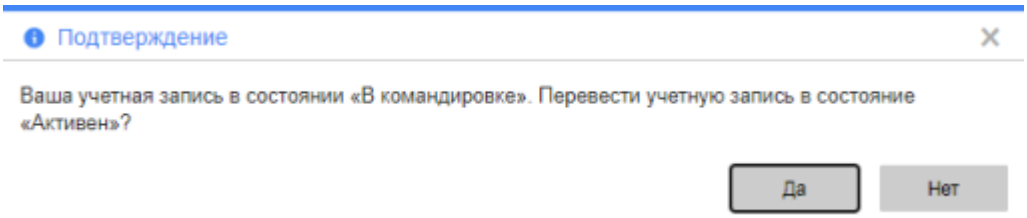


Рисунок 33. Предложение сменить состояние учётной записи пользователя

> Также запрос появляется при обновлении страницы клавишей `F5` и при возобновлении приостановленного подключения (по умолчанию если пользователь бездействовал в течение 5 часов).

При нажатии кнопки **Да** учётная запись пользователя будет переведена в активное состояние. Состояние учётной записи сотрудника в *Справочнике сотрудников* изменится на `Активен`. При отказе от изменения состояние, например при нажатии кнопки **Нет** или ✕, учётная запись останется неактивной. Запрос

появится повторно через 24 часа с момента отказа от изменения состояния.

Аутентификация в Web-клиенте

Web-клиент Docsvision поддерживает несколько способов аутентификации. В зависимости от предпочитаемого способа аутентификации, в Web-клиенте можно авторизоваться:

- С помощью учётной записи Active Directory или другой службы каталогов LDAP(S).
- С помощью расширения аутентификации:

  При использовании авторизации через расширения аутентификации потребуется дополнительная опция лицензии в зависимости от расширения. См. подробнее "Регистрация расширений авторизации".

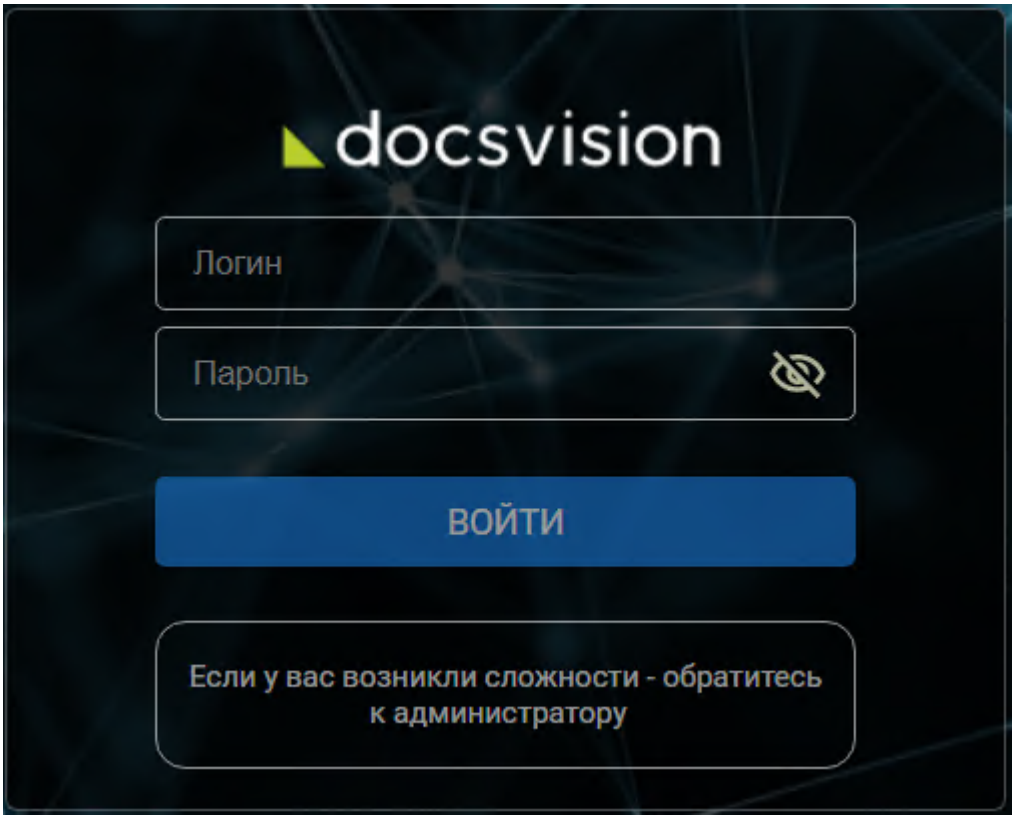


Рисунок 34. Страница аутентификации

Вход с помощью учётной записи

Вход с помощью учётной записи считается основным способом аутентификации.

Не рекомендуется использовать учётные записи системных пользователей для авторизации на клиентских рабочих местах, т.к.

> это может привести к ошибкам в работе системы.

Чтобы войти в Web-клиент с помощью учётной записи:

1. Откройте Web-клиент удобным способом.
2. При открытии Web-клиента:
   - Пользователь будет авторизован автоматически, если настроена прозрачная аутентификация Kerberos.
   - Пользователь должен будет указать учётные данные на странице аутентификации, чтобы войти под своей учётной записью.
     i. На странице аутентификации в поле *Логин* укажите имя учётной записи.
     ii. Введите *Пароль* учётной записи.
     iii. Нажмите кнопку **Войти**.
3. Если в справочнике сотрудников найдена соответствующая карточка сотрудника, будет открыта главная страница. В противном случае появится сообщение об ошибке.

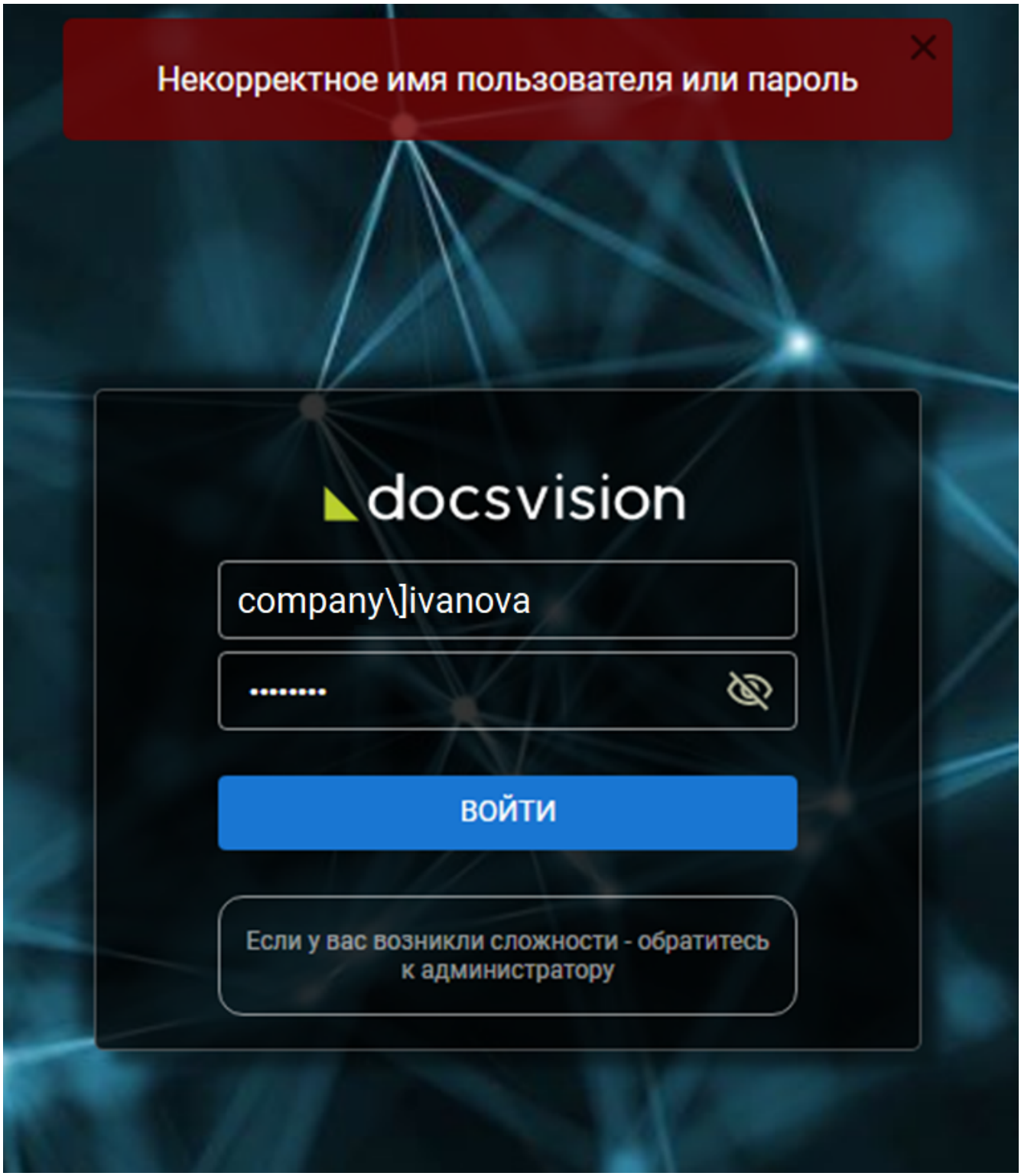


Рисунок 35. Ошибка аутентификации

4. После успешной авторизации пользователь будет переадресован на предыдущую страницу.

Вход с помощью сторонних расширений

Если вы хотите использовать УЗ коннектора вместо УЗ Active Directory или другой службы каталогов:

1. Сообщите администратору, какую учётную запись (УЗ) вы желаете использовать для работы с Web-клиентом.
2. Передайте администратору логин из внешней системы. Подробнее о поддерживаемых коннекторах и их настройке см. "Регистрация расширений авторизации".
3. Администратор в Справочнике Сотрудников Web-клиента отметит УЗ, которую сотрудник может использовать для входа.
4. Авторизуйтесь в Web-клиенте желаемым способом.

Данный раздел документации описывает вход в систему с помощью сторонних сервисов.

- Инструкцию по настройке страницы входа (добавлению формы входа) см. в разделе "Настройка страницы авторизации".
- Инструкцию по регистрации расширений авторизации см. в документации модуля Платформа.
- Как предоставить пользователю возможность входить с использованием сторонних расширений, см. в разделе "Настройка способов входа".

Если в системе отсутствует требуемая опция лицензии для использования выбранного сервиса аутентификации, на странице входа отображается соответствующее сообщение:

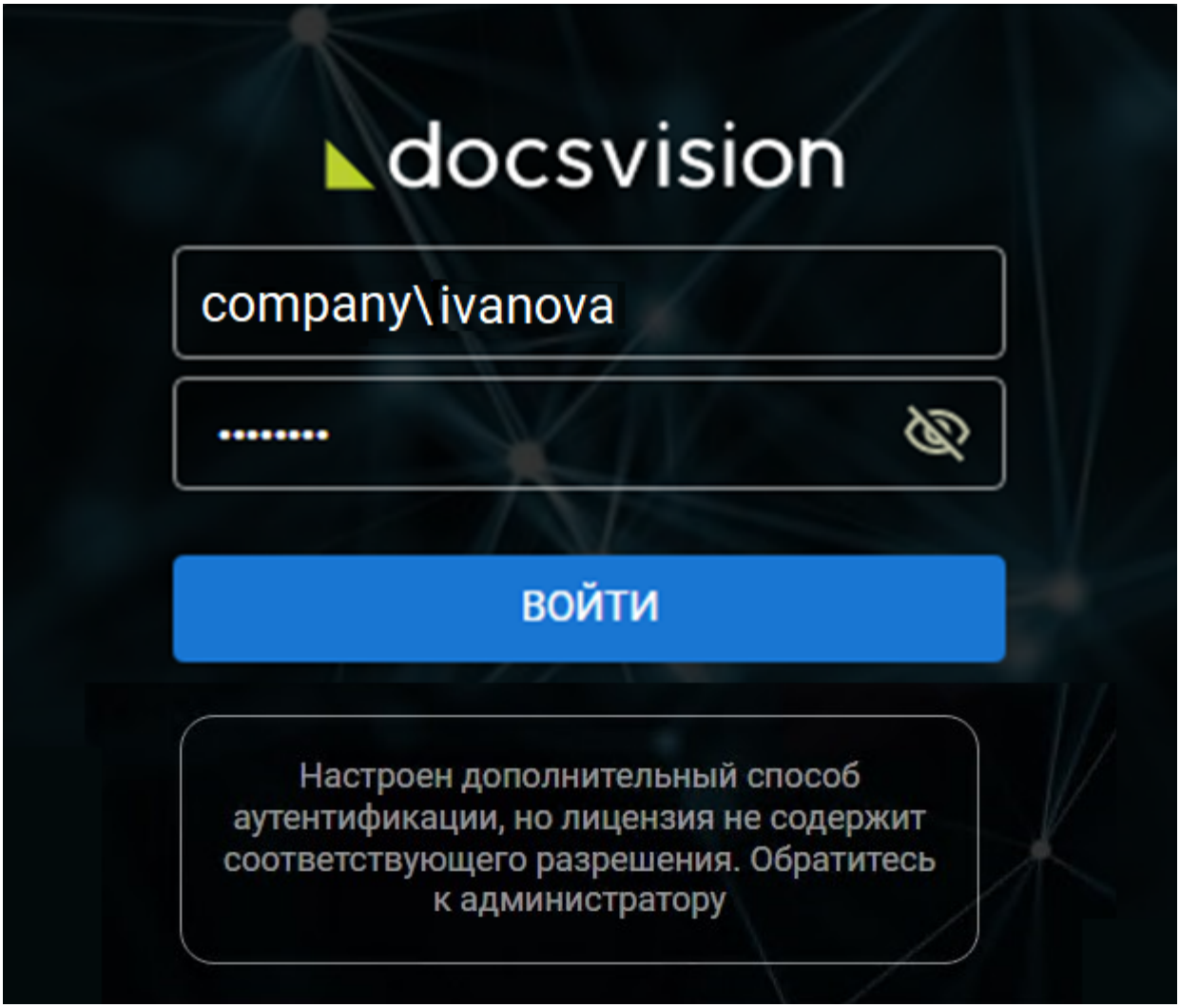


Рисунок 36. Для способа аутентификации требуется лицензия

Чтобы войти в Web-клиент с помощью стороннего расширения:

1. Откройте Web-клиент удобным способом.
2. Нажмите на ссылку *Другой способ входа*.
3. На странице аутентификации в поле *Логин* укажите логин и пароль для входа в аккаунт стороннего сервиса.
4. Нажмите кнопку **Войти**.

   Если в справочнике сотрудников найдена соответствующая карточка сотрудника, будет открыта главная страница. В противном случае появится сообщение об ошибке.

5. Сообщение об ошибке появится, если была указана неверная пара Логин/Пароль или не найден соответствующий сотрудник.

   Попробуйте другой способ аутентификации.

6. При установке некорректного корневого сертификата сайта возможно появление ошибки `SSL-соединение не может быть установлено, см. внутренне исключение` в журнале Web-клиента. В случае возникновения ошибки обратитесь к администратору.

Для входа в Web-клиент важно соответствие указанного адреса электронной почты, телефона или СНИЛС указанным в учётной записи Госуслуг. Если соответствие не установлено, вход не будет осуществлён. Для аутентификации на портале Госуслуг можно использовать любой из доступных логинов: email, телефон, СНИЛС. При первом входе с учётной записью ЕСИА потребуется предоставить права доступа.

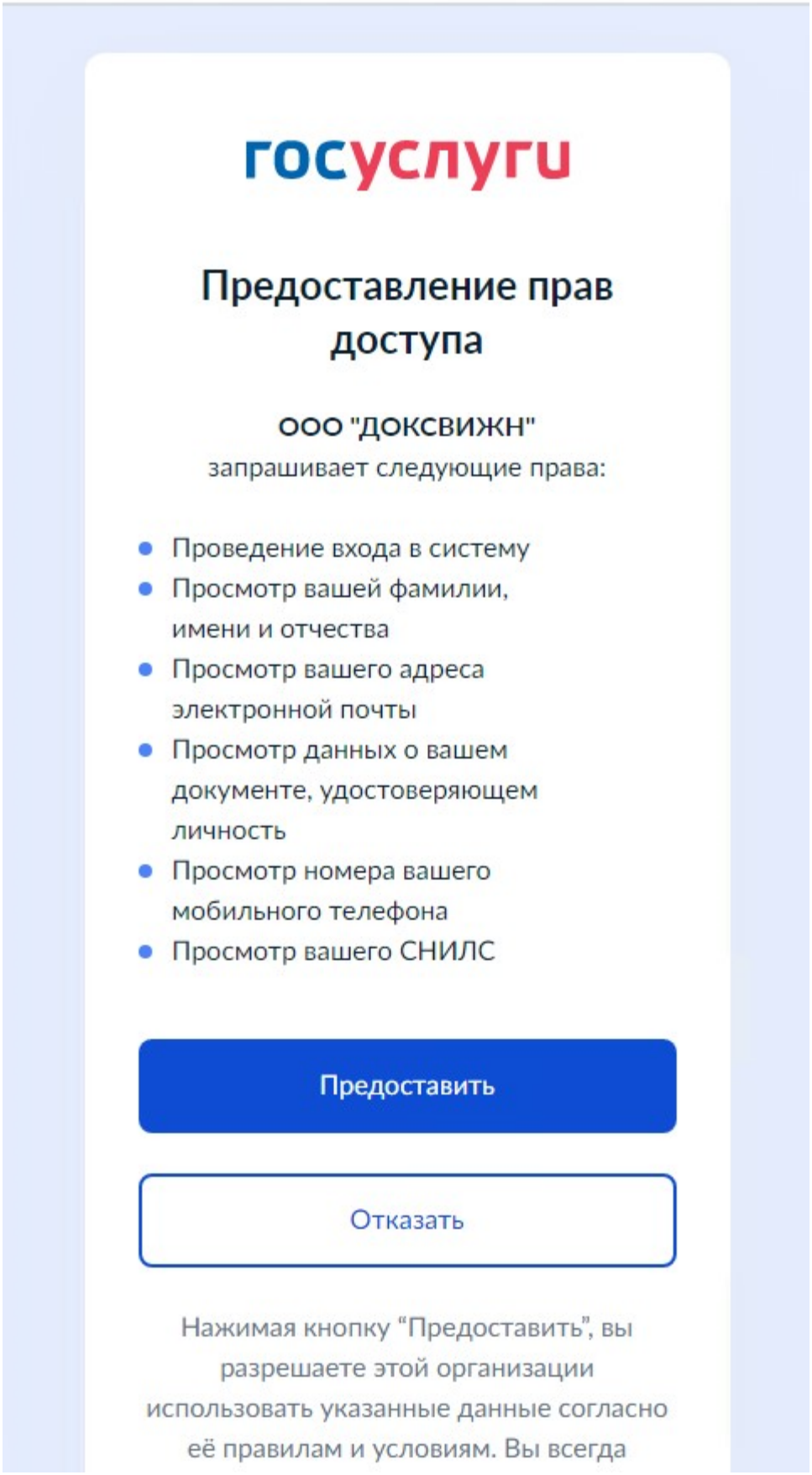


Рисунок 37. Предоставление прав доступа

Когда права доступа предоставлены, сотрудник может использовать учётную

запись Госуслуг для входа в Web-клиент.

Возврат к аутентификации

Для возврата к странице аутентификации перейдите по адресу `http://Сервер-Web-клиента/Account/Login` или воспользуйтесь функцией **Выход** в меню пользователя.

Если открыть страницу аутентификации с дополнительными параметрами `http://%WebClientServer%/Account/Login?userName=userDomain\userAccount`, значение поле "Логин" будет вставлено из адресной строки.

Если пользователь ранее входил с использованием одного из способов, эта информация хранится в браузере и ему будет показана страница приветствия с вводом пароля:

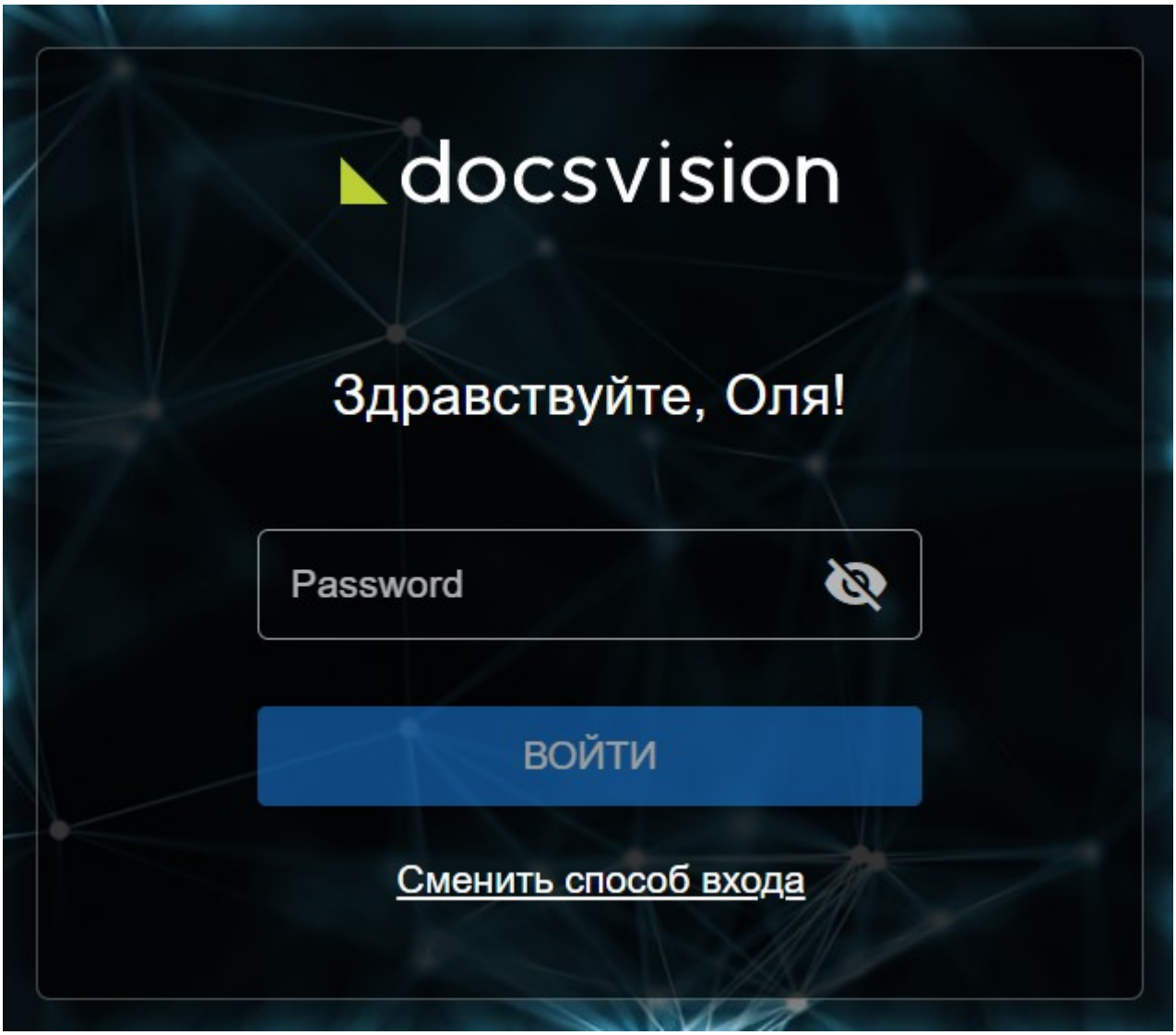


Рисунок 38. Возврат к аутентификации

Прохождение капчи

При работе Web-клиент по протоколу HTTPS (см. руководство администратора) может потребоваться решить капчу (CAPTCHA) — защитный механизм, предлагающий пользователю выполнить задачу перед входом в систему. Механизм предназначен для защиты системы от попыток автоматического подбора пароля.

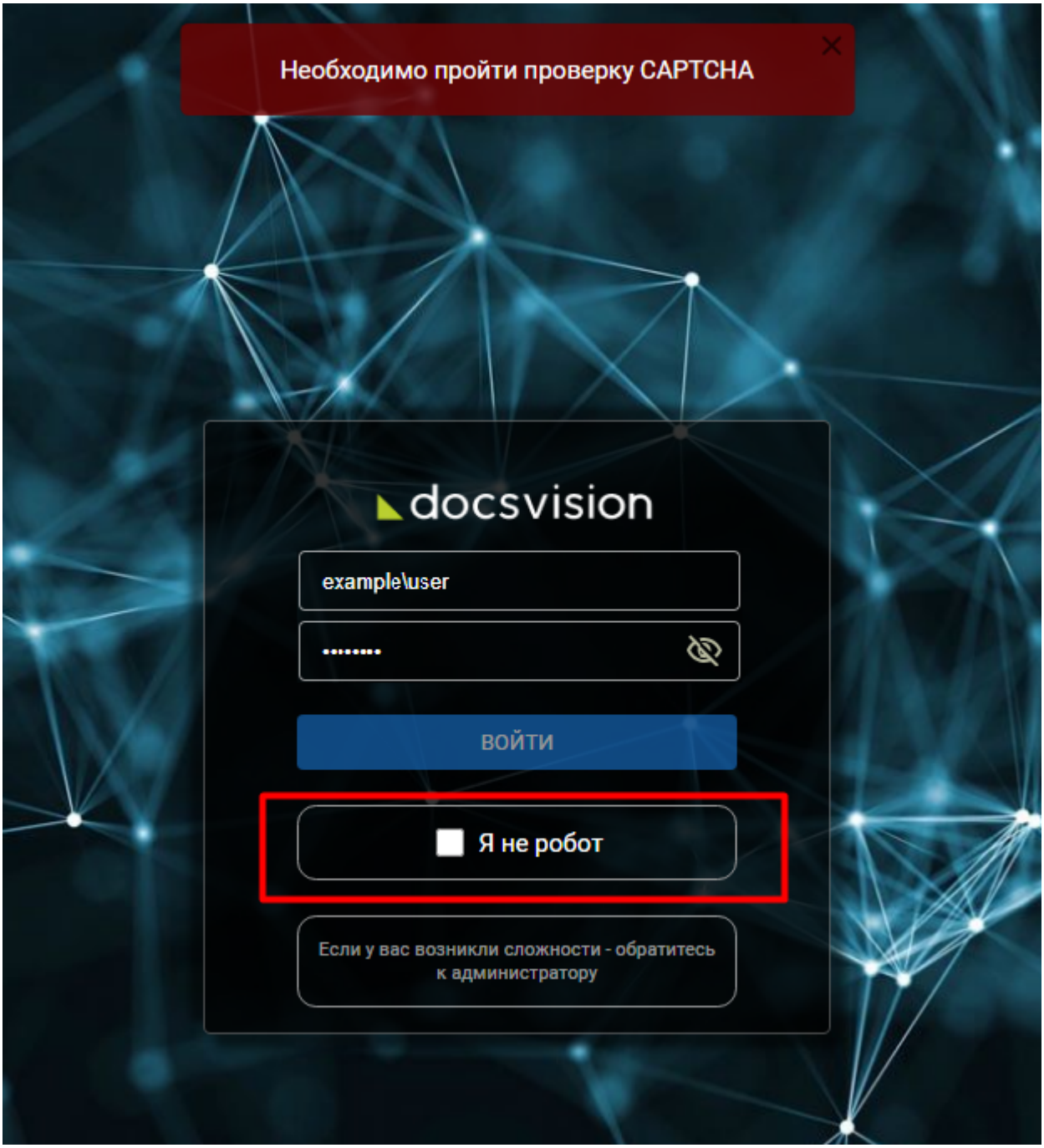


Рисунок 39. Запрос капчи

Капча отображается под формой ввода логина и пароля на странице входа в Web-клиент по умолчанию после трёх неудачных попыток ввода пароля в

течение 5 минут, время и количество попыток может быть изменено администратором. После верного решения капчи пользователь будет успешно авторизован в системе, счётчик неверных попыток будет сброшен. Счётчик также автоматически сбрасывается после неактивности (время задаётся администратором, по умолчанию 5 минут).

Работа в мультитенантном режиме

Web-клиент поддерживает работу в мультитенантном режиме без дополнительных настроек. В мультитенантном режиме имеет значение адрес открытия Web-клиента.

До первой точки указывается псевдоним БД, в которой планируется работа, например, `\http://oldDB.company.com/Docsvision/WebClient/#/Dashboard`.

Подробнее о мультитенантном режиме см. в разделе документации по администрированию Web-клиента.

Установка дополнительных компонентов

Установка компонентов КриптоПро

Для формирования усиленной подписи с использованием сертификатов ЭП необходимо установить компоненты КриптоПро и сертификаты.



Для работы с электронной подписью необходимы компоненты Криптопро. По вопросам их приобретения и лицензирования обращайтесь в компанию ООО "КРИПТО-ПРО".

Формирование усиленной подписи возможно только на персональных компьютерах, соответствующих требованиям.



Шаги, необходимые для настройки КриптоПро в ОС Mac OS можно найти в документации КриптоПро.

Установите расширение для браузера:

1. В Web-клиенте нажмите  (может содержать фото) **> Профиль**. Будет открыто диалоговое окно профиля пользователя.
2. Перейдите в раздел *Установки*.
3. Перейдите по ссылке *Установить КриптоПро ЭЦП Browser plug-in*. Будет загружен файл `cadesplugin.exe`.

4. Закройте веб-браузер и запустите загруженный файл `cadesplugin.exe`. Будет предложено установить *КриптоПро ЭЦП Browser plug-in*.
5. Согласитесь с установкой и дождитесь её завершения.
6. Установите и активируйте расширение для браузера в зависимости от вашей операционной системы и используемого браузера:
   - Windows
   - Linux
   - MacOS

   a. Если расширение плагина имеется в списке установленных, но отключёно, необходимо самостоятельно включить его в браузере.

   b. Если расширение отсутствует в списке или недоступно в магазине приложений, вы можете установить расширение, воспользовавшись инструкцией на сайте КриптоПро. Вы также можете продолжить работу с электронной подписью в браузерах Chromium-Gost или Яндекс Браузер с полной поддержкой необходимой функциональности.
7. Выполните указания в зависимости от типа сертификата:

   a. Если используется неквалифицированный сертификат

      i. Установите неквалифицированный сертификат пользователя в хранилище *Личное* на уровне *Текущий пользователь*.

   b. Если используется квалифицированный сертификат КриптоПро

      i. Установите на компьютер СКЗИ КриптоПро CSP (обратитесь к системному администратору).

      ii. Установите квалифицированный сертификат пользователя в хранилище *Личное* на уровне *Текущий пользователь*.

      iii. Установите корневой сертификат удостоверяющего центра[0] в хранилище *Доверенные корневые центры сертификации* на уровне *Локальный компьютер*.

Интерфейс Web-клиента

При запуске Web-клиента будет открыто основное окно с главной страницей. Пример основного окна Web-клиента, открытого на персональном компьютере, приведён на следующем рисунке.

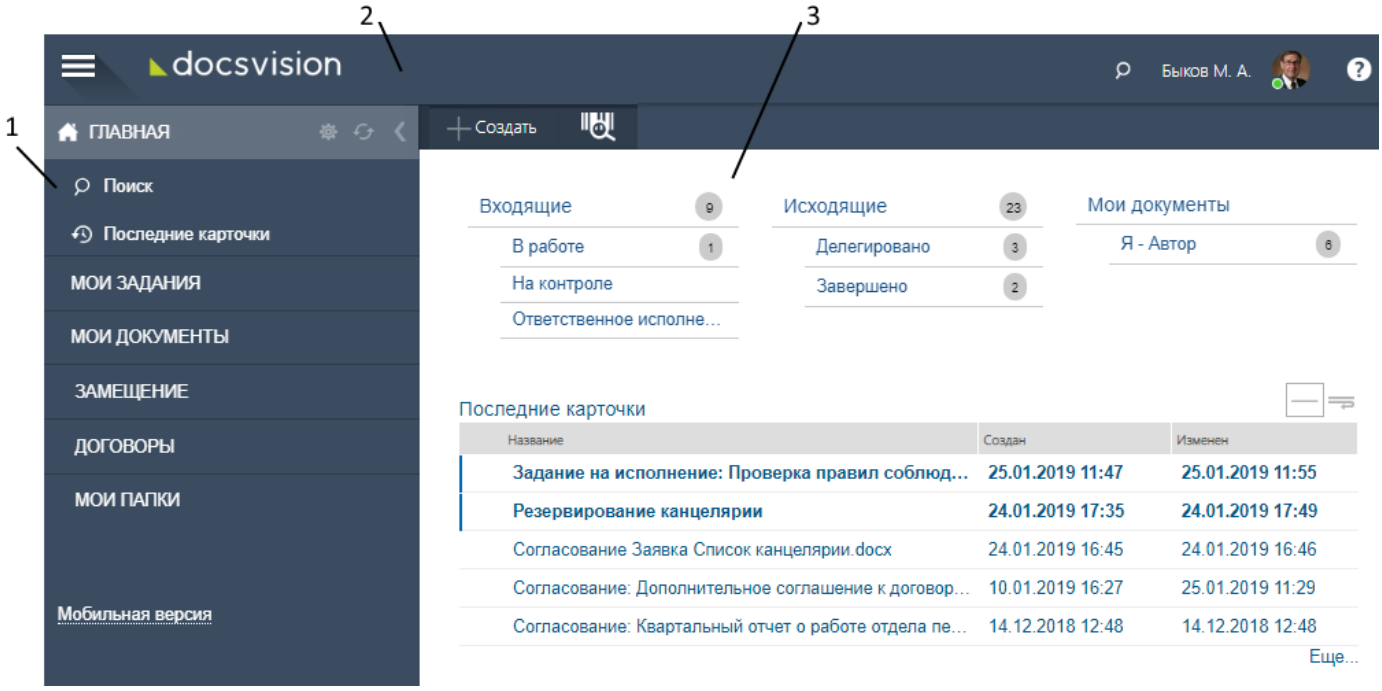


Рисунок 40. Окно Web-клиента с открытой главной страницей

Окно Web-клиента содержит следующие элементы:

1. Основное меню.
2. Панель меню.
3. Область представления папки.

Далее в этом разделе будут описаны элементы интерфейса Web-клиента.

Через три месяца после обновления, при запуске Web-клиента появляется предложение пройти опрос.

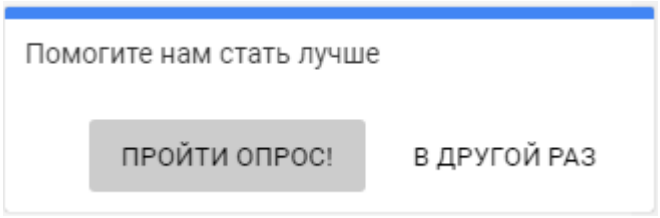


Рисунок 41. Предложение пройти опрос

- Опрос можно пройти сразу же (откроется форма опроса удовлетворённости пользователей) или сделать это в другой раз.
- Опрос можно открыть в любое время из меню пользователя.
- Если закрыть уведомление, нажав **В ДРУГОЙ РАЗ**, в следующий раз уведомление появится через 3 месяца.
- Если нажать **ПРОЙТИ ОПРОС**, но закрыть его менее чем за 30 секунд, в

следующий раз уведомление появится через 3 месяца.

- Если нажать **ПРОЙТИ ОПРОС** и закрыть более, чем за 30 секунд, в следующий раз уведомление появится после обновления версии Web-клиента.
- После прохождения опроса, его результаты считаются актуальными для текущей версии Web-клиента. В следующий раз опрос пройти будет предложено в новой версии Web-клиента.

Главная страница

Главная страница — страница Web-клиента, которая отображается при переходе по ссылке *Главная*. Главная страница содержит *дашборд* — набор информационных блоков с различными данными.

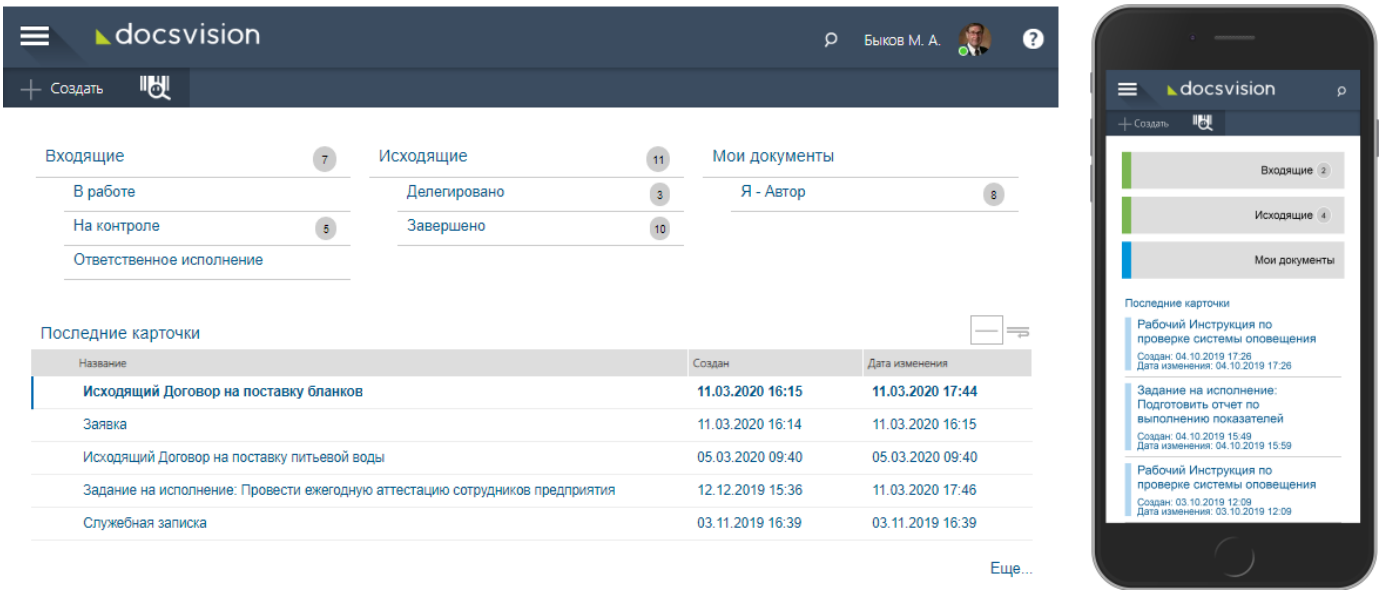


Рисунок 42. Главная страница в полной и мобильной версиях

Дашборд содержит ссылки на папки, предназначенные для доступа к документам и заданиям пользователей. Рядом с названием папки отображается количество непрочитанных карточек. При переходе по ссылке в Области представления будет открыта соответствующая папка.

По умолчанию на *Дашборде* отображаются папки входящих и исходящих заданий, а также папка документов, автором которых является пользователь Web-клиента. Состав папок, представленных на *Дашборде*, может быть изменён.

Список *Последние карточки* содержит ссылки на пять последних карточек, с которыми работал пользователь. Карточки отсортированы по дате создания в обратном порядке. Чтобы перейти к полному списку последних карточек,

нажмите кнопку **Ещё...**.

Главная страница с диаграммами

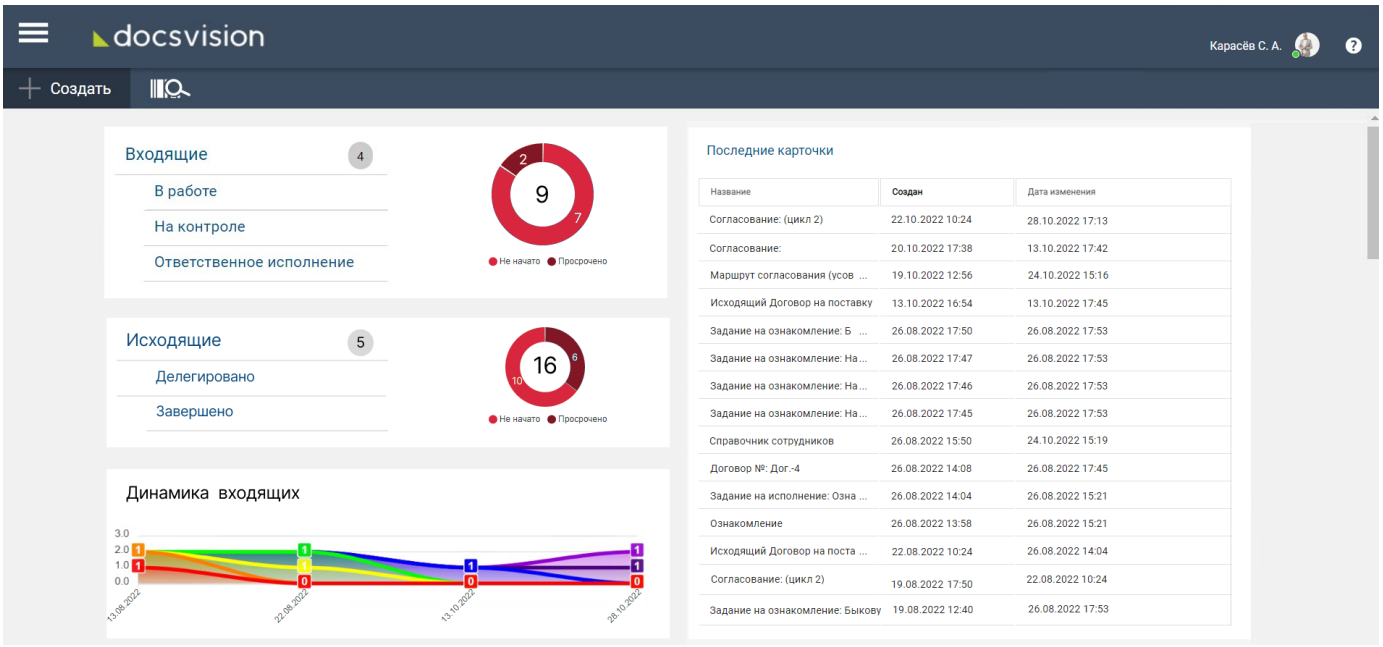


Рисунок 43. Диаграммы на главной странице Web-клиента

Главная страница может отображать статистику по документам в виде диаграмм. Диаграммы отображают статистику в соответствии с заданным поисковым запросом. Отображение диаграмм на главной странице и поисковый запрос настраивает администратор системы.

Основное меню

Полная версия

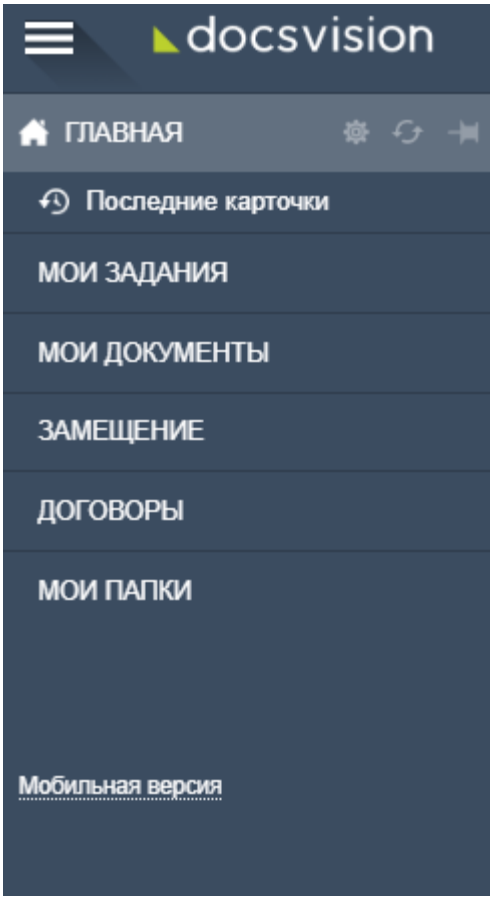


Рисунок 44. Основное меню в полной версии

Таблица 7. Основное меню полной версии Web-клиента:

Кнопки управления:	
	Открывает меню настройки дерева папок.
	Перезагружает настройки папок и сохранённых поисковых запросов.
/	Закрепляет/открепляет Основное меню.
Дерево папок	Общие папки и папки, подключенные пользователем.
Ссылка "Мобильная версия"	Переключает пользовательский интерфейс с обычной версии на мобильную.

По умолчанию основное меню скрыто. Нажмите кнопку , чтобы временно открыть его. Чтобы закрепить панель в открытом состоянии, нажмите кнопку . Для возврата к режиму временного отображения нажмите кнопку или .

При настройке Web-клиента может быть включено альтернативное поведение основного меню. При этом кнопки

закрепления/открепления скрываются, а меню автоматически закрепляется/откреплется при нажатии .

В мобильной версии Web-клиента недоступные кнопки меню скрыты. Вместо недоступных добавляются кнопки открытия Меню пользователя и *Справки* по работе с программой, а также перехода к полной версии Web-клиента.

Мобильная версия

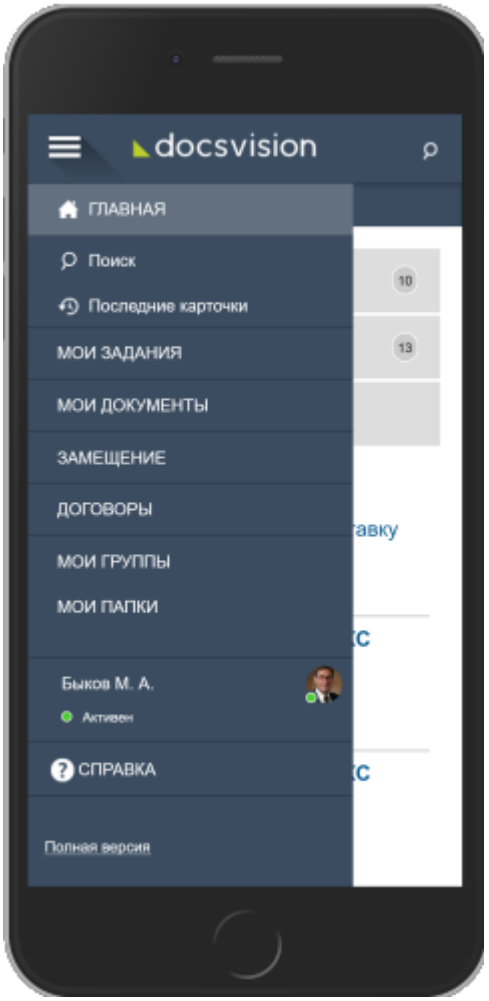


Рисунок 45. Основное меню в мобильной версии

Таблица 8. Основное меню мобильной версии Web-клиента:

Кнопки управления:	
	Перезагружает настройки папок и сохранённых поисковых запросов.
/	Закрепляет/открепляет Основное меню. Кнопка доступна всегда, но в мобильной версии функция работает только на планшетах.

Дерево папок	Общие папки и папки, подключенные пользователем.
Ссылка "Полная версия"	Переключает пользовательский интерфейс с обычной версии на мобильную.

Стандартные папки



Состав элементов в дереве папок может быть изменён при настройке Web-клиента.

Дерево папок включает следующие ссылки:

Главная

Открывает главную страницу Web-клиента.

Избранное

Открывает папку "Избранных карточек".

Последние карточки

Открывает список последних открытых карточек.

Мои задания

Задания пользователя:

Входящие

Задания в состояниях `Не начато`, `Делегировано`, `Возвращено с делегирования` и `На доработке`, в которых исполнителем является текущий пользователь.

В работе

Задания текущего пользователя в состояниях `В работе` и `Отложено`.

Завершено

Задания текущего пользователя во всех состояниях, кроме `Подготовка`, `Отозвано` и `Отклонено`.

На контроле

Задания во всех состояниях кроме `Подготовка` (включая завершённые за последние 30 дней), в которых текущий пользователь является контролёром.

Ответственное исполнение

Задания во всех состояниях кроме Подготовка, Отозвано и Завершено, в которых текущий пользователь является ответственным исполнителем и не является текущим исполнителем.

Исходящие

Незавершённые задания, в которых текущий пользователь является Автором.

Делегировано

Задания в состояниях Делегировано и В работе, которые были делегированы текущим сотрудником.

Завершено

завершённые и отменённые за последние 30 дней задания, в которых текущий пользователь является Автором.

Поиск заданий

Запускает диалог поиска заданий по атрибутам.

Мои документы

Документы, в которых текущий сотрудник является автором.

Поиск документов

Запускает диалог поиска документов по атрибутам.

Я автор

Открывает список документов, в которых пользователь является автором.

Замещение

Задания постоянного и временного заместителя:

Входящие

Задания во всех состояниях кроме Завершено и Отозвано, в которых текущий пользователь является постоянным или временным заместителем исполнителя.

В работе

Задания заместителя в состояниях В работе и Отложено.

На контроле

Задания во всех состояниях кроме Подготовка (включая завершённые за последние 30 дней), в которых текущий пользователь является заместителем контролёра.

Ответственное исполнение

Задания во всех состояниях кроме Подготовка, Отозвано и Завершено, в которых замещаемый сотрудник является ответственным исполнителем и не является текущим исполнителем.

Исходящие

Незавершённые задания, в которых текущий пользователь является заместителем Автора.

Делегировано

Задания в состояниях Делегировано и В работе, которые были делегированы замещаемым сотрудником.

Завершено

Завершённые и отменённые за последние 30 дней задания, в которых текущий пользователь является заместителем Автора.

Договоры

Содержит папки поиска договорных документов и отчёты:

Действующие договоры

Система виртуальных папок с договорами, отбираемыми по различным параметрам:

По контрагенту

Действующие договоры, сгруппированные по контрагенту.

По сроку действия

Действующие договоры, сгруппированные по сроку действия (по кварталам).

Договоры с доп. соглашением

Запускает диалог поиска дополнительных соглашений и актов к договорам, заключённым за определённый период.

Заключённые договоры

Система виртуальных папок с договорами по годам заключения:

2018 — договоры и дополнительные соглашения

Договоры и дополнительные соглашения, заключённые в 2018 году и сгруппированные по контрагенту.

2019 — договоры и дополнительные соглашения

Договоры и дополнительные соглашения, заключённые в 2019 году и сгруппированные по контрагенту.



Папки для выбора договоров за другие года могут быть созданы при настройке Web-клиента.

Исполнение договоров

Запускает диалог поиска актов, зарегистрированных за определённый период.

Поиск

Запускает диалог поиска договорных документов по атрибутам.

Договоры к ознакомлению

Договоры для сотрудников группы ознакомления. Она представляет собой структуру папок с настроенными поисковыми запросами, предназначенную для поиска карточек договоров, с которыми может ознакомиться текущий пользователь:

Заключённые

Заключённые договоры и дополнительные соглашения, в которых текущий пользователь входит в *Группу ознакомления* с договором.

Согласование

Договоры и дополнительные соглашения в состоянии Подготовка, На согласовании, На согласовании с контрагентом, в которых текущий пользователь входит в *Группу ознакомления* с договором.

Мои договоры

Договоры в состоянии Подготовка, в которых текущий или временно замещаемый им сотрудник является *Ответственным* или *Подготовившим*. Включает подпапки с дополнительным условием по состоянию договора:

Завершённые

Договоры и дополнительные соглашения в состоянии Завершено.

Заключённые

Договоры и дополнительные соглашения в состоянии Заключёно.

Подготовка и согласование

Договоры в состоянии Подготовка, На согласовании, На согласовании с контрагентом, Согласован, На подписании, Подписан, На подписании контрагентом.

Отчёты

Договоры и ДС без подписанного оригинала

Запускает диалог поиска договоров и дополнительных соглашений в состояниях Подписан и Заключён, для которых отсутствует подписанный оригинал (в карточке не заполнено поле Получен оригинал).

Договоры с истекающим сроком окончания

Запускает диалог поиска договоров и дополнительных соглашений в состояниях Подписан и Заключён, у которых заканчивается срок действия.

Мои группы

Содержит папки *рабочей группы* пользователя. Недоступна, если пользователь не включён в рабочую группу.

Мои папки

Содержит папки, подключенные пользователем.

Справочники

Список справочников, доступных в Web-клиенте. На отображение справочников в списке могут влиять настройки системы и установленные модули. В списке могут отображаться, например, "Справочник сотрудников".

Панель меню

Панель меню расположена в верхней части пользовательского интерфейса Web-клиента.

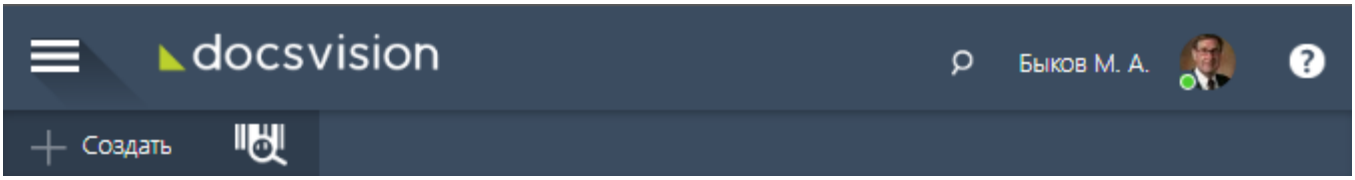


Рисунок 46. Панель управления

Таблица 9. Панель меню содержит следующие кнопки:

	Открыть список папок.
	Открыть строку поиска.
	Открыть меню пользователя. На кнопке отображается фото (если установлено), а также цветовая метка, соответствующая состоянию учётной записи сотрудника.  Состояние может быть изменено администратором в Справочнике сотрудников или самим пользователем в окне Профиль пользователя. В мобильной версии Web-клиента меню пользователя находится в основном меню.
	Открыть страницу с руководством пользователя.
	Открыть меню создания карточки.
	Открыть строку поиска карточки по штрих-коду.

После перехода в папку, на панели меню будет отображено название текущей папки, а также переключатель представления.



Рисунок 47. Панель управления страницы папки

Меню пользователя

Для доступа к меню пользователя нажмите кнопку  на панели меню в полной версии Web-клиента или в основном меню в мобильной версии.

Будет открыто меню, представленное на рисунке ниже.

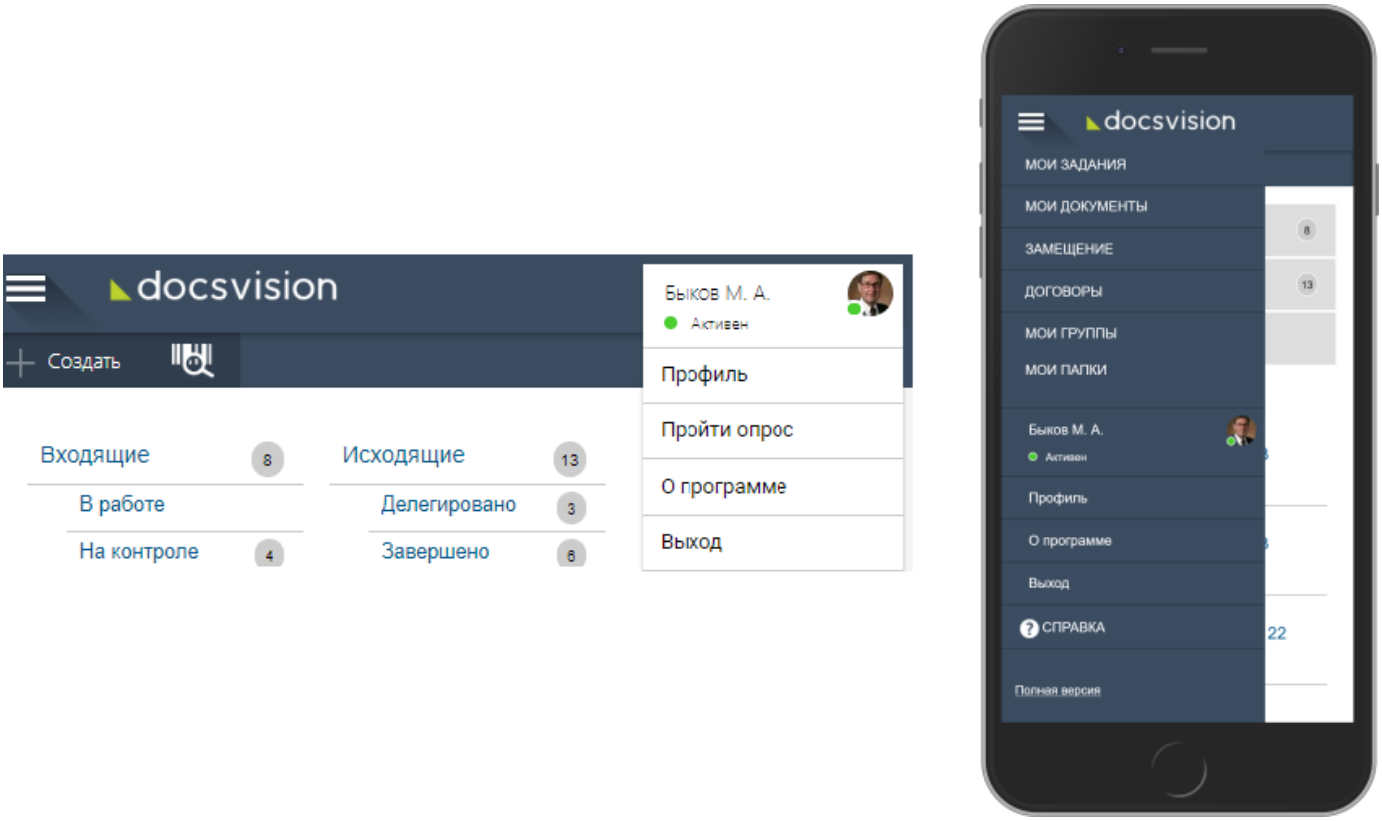


Рисунок 48. Меню пользователя в полной и мобильной версиях

Меню пользователя содержит следующие элементы:

Верхняя строка — В строке расположены: фото, ФИО и статус учётной записи сотрудника.

Профиль — Открывает Профиль пользователя.

Пройти опрос Открывает форму опроса удовлетворенности пользователей.

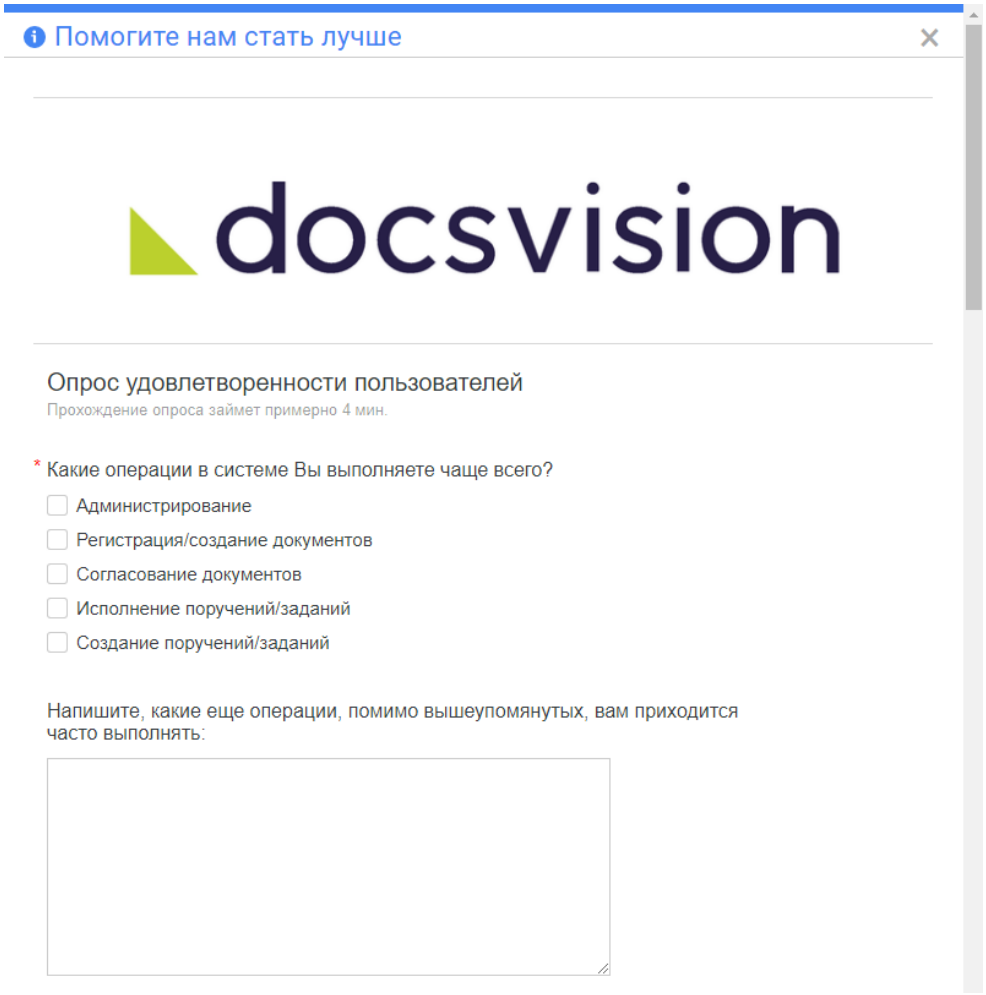


Рисунок 49. Опрос удовлетворенности пользователей

Ответы будут направлены в компанию DocsVision и использованы в целях улучшения продукта.

О программе Открывает окно с информацией о Web-клиенте.

Выход Нажатие на кнопку закончит пользовательский сеанс работы с Web-клиентом.

Информация о Web-клиенте

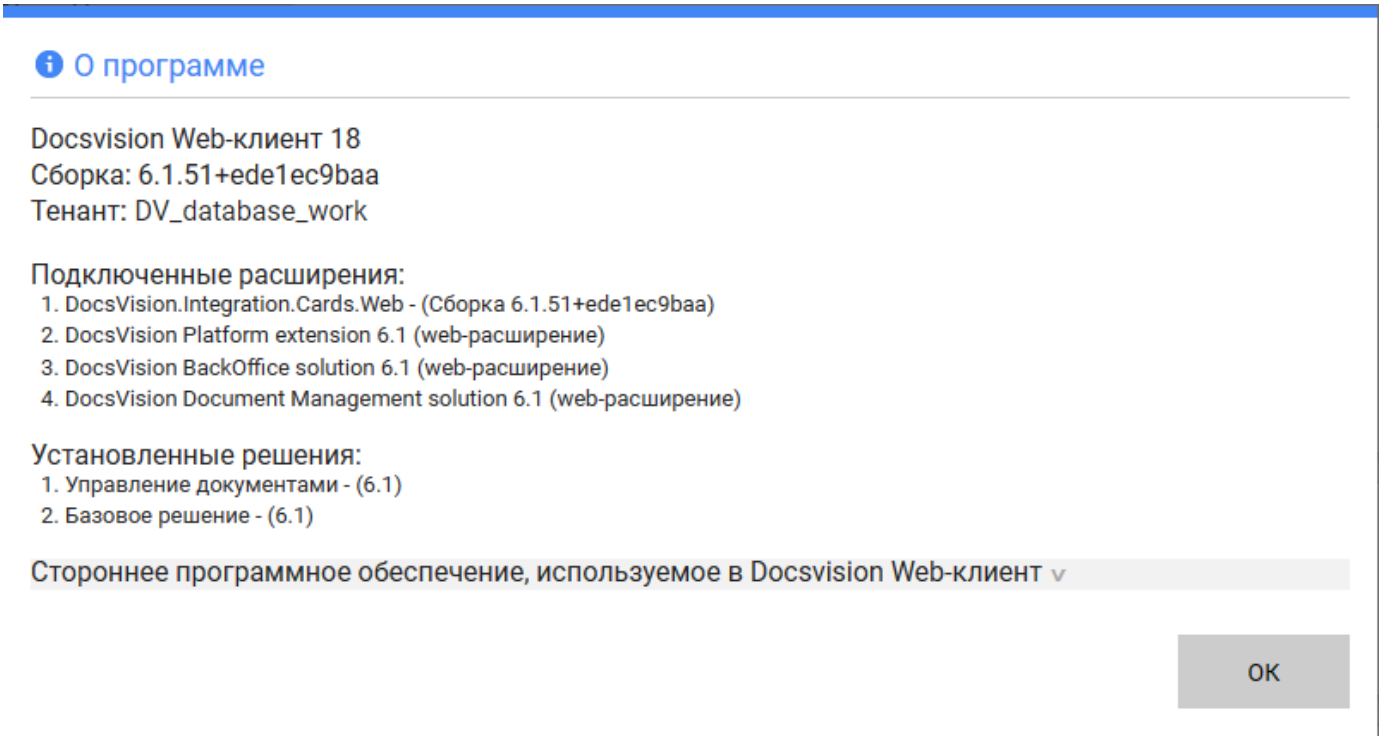


Рисунок 50. Просмотр иинформации о Web-клиенте

В окне информации отображаются следующие сведения:

- Версия сборки Web-клиента.
- Тенант (название текущей используемой базы данных Docsvision).
- Список подключенных серверных расширений Web-клиента с версиями.
- Список установленных Решений с версиями.
- Список используемого стороннего программного обеспечения.

Профиль пользователя

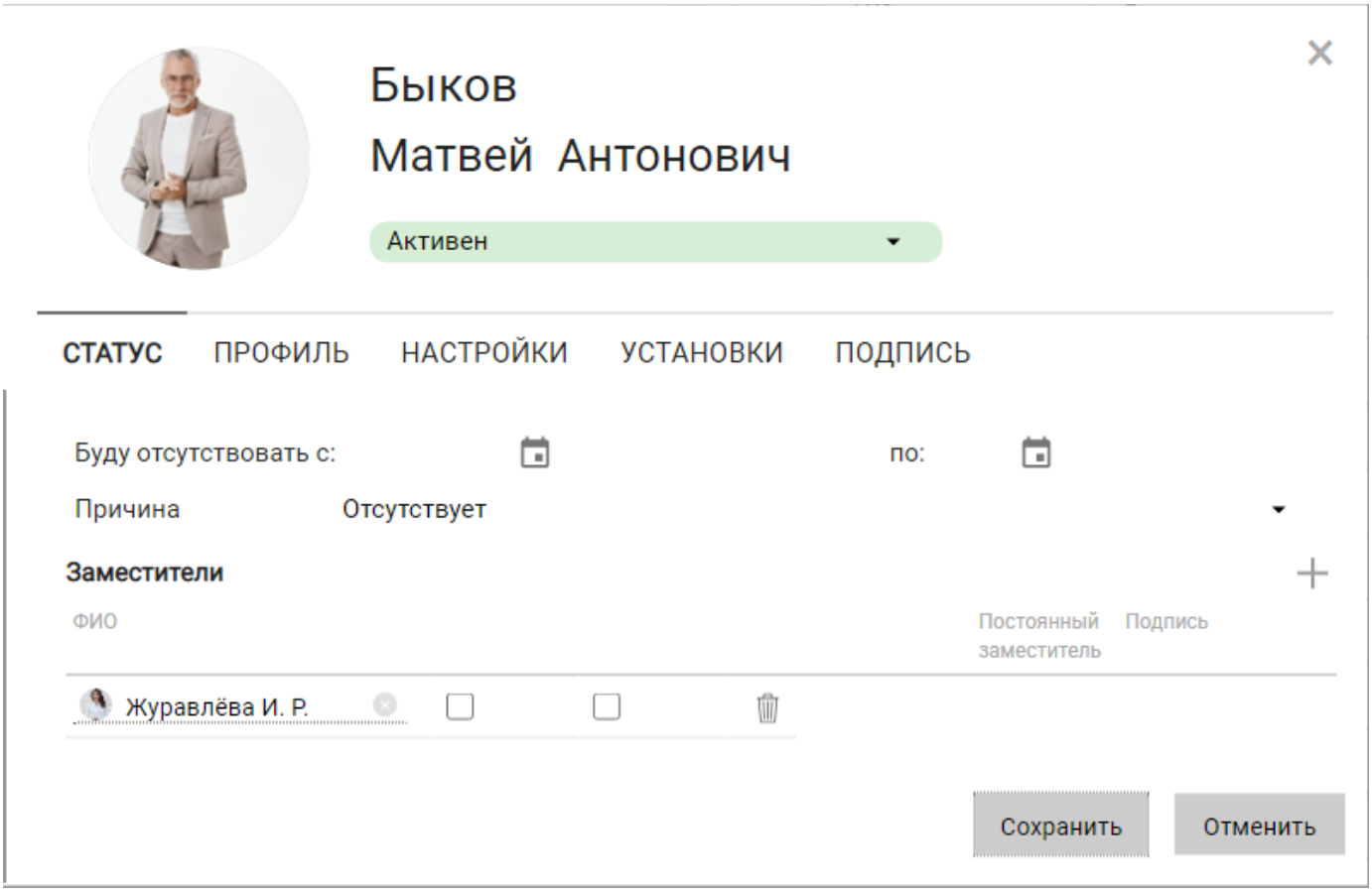


Рисунок 51. Профиль пользователя в полной версии Web-клиента

Страница профиля содержит:

- Фотографию сотрудника (установленную в Справочнике сотрудников).

  Поддерживаются файлы изображений в форматах: `.jpeg` и `.jpg`. Рекомендуемый размер фотографии 130x130 пикселей, изображения большего размера будут сжаты до соответствующего разрешения.

- ФИО сотрудника.
- Раскрывающийся список для изменения статуса учётной записи сотрудника.
- Вкладки, например, *Статус*, *Профиль*, *Настройки*. Подробнее вкладки описаны ниже.

Статус

В данной вкладке профиля содержится информация о планируемых периодах и причине отсутствия сотрудника, а также список его заместителей.

Пользователь может установить период неактивности в формате `с ⋯ по` в специальной графе:



Рисунок 52. Указание периода неактивности

Если период недоступности указан, в информации о сотруднике отображаются период и причина отсутствия, заместители сотрудника отображаются. Если период недоступности не задан, отображается только статус.

Флаг **Постоянный заместитель** даёт заместителю право находить, просматривать и исполнять задания замещаемого, даже если основной исполнитель активен и сам его исполняет.

Флаг **Подпись** даёт заместителю право подписываться за замещаемого в заданиях.

В мобильной версии Web-клиента флаг **Подпись** недоступен.

Чтобы сохранить изменения, нажмите кнопку **Сохранить**. Изменения сохраняются в *Справочник сотрудников*.

Когда пользователь устанавливает неактивный статус, статус устанавливается с 00:00 первого дня периода неактивности до 23:59 последнего дня периода.

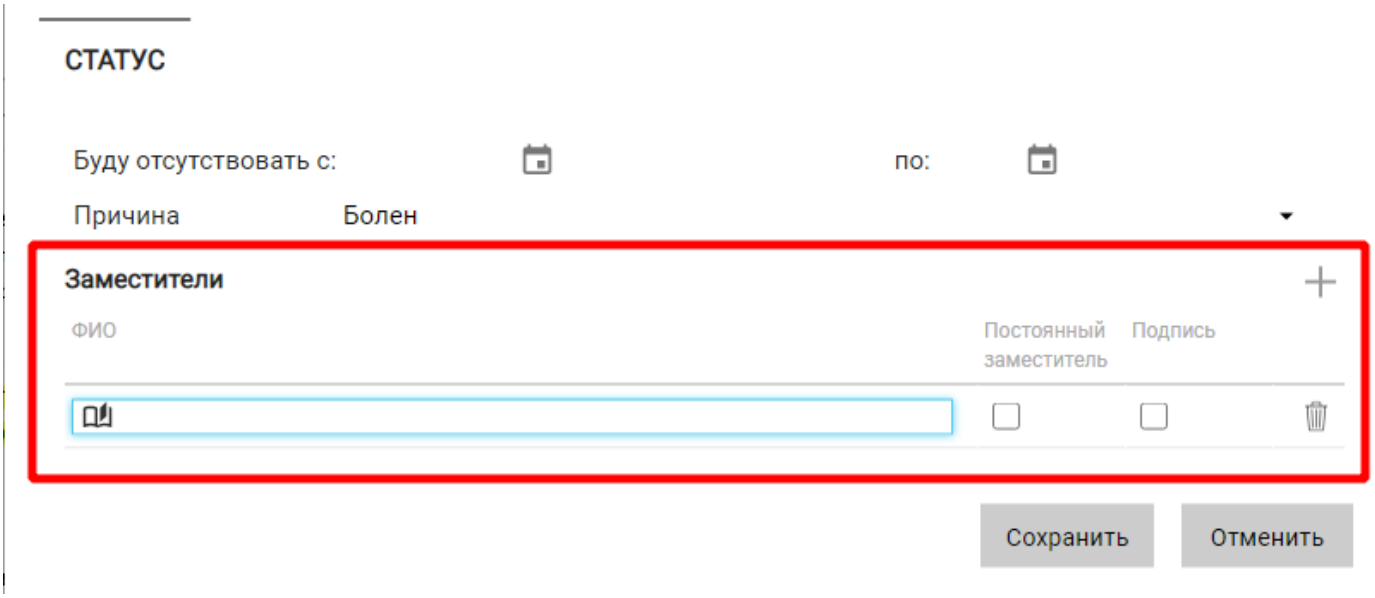


Рисунок 53. Указание заместителей

Из меню смены статуса пользователь также может добавить или удалить заместителей.

Если область добавления заместителей не отображается, нажмите на кнопку +, чтобы добавить строку выбора. Нажмите на кнопку повторно, чтобы выбрать ещё одного заместителя.

Для выбора используется элемент управления Записи справочника сотрудников. Он открывает справочник сотрудников только на чтение, добавить, удалить или изменить сотрудника из этого диалога нельзя. Для выбора доступны узлы справочника, на которые у текущего пользователя есть права.

Чтобы отредактировать данные о статусе (статус, даты, заместители), права на редактирование не нужны. Чтобы изменить информацию на других вкладках профиля пользователя, права потребуются.

Профиль

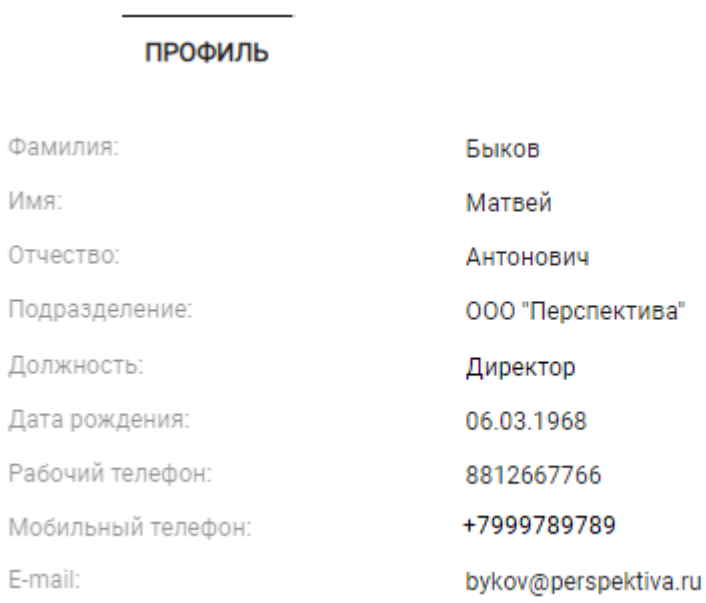


Рисунок 54. Вкладка "Профиль" профиля пользователя

Данный раздел содержит личные данные сотрудника. Личные данные сотрудника можно изменить в Справочнике сотрудников.

Настройки

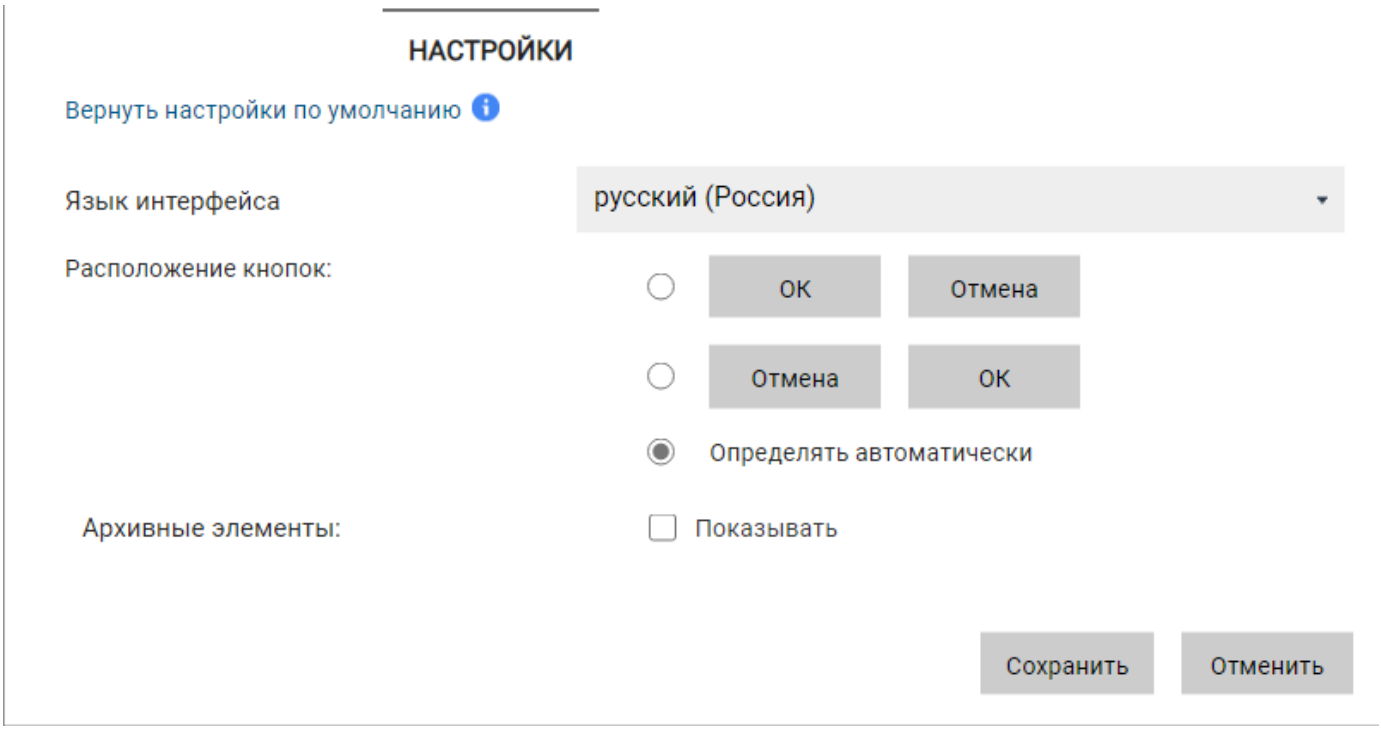


Рисунок 55. Вкладка "Настройки" профиля пользователя

Элементы вкладки "Настройки":

Ссылка "Вернуть настройки по умолчанию"

Нажатие на ссылку очищает все настройки и данные, сохранённые в локальном хранилище браузера. Очищаются, в том числе, следующие настройки (список не полный): расположение кнопок, язык интерфейса, настройки модуля.

Нажатие на ссылку сбрасывает настройки пользователя. Не путать с кнопкой "Сбросить настройки представления" в настройках грида.

Список последних выбранных сотрудников в ЭУ "Записи справочника сотрудников" очищаются автоматически при перезапуске службы Сервера Web-клиента.

Раскрывающийся список "Язык интерфейса"

Позволяет изменить язык пользовательского интерфейса.

Расположение кнопок	Переключает взаимное расположение кнопок "ОК" и "Отмена" в пользовательском интерфейсе. Расположение кнопок может определяться автоматически в зависимости от операционной системы (пункт "Определять автоматически").

Архивные элементы

Настройка регулирует отображение архивированных карточек для текущего пользователя.

Таблица 10. Настройка отображается только для только для пользователей, входящих хотя бы в одну из следующих групп:

Группы на сервере Docsvision	**Группы в справочнике сотрудников Docsvision**
DocsVision Administrators	*Администраторы*
DocsVision Security Administrators	*Администраторы безопасности*
DocsVision Archive Operators.	*Операторы архива*

Если флаг установлен, помимо обычных карточек во всех папках будут отображаться также архивированные карточки. Архивные карточки также будут отображаться в результатах поиска.

Установки

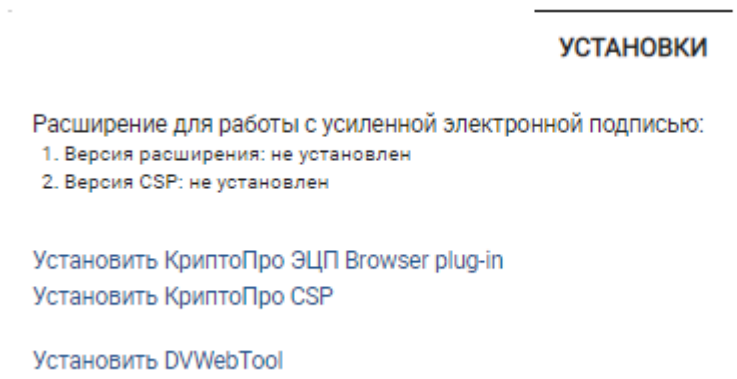


Рисунок 56. Вкладка "Установки" профиля пользователя

В данной вкладке содержится информация об установленных расширениях для работы с ЭП и ссылки для загрузки компонентов КриптоПро. Компоненты КриптоПро необходимы для работы с сертификатами ЭП. В данной вкладке

также отображается ссылка для загрузки программы *DVWebTool* и локальными устройствами и программами, см. подробнее пункт "Установка DVWebTool и DVSupService" в руководстве администратора.

Если компоненты КриптоПро установлены, будут показаны их версии.



Вкладка *Установки* профиля пользователя доступна только в полной версии Web-клиента.

Подпись

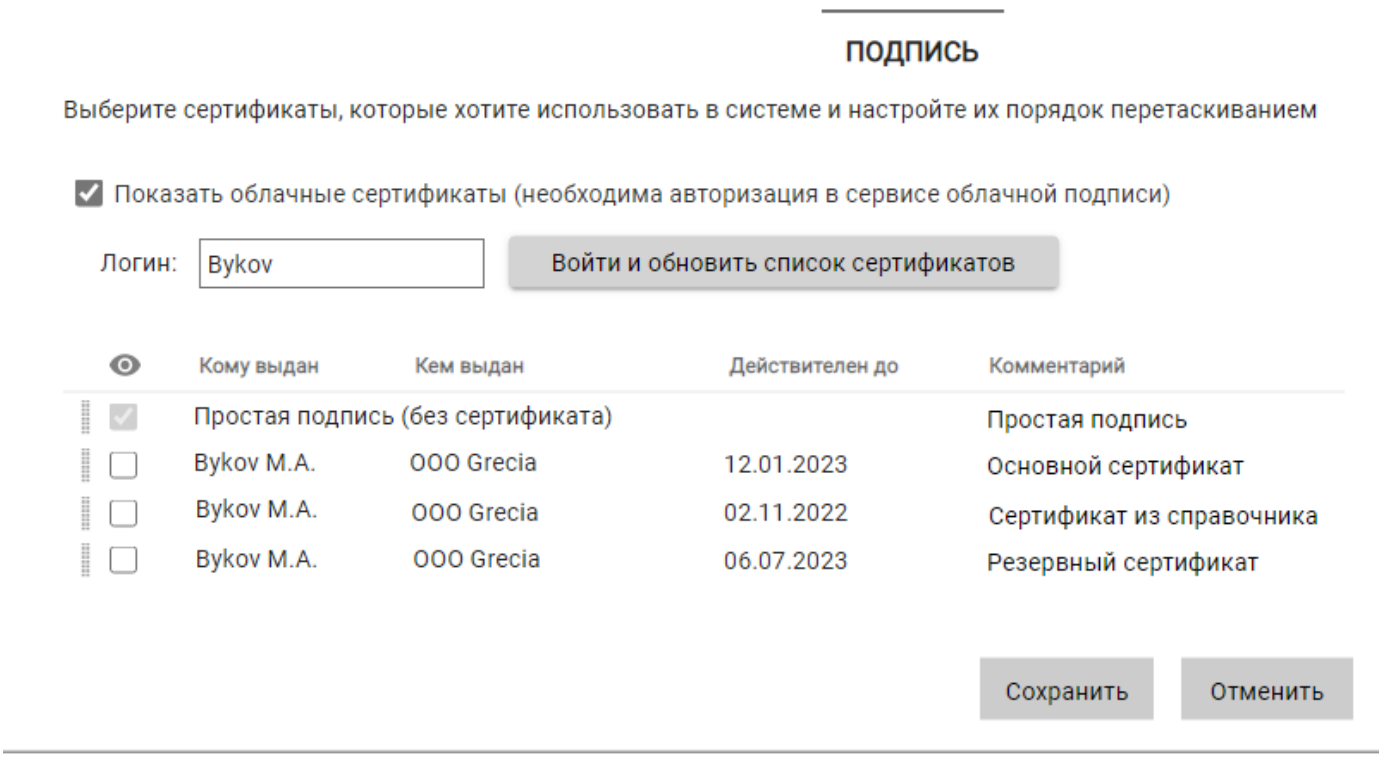


Рисунок 57. Локальные сертификаты на вкладке "Подпись"

На вкладке *Подпись* пользователь может настроить сертификаты подписи, которыми он планирует пользоваться при выполнении заданий на подписание.



Загрузка сертификатов в формате `.pfx` не поддерживается.

Пользователь может управлять локальными сертификатами, установленными на машине пользователя и облачными сертификатами, установленными в хранилище сервиса электронной подписи КриптоПро.

Доступные сертификаты отображаются на вкладке *Подпись*. По умолчанию флаги напротив доступных подписей сняты, то есть сертификаты не отображаются в диалоге подписания. Чтобы сертификаты были доступны для

выбора в диалоге подписания, установите флаги напротив каждого требуемого сертификата.

Для сертификата отображается владелец (фамилия имя), организация, выдававшая сертификат, срок действия сертификата и комментарий. Комментарий — произвольный текст, который помогает идентифицировать сертификат.

Помимо сертификатов, в списке также отображаются:

- ***Простая подпись*** — всегда отображается в диалоге настроек, но может быть недоступна при подписании в зависимости от настроек типа карточки.
- Сертификат, настроенный для пользователя в *Справочнике сотрудников*. Такой сертификат не может быть скрыт из списка.

Порядок следования сертификатов настраивается перетаскиванием:

1. Наведите курсор на строку сертификата.
2. Перетащите строку вверх или вниз по желанию.

Средства для работы с локальными подписями настраиваются на вкладке Установки.



Подробнее работа с компонентами КриптоПро описана в соответствующем разделе документации по администрированию. Работа с облачными подписями описана в разделе Подписание облачной подписью.

Статус учётной записи сотрудника

Статус активности учётной записи сотрудника (активен, болен и др.) может быть изменён в Справочнике сотрудников, а также самим пользователем в Web-клиенте.

Чтобы изменить статус своей учётной записи:

1. Откройте страницу Профиль пользователя.
2. Выберите новый статус учётной записи из раскрывающегося списка.

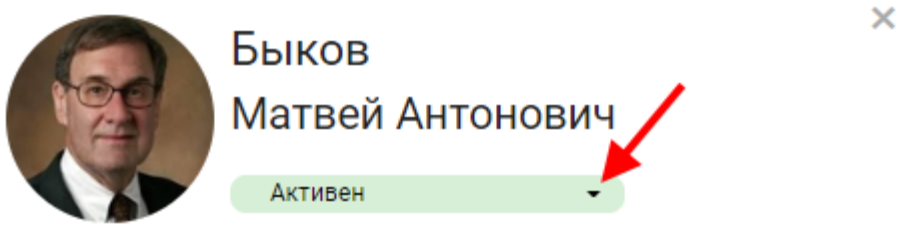


Рисунок 58. Кнопка смены статуса в профиле пользователя

Если выбран статус ***Активен***, список будет выделен зелёным, иначе — серым.

Подробнее о состояниях см. в пункте Состояния сотрудника при работе со справочником сотрудников.

Для отслеживания и изменения статуса на время периода неактивности и возврата к активному статусу по окончанию данного периода используется служебный бизнес-процесс.

3. Нажмите кнопку "Сохранить".

Период отсутствия сотрудника

Пользователь самостоятельно может установить период, в который он будет отсутствовать. Задания сотрудника в этот период будут передаваться заместителям, если таковые назначены.

1. Откройте Профиль пользователя.

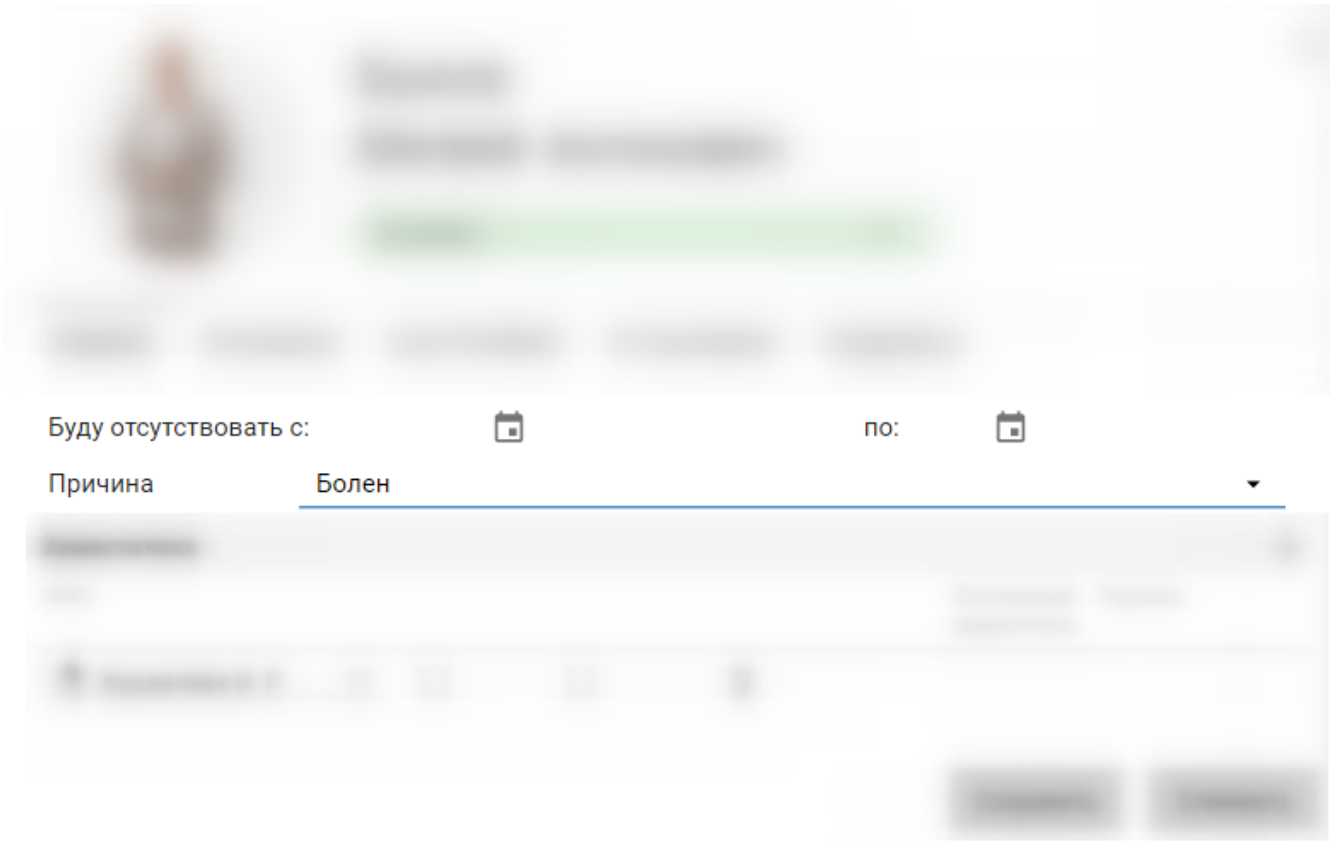


Рисунок 59. Настройки отсутствия сотрудника

2. Укажите начальный и конечный период неактивности в строке *Буду отсутствовать с .. по ….*

> Указанная дата `с ⋯` должна быть раньше даты `по ⋯`. Обе даты должны быть либо позже текущей даты, либо быть пустыми. Установить период неактивности в прошлом нельзя.

3. Выберите причину отсутствия в списке ***Причина***.
4. Нажмите **Сохранить**.

   Если в сроках была допущена ошибка, при сохранении появится предупреждение.

Период отсутствия также можно настроить из справочника сотрудников.

Язык пользовательского интерфейса

При желании язык пользовательского интерфейса Web-клиент можно изменить.

Чтобы изменить язык пользовательского интерфейса:

1. Откройте *Профиль пользователя*.

2. Перейдите в раздел Настройки.
3. Выберите язык интерфейса в списке *Язык интерфейса*.
4. Нажмите кнопку **Сохранить**.

Web-клиент будет перезагружен с выбранным языком пользовательского интерфейса.

Временем обновления данных в разметке профиля пользователя можно управлять при помощи параметров в конфигурационном файле, см. подробнее Кэширование обращений к справочникам и карточкам.

Строка поиска

Строка поиска предназначена для ввода текста поискового запроса. Если полнотекстовый поиск неактивен, поиск осуществляется только по текущей папке. Если полнотекстовый поиск активен, в правой части строки поиска становится доступным выбор области поиска.



Рисунок 60. Строка поиска

По умолчанию строка поиска скрыта. Чтобы её открыть, нажмите кнопку . При неактивном полнотекстовом поиске кнопка поиска отображается только в папках.

Подробнее о полнотекстовом поиске см. полнотекстовый поиск.

Область представления папки

Область представления служит для отображения списка карточек выбранной папки или другого содержимого, установленного настройками папки.

Область представления в полной версии

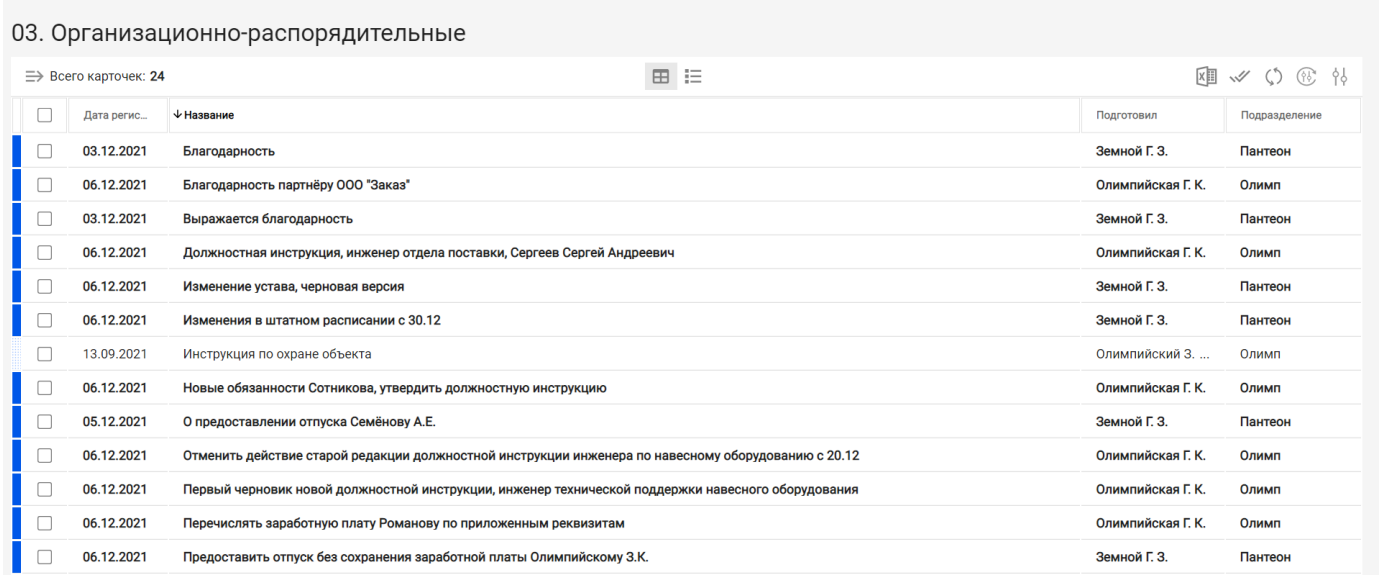


Рисунок 61. Область представления папки в полной версии

Область представления содержит:

Панель инструментов — Описание элементов на панели инструментов см. в разделе Новый грид в представлениях.

Таблица Представления — В таблице отображаются данные карточек папки.

Жирным шрифтом выделяются непрочитанные пользователем карточки. Подробнее см. в пункте Способы оповещения о появлении новых карточек.

Список карточек выводится на экран порциями по 15 карточек. Ограничение может быть изменено администратором.

Область представления в мобильной версии

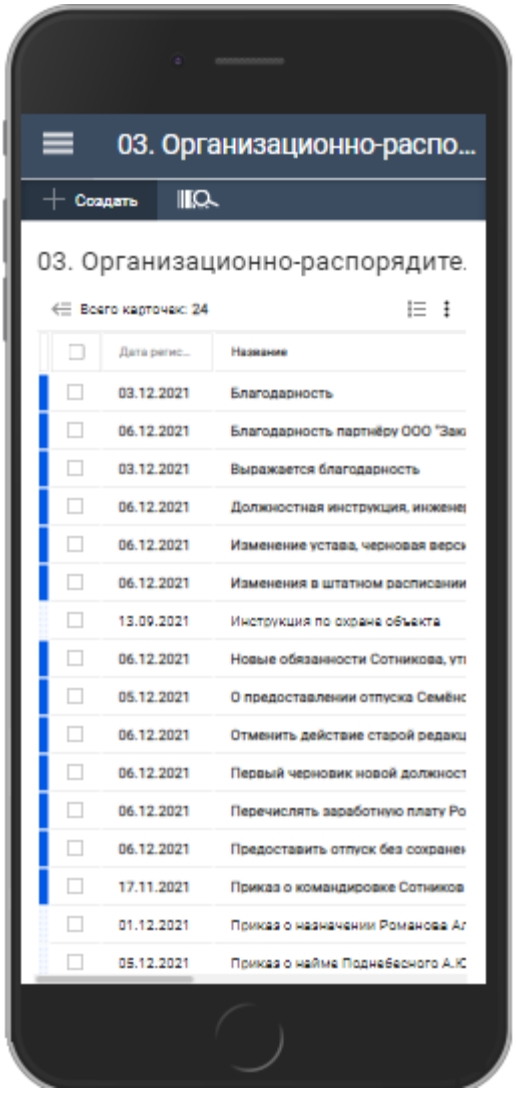


Рисунок 62. Область представления папки в мобильной версии

В мобильной версии представления папки отображаются аналогично полной версии, только в уменьшенном виде.

Подробнее о мобильной версии представления см. в разделе Мобильная версия грида.

Работа с папками

Работа в Web-клиенте

Работа с папками

Описание работы с папками и представлениям.

- Создать новую папку

- Работа с представлениями (грид)
- Экспортировать данные представления в Excel
- Больше информации в навигационном меню слева…

Создать новую папку



Данная функция недоступна в мобильной версии.

Пользователь может настраивать, удалять папку и создавать подпапки, если такая возможность разрешена для соответствующего ЭУ в свойстве "Управление папками".

Свойство поддерживается следующими элементами управления:

- "Папки / Пользовательские папки".
- "Папки / Групповые папки".
- "Папка / Известная папка".

В Web-клиенте можно создавать только простые папки.

Чтобы создать подпапку:

1. В Основном меню наведите курсор мыши на папку, для которой создаётся подпапка.

   Будет показана кнопка ⋮.

   Кнопка скрыта, если функция недоступна пользователю.

2. Нажмите ⋮ **> Создать папку**.

   Будет открыта панель создания папки.

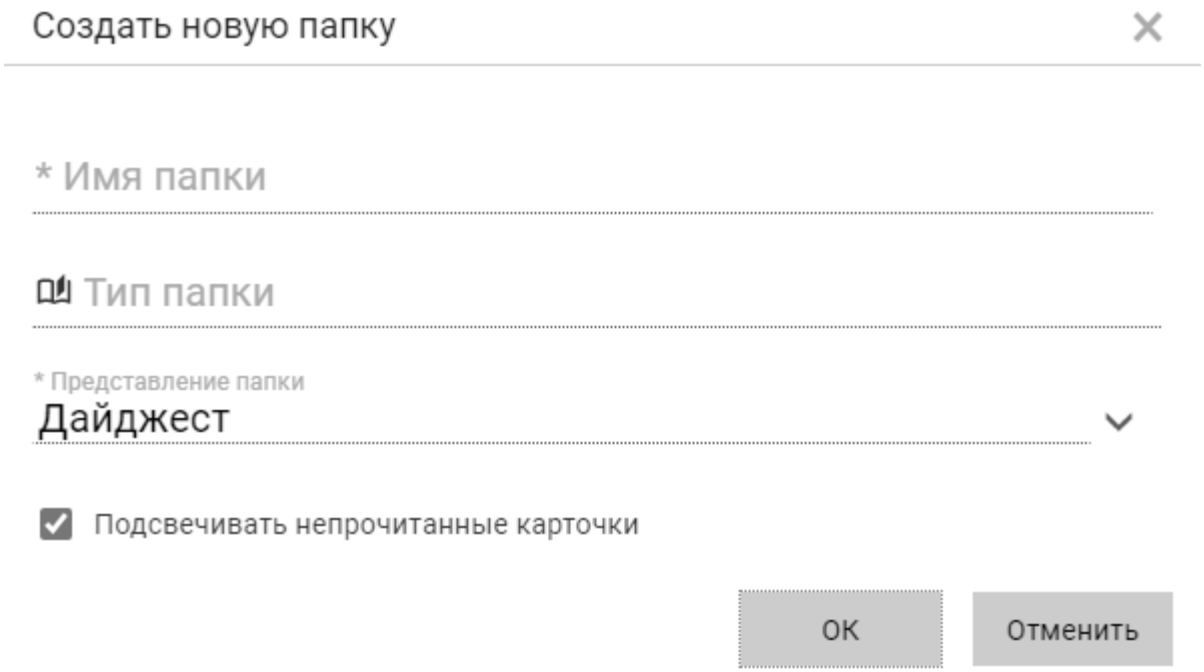


Рисунок 63. Панель создания папки

Если у пользователя отсутствуют права на создание папок, при выборе функции появится предупреждение.

3. Введите *имя папки*.
4. Выберите *тип папки* из Справочника типов папок.

   Новая папка будет создана с настройками типа папок, установленными в *Справочнике типов папок*.

   Данные настройки включают:

   - Стандартное представление папки.
   - Ограничения типов карточек, которые могут быть созданы в папке.
   - Ограничение используемых шаблонов.
   - Включение подсвечивания непрочитанных карточек и др. Полное описание настроек типа папки приведено в документации модуля Платформа.

5. Если требуется, измените используемое по умолчанию представление папки.
6. Если требуется, установите/снимите флаг **Подсвечивать непрочитанные карточки**.

Флаг включает или выключает выделение новых и изменённых карточек в папке, и отображение количества непрочитанных карточек.

7. Нажмите кнопку **ОК**. Для выбранной папки будет создана дочерняя папка с установленными параметрами.

Удалить папку



Данная функция недоступна в мобильной версии.

Пользователь может настраивать, удалять папку и создавать подпапки, если такая возможность разрешена для соответствующего ЭУ в свойстве "Управление папками".

Свойство поддерживается следующими элементами управления:

- "Папки / Пользовательские папки".
- "Папки / Групповые папки".
- "Папка / Известная папка".

Чтобы удалить папку:

1. В Основном меню наведите курсор мыши на удаляемую папку.

   Будет показана кнопка ⋮.

   Кнопка скрыта, если функция недоступна пользователю.

2. Нажмите ⋮ **> Удалить папку**.

   Будет показан диалог подтверждения удаления.

   Если отсутствуют права на удаление папки, при выборе появится предупреждение.

3. Подтвердите удаление папки.

   Папка будет удалена со всеми подпапкам. В зависимости от настроек Web-клиента папка удаляется в Корзину или окончательно.

Изменить настройки папки



Данная функция недоступна в мобильной версии.

Пользователь может настраивать, удалять папку и создавать подпапки, если такая возможность разрешена для соответствующего ЭУ в свойстве "Управление папками".

Свойство поддерживается следующими элементами управления:

- "Папки / Пользовательские папки".
- "Папки / Групповые папки".
- "Папка / Известная папка".

Настройки стандартных папок не могут быть изменены. Используйте пользовательские папки.

Чтобы изменить настройки папки:

1. В основном меню наведите курсор мыши на папку, настройки которой изменяются.

   Будет показана кнопка ⋮.

   Кнопка скрыта, если функция недоступна пользователю.

2. Нажмите ⋮ **> Настройки папки**.

   Будет открыта панель настроек папки.

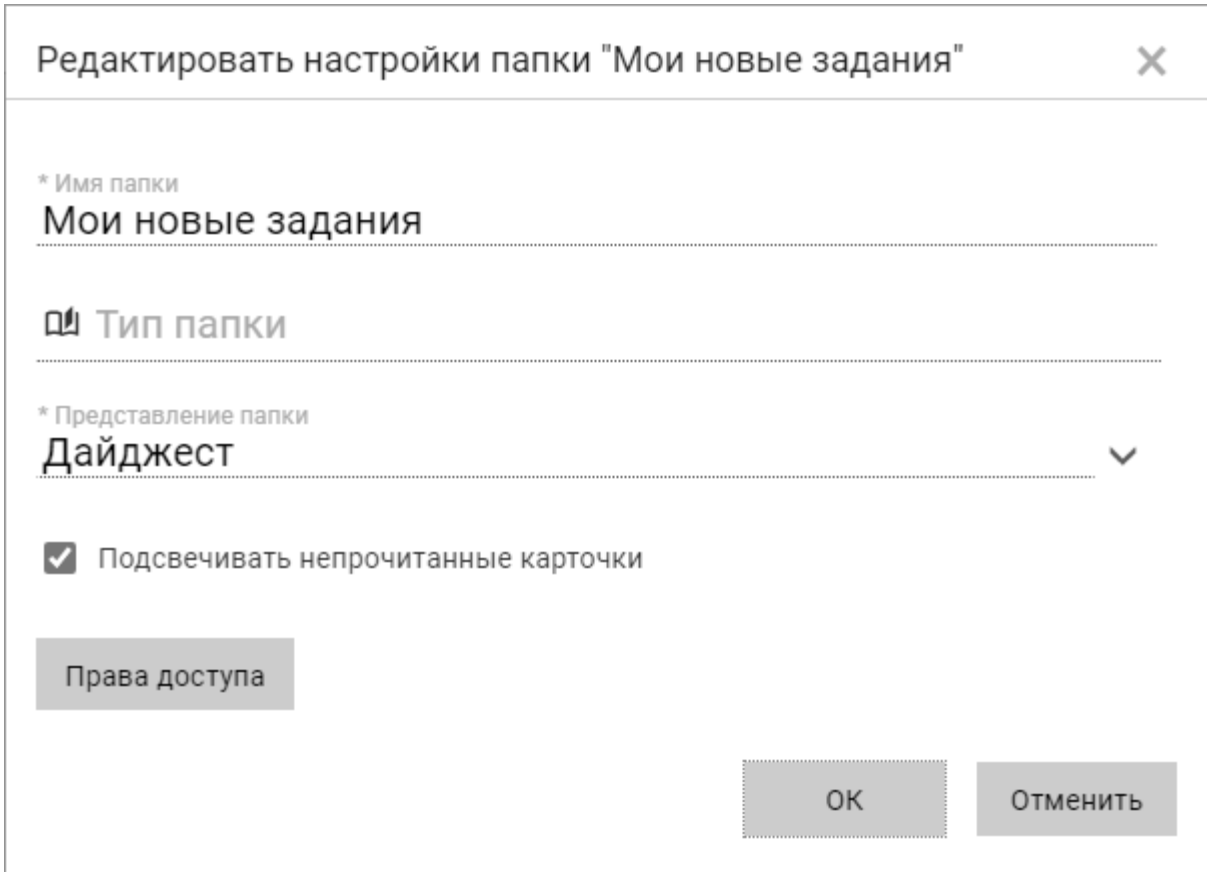


Рисунок 64. Панель настроек папки



При отсутствии прав на изменение папки, панель настроек будет открыта в режиме чтения (без возможности изменения настроек).

3. Измените требуемые настройки папки: Имя папки, Тип папки (см. Создание новых папок) и представление по умолчанию.
4. Нажмите кнопку **Права доступа**, если нужно изменить настройки безопасности папки.
5. Нажмите кнопку **OK** для сохранения изменений.

Назначить права доступа к папке

Пользователь Web-клиента может ограничить доступ других пользователей к папке[1] с помощью дискреционной модели безопасности.

Чтобы установить права доступа:

1. В Основном меню наведите курсор мыши на выбранную папку.

Будет показана кнопка ⋮.

Кнопка скрыта, если функция недоступна пользователю.

2. Нажмите кнопку ⋮ **> Права доступа**.

Будет открыта панель настройки прав доступа.

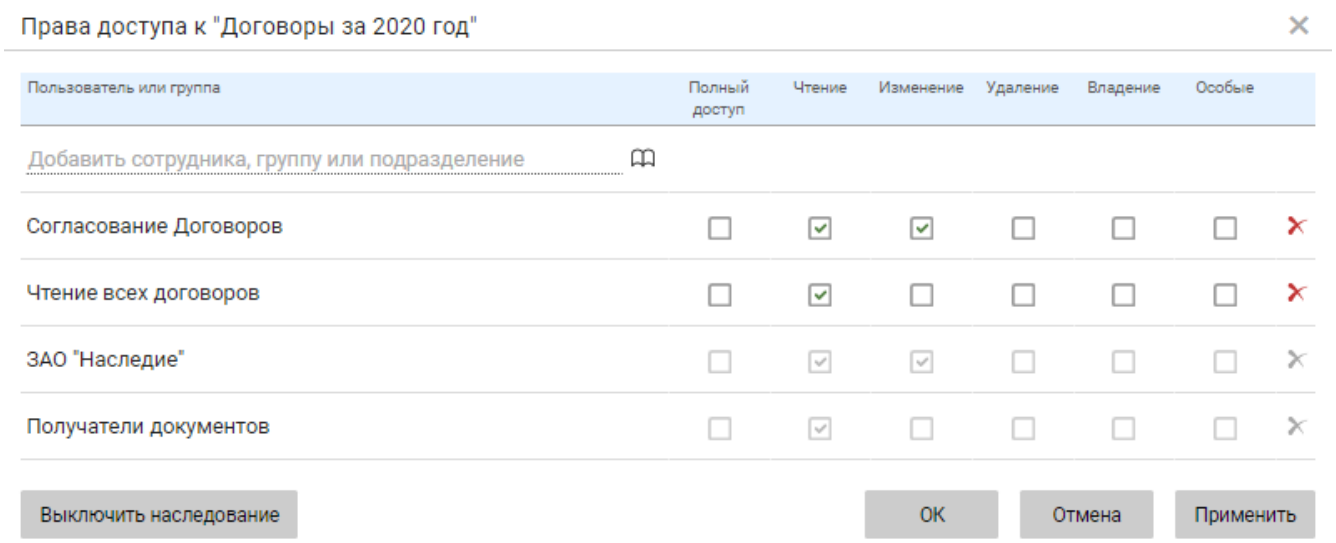


Рисунок 65. Панель настройки прав доступа

Если у пользователя отсутствуют права на владение папкой, возможность изменения прав доступа будет недоступна.

Панель настройки прав доступа также можно открыть из панели настроек папки, если у пользователя есть права на изменение папки.

В появившемся окне можно установить права доступа, например:

- Изменить существующие права доступа.
- Предоставить требуемый уровень доступа сотрудникам, группам сотрудников и/или подразделениям.
- Отключить или включить наследование.

Инструкция по работе с панелью настройки прав доступа приведена в пункте Настройка безопасности объектов.

После завершения настройки нажмите кнопку **ОК** или **Применить**, чтобы сохранить изменения. Чтобы отменить изменения, нажмите кнопку **Отмена**.

Общие операции с папками

Выбрать папку

Чтобы выбрать папку, кликните по её названию в "Основном меню" или "Дашборде".

В зависимости от типа папки, будет открыт список карточек папки, окно настройки параметрического поиска или другая страница.

Обновить содержимое папки

Содержимое открытой папки не обновляется автоматически. При необходимости можно настроить автоматическое обновление содержимого папки, подробная инструкция описана в руководстве пользователя Windows-клиента, раздел "Обновление представления папки".

Чтобы обновить содержимое папки:

1. Наведите курсор мыши на панель управления грида.
2. Обновите содержимое папки нажатием кнопку .

Содержимое папки также обновляется при возврате в текущую папку из другой.

Подключить пользовательские папки

Данная функция недоступна в мобильной версии.

Пользователь может добавить в дерево папок необходимые папки, на которые у него есть права доступа.

Подключение пользовательских папок

Чтобы подключить пользовательские папки:

1. Нажмите на кнопку в Основном меню.

Будут открыты настройки папок основного меню.

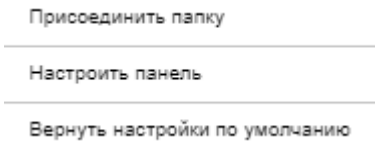


Рисунок 66. Настройка папок основного меню

2. Выберите в меню пункт *Присоединить папку*.

   Появится окно со списком доступных папок.

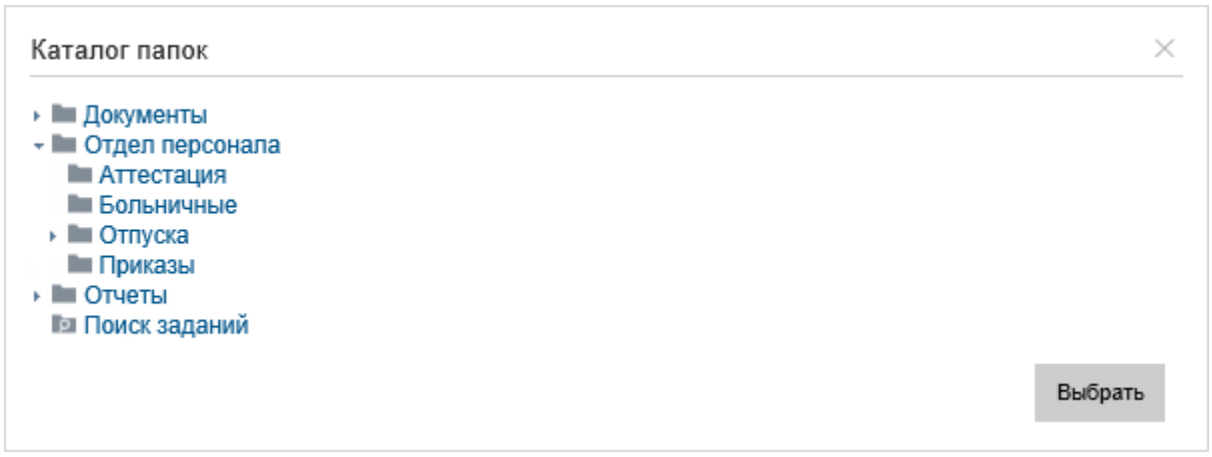


Рисунок 67. Выбор из доступных папок

3. Выберите одну или несколько папок, которые нужно подключить.

   Для выбора нескольких папок удерживайте кнопку `Ctrl` или `Shift`.

4. Нажмите кнопку **Выбрать**.

   Выбранные папки будут добавлены в раздел *МОИ ПАПКИ* дерева папок.

Видимость дочерних папок

Видимость дочерних папок зависит от типа подключенной родительской папки:

- Дочерние папки стандартных (значок) и виртуальных (значок) папок всегда отображаются.
- Дочерние папки папки-делегата (значок) отображаются, если в настройках папки включён режим *Показывать подпапки*.

Отображение внешней страницы в папке

Если подключаемая папка настроена на открытие внешней страницы вместо представления (см. документацию модуля Windows-клиент, раздел "Ссылка"), должен выполняться ряд условий:

1. Если папка настроена на открытие отчёта службы Microsoft SQL Server **2016** Reporting Services, в URL-адрес отчёта в настройках папки нужно добавить параметр `rs:Embed=true`. Например, `http://CompanyReportServer.com/Reports/DocumentsReport?rs:Embed=true`.

   Если используется Microsoft SQL Server с версией ниже 2016, добавлять указанный параметр не нужно.

   > Для доступа к отчёту используются данные пользователя, от имени которого запущен браузер. А не того, от чьего имени выполнен вход в Web-клиент.

2. Сервер SQL Server Reporting Services и сервер Web-клиент должны находиться в одном домене.

   > Подключаемая страница должна разрешать встраивание. Открытие папки со страницей, встраивание которой запрещено, завершится с ошибкой `Refused to display 'Site' in a frame because it set 'X-Frame-Options' to 'DENY'`.

3. После изменения URL-адреса в настройках папки нужно перезапустить службу Сервера Web-клиента и повторно зайти в Web-клиент.

   > Если подключенная папка содержала подпапки-делегаты, которые были удалены, нужно отключить и повторно подключить папку, чтобы обновить структуру папок.

Отключить папки

> Данная функция недоступна в мобильной версии.

Чтобы отключить видимость определённых папок:

1. Нажмите на кнопку ⚙ в Основном меню.

   Будут открыты настройки папок основного меню.

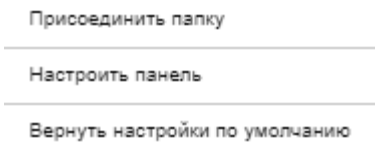


Рисунок 68. Настройка папок основного меню

2. Выберите в меню пункт *Настроить панель*.

> Слева от названий папок появятся флаги управления подключенными папками.

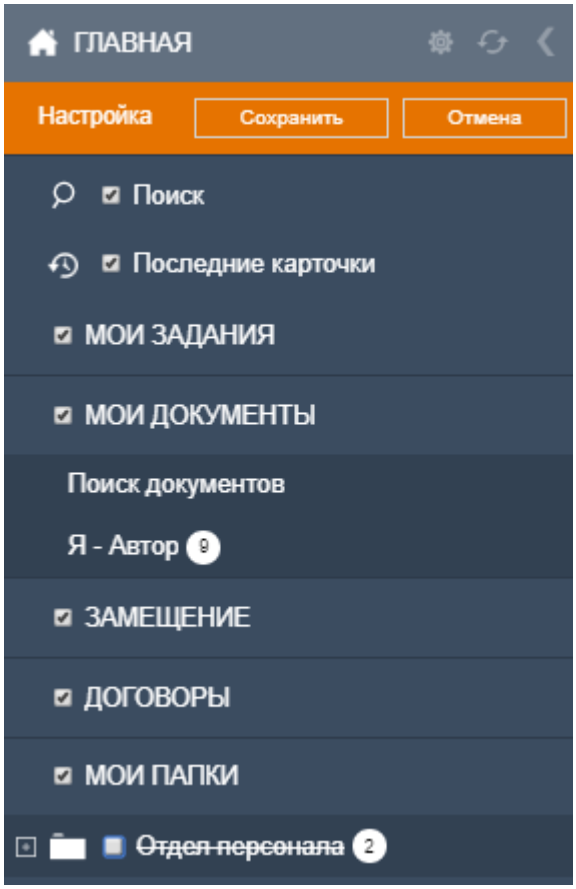


Рисунок 69. Управление подключенными папками

3. Снимите флаги с папок, которые нужно отключить.
4. Нажмите кнопку **Сохранить**, чтобы применить изменения.

> Папки, подключенные пользователем, будут отключены. Стандартные папки будут скрыты.

Обновить системные настройки

С целью оптимизации на стороне пользователя Web-клиента сохраняются некоторые настройки папок: используемое представление, поисковый запрос и др.

Если данные настройки были изменены, но изменения не отразились в Web-клиенте, нажмите кнопку  в основном меню для принудительного обновления настроек.

Сбросить пользовательские настройки

Web-клиент сохраняет и применяет при следующем открытии пользовательские настройки папок:

- Выбранное представление.
- Настройки фильтрации и сортировки.
- Номер страницы представления.
- Ширина колонок.
- Способ переноса текста в строках представления.
- Результаты поиска по представлению.

Чтобы сбросить пользовательские настройки папки:

1. Наведите курсор мыши на строку с названием папки.

   Справа от названия появятся кнопки обновления и сброса настроек.

2. Нажмите кнопку .

   Настройки папки будут сброшены на стандартные.

Восстановить стандартный список папок



Данная функция недоступна в мобильной версии.

При восстановлении стандартного списка папок все стандартные папки, скрытые пользователем, будут снова показаны. Восстановление не распространяется на пользовательские папки.

Чтобы восстановить стандартный набор папок:

1. Нажмите на кнопку ⚙ в Основном меню.

   Будут открыты настройки папок основного меню.

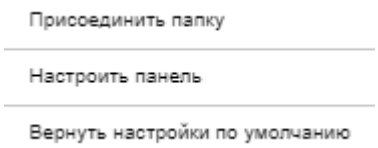


Рисунок 70. Настройка папок основного меню

2. Выберите в меню пункт *Вернуть настройки по умолчанию*.

   Состав стандартных папок вернется к начальному состоянию.

Оповещение о появлении новых карточек

Web-клиент использует несколько способов оповещения пользователя о появлении новых карточек: созданных и изменённых другими пользователями.

Оповещение в сообщении

Если открытой папке изменились данные, будет показано сообщение с текстом:

- *В текущей папке есть новые непрочитанные карточки!*

  Если в папке изменилось количество непрочитанных карточек.

- *Отображаемые данные устарели, пожалуйста, обновите представление.*

  Если изменилось количество карточек в папке.

При появлении таких сообщений нужно обновить содержимое папки, чтобы увидеть изменения.

Счетчик непрочитанных карточек

Количество непрочитанных карточек папки (новых, изменённых другими пользователями) показывается в дереве папок.

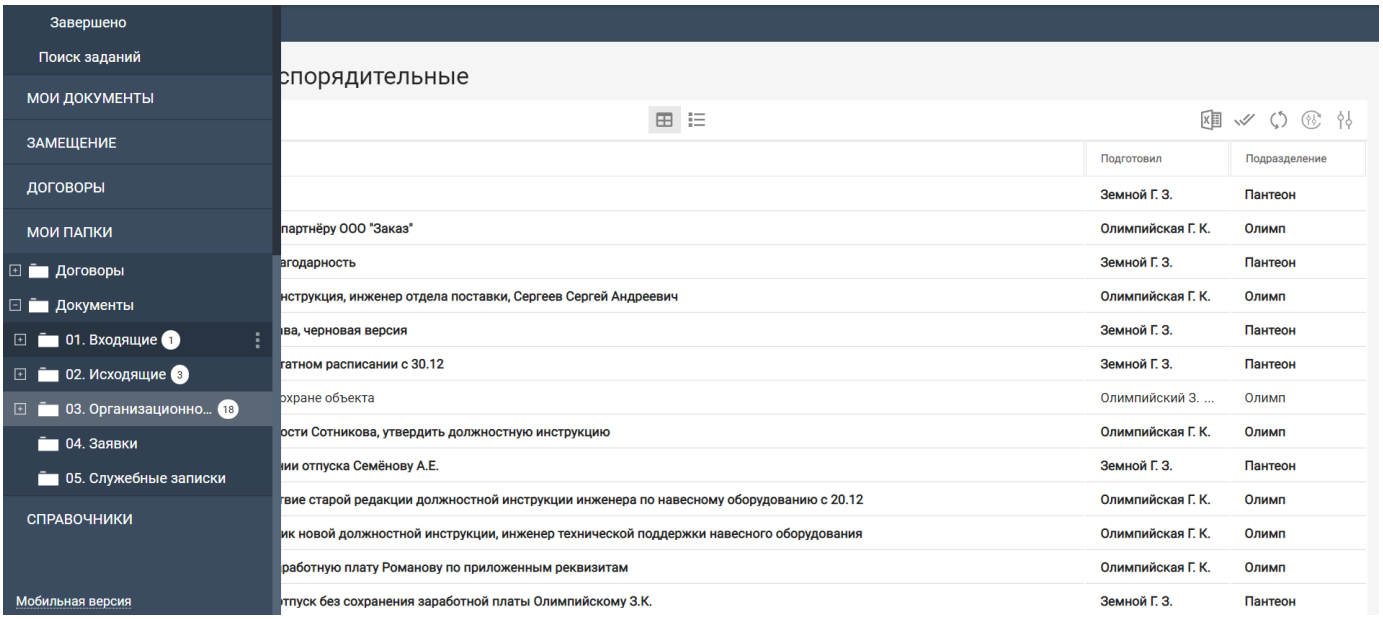


Рисунок 71. Счетчик непрочитанных карточек папки

Счетчик непрочитанных карточек в папке автоматически обновляется в фоновом режиме, а также при обновлении содержимого папки пользователем.

Web-клиент не будет показывать и обновлять счетчик непрочитанных карточек папки если в настройках папки:

- Выключена подсветка непрочитанных карточек.
- Включена постраничная проверка прав доступа.

Выделение непрочитанных карточек

Web-клиент выделяет непрочитанные карточки жирным шрифтом.

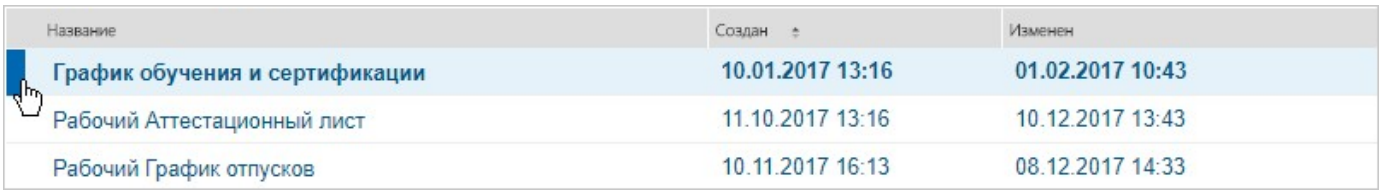


Рисунок 72. Отображение непрочитанной карточки

После открытия карточки шрифт станет обычным.

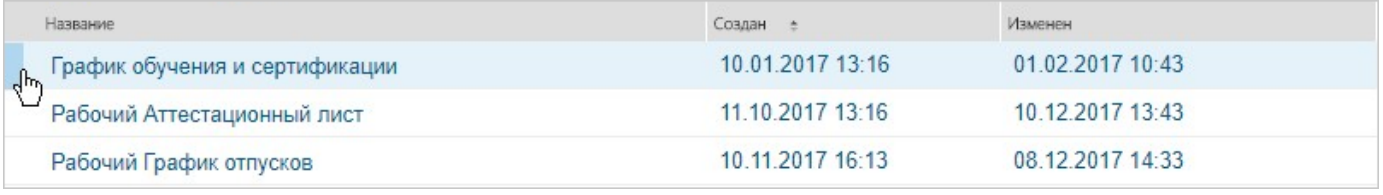


Рисунок 73. Отображение прочитанной карточки "Документ"

Web-клиент не будет выделять непрочитанные карточки если в настройках папки:

- Выключена подсветка непрочитанных карточек.
- Включена постраничная проверка прав доступа.
- Отключена проверка прав доступа.

Изменить признак непрочитанности

Чтобы изменить вручную признак непрочитанности:

1. Наведите курсор мыши на строку с данными карточками. Слева от названия карточки отобразится дополнительный блок (см Рисунок 2 и Рисунок 3).
2. Щелкните по блоку, чтобы изменить признак непрочитанности.

На все способы оповещения влияет значение настройки папки *Автоматическое обновление* (см. документацию *Windows-клиента*, раздел "Обновление представления папки"). Если автоматическое обновление выключено, папка не будет проверяться на наличие изменений, оповещения об обновлении не будут формироваться.

Работа с представлениями (грид)

Новый грид стал доступен для всех пользователей на всех устройствах. Грид имеет два режима отображения: таблица (используется по умолчанию) или лента. Переключить режим можно кнопками  на панели управления.

Новый грид заменяет старый вид представления. Пользовательские групповые операции, созданные для старого грида, будут корректно работать в новом.

Все пользовательские настройки грида остаются доступными даже после того, как пользователь вышел из своей учётной записи и зашёл вновь.

Настройки сделанные текущим пользователем сохраняются только для текущего пользователя и не распространяются на других.

Пользовательские настройки грида сохраняются для текущего пользователя в текущей папке с текущим представлением до тех пор, пока не будут очищены нажатием на кнопку **Сбросить настройки** на панели управления.

Новый грид поддерживает прозрачность. Прозрачность регулируется свойством

Непрозрачность элемента управления Карточки папки. По умолчанию значение свойства **Непрозрачность** равняется 100%, то есть грид полностью непрозрачен. При желании значение может быть изменено.

Функциональные возможности нового грида

Новый грид предоставляет функции настройки представления разметок. Функции можно вызвать с панели инструментов или из контекстного меню. На компьютерах контекстное меню вызывается правой кнопкой мыши, на планшетах — долгим касанием.

Строки

Каждая строка грида — отдельная карточка. Перед каждой строкой отображается флаг, который можно установить. Если флаг установлен, значит строка (карточка) выбрана.

Если установить флаг в заголовке колонки с флагами, будут выбраны все карточки в папке.

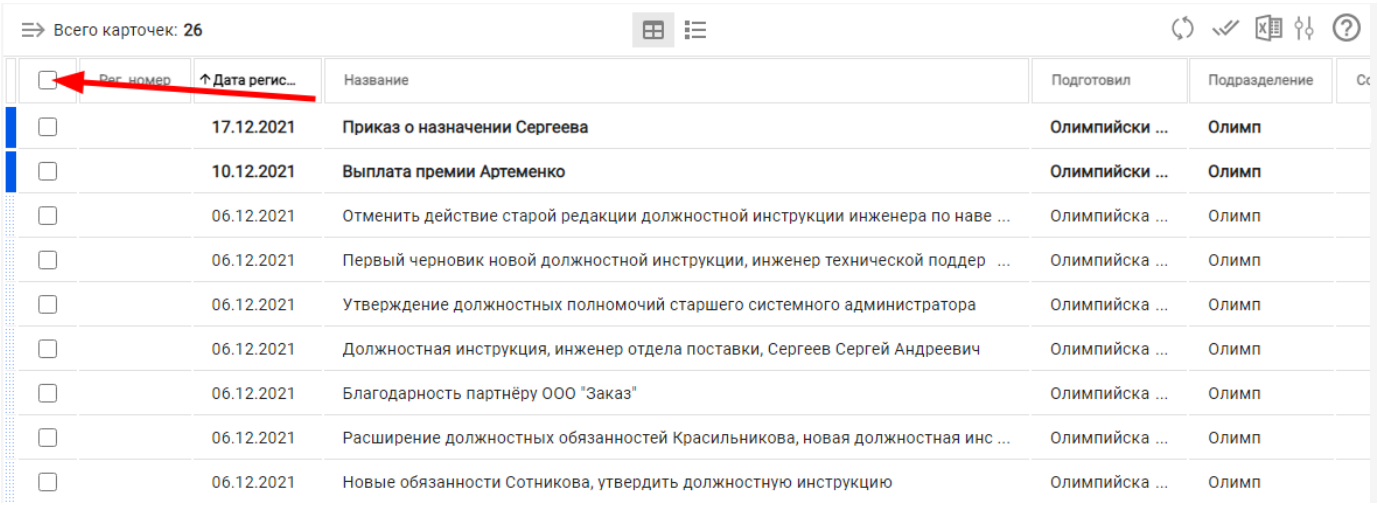


Рисунок 74. Выбрать все карточки

По умолчанию непрочитанные карточки выделяются синим цветом. Администратор может настроить свой цвет или добавить дополнительные цвета для других состояний. Инструкция по настройке цвета представления приведена в руководстве по настройке Docsvision.

Чтобы отметить карточку прочитанной, нажмите на синий индикатор слева от заголовка строки. Нажмите на индикатор повторно, чтобы снова отметить карточку как непрочитанную.

Прокрутка

Карточки в новом гриде загружаются на одной странице порциями. При прокрутке списка карточек вверх или вниз появляются кнопки навигации. При нажатии на кнопку список карточек будет прокручен до самой верхней или самой нижней строки соответственно.

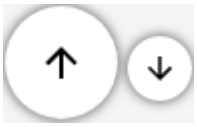


Рисунок 75. Кнопки навигации вверх-вниз

При наведении курсора на область прокрутки, отображается количество карточек, отображаемое на странице в формате `N – X` (например, `3 – 22`). N и X — первая и последняя строки, отображаемые на экране.



Рисунок 76. Отображение строк в области прокрутки

Выделение строк цветом

Если в РМА настроено выделение строк представления цветом, строки грида также будут выделяться цветом.

Непрочитанные карточки выделяются цветом без дополнительных настроек.

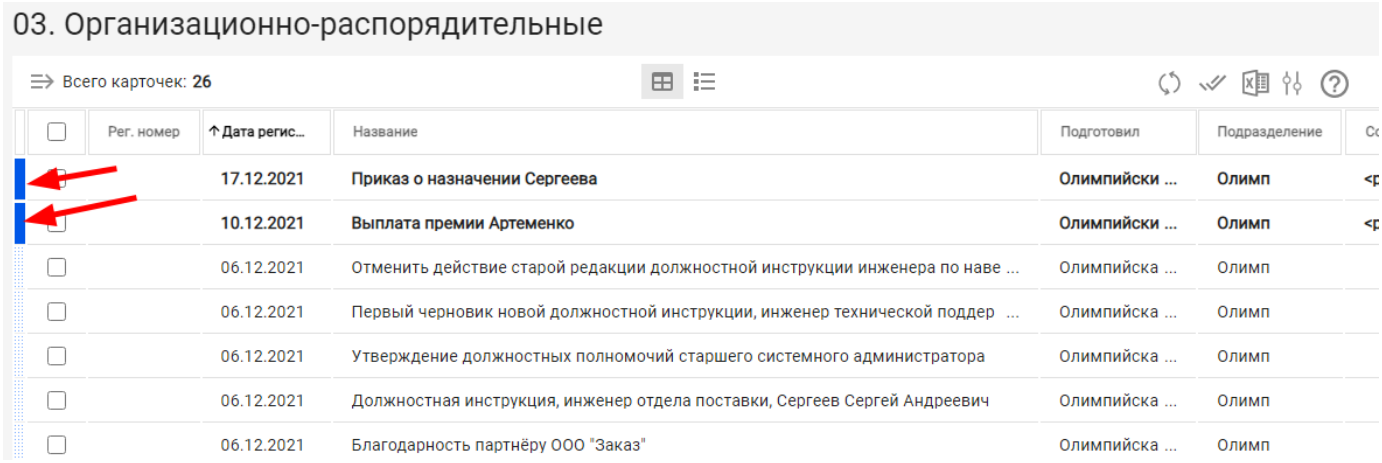


Рисунок 77. Выделение непрочитанных карточек

Колонки

Информация о карточке (тип, название, автор, статус и т.д.) отображается в

отдельных колонках. Наведите курсор мыши на правую границу колонки, чтобы изменить её ширину.



Рисунок 78. Регулировка ширины колонки "Подготовил"

В версии для планшетов изменить размер колонки можно долгим удержанием пальца на её границе.

Дробные и целые числовые значения в таблице грида выравниваются по правому краю колонки.

Регистратор	Стоимость
Олимпийский З.К.	8 500,00
Олимпийский З.К.	1 985,90

Рисунок 79. Выравнивание числовых значений

HTML-код в ячейках грида будет отображаться в виде строки кода или с форматированием в зависимости от настроек представления в PMA.

Автоматическое обновление папки

При необходимости можно настроить автоматическое обновление содержимого папки, подробная инструкция описана в руководстве пользователя Windows-клиента, раздел "Обновление представления папки". Если в папке включено автообновление данных, вы можете продолжить работать с отображаемыми данными или обновить представление, чтобы показать новые карточки. Если в папке появились новые непрочитанные карточки, появляется следующее уведомление:

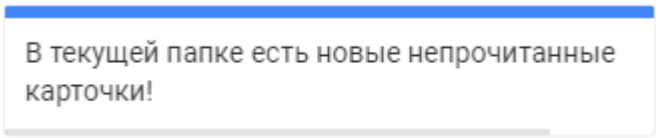


Рисунок 80. Есть новые непрочитанные карточки

Панель инструментов

Панель инструментов грида (toolbar) предоставляет следующие функции:

- Группировка и индикация количества карточек.
- Обновить данные представления.
- Переключение режима грид — лента.
- Прочитать все карточки в папке.
- Экспорт представления в Excel.
- Изменить пользовательские настройки, в том числе:
  - Используемое представление.
  - Отображение и порядок колонок.
  - Перенос или обрезание строк.
  - Сбросить пользовательские настройки.

Группировка

Если перетащить заголовок колонки на панель инструментов, содержимое колонки будет сгруппировано.

Фильтрация

Фильтр можно применить нажатием кнопки в заголовке колонки.

Сортировка

Сортировка в гриде осуществляется по данным колонок в алфавитном порядке, от меньшего числа к большему. Порядок может быть инвертирован нажатием на заголовок колонки.

Групповые операции

Новый грид поддерживает работу с групповыми операциями.

Доступные групповые операции нового грида:

- Прочитать.
- Копировать ярлык карточки.
- Переместить ярлык карточки.
- Удалить ярлык карточки.
- Групповое делегирование.

Панель групповых операций становится активной, когда установлен флаг слева от заголовка карточки. Если групповых операций слишком много, панель можно

минимизировать кнопкой ^.



В браузере Internet Explorer панель скрывается не до конца.

Количество карточек

В папках может отображаться количество карточек. В обычных условиях отображается абсолютное число карточек в папке. Например, 100. Если включена постраничная проверка прав в представлении, отображается абсолютное число карточек с плюсом. Например, 100+. Если в представлении включён фильтр или группировка, отображается число сгруппированных карточек. Фильтрация, группировка и постраничная проверка прав могут быть скомбинированы.

Если в папке настроено автоматическое обновление и система определяет, что в папке есть новые карточки, выводится сообщение "В текущей папке есть новые непрочитанные карточки!". Можно обновить содержимое папки, чтобы показать новые данные, или продолжить работу без обновления.

Контекстное меню

При нажатии правой кнопкой мыши в режиме ПК и долгом нажатии пальца в режиме планшета в области грида вызывается контекстное меню. Контекстное меню доступно, когда активно свойство Контекстное меню для ЭУ Карточки папки.

Функции контекстного меню, дублирующие функции панели инструментов будут недоступны, если отключены в настройках ЭУ "Карточки папки".

Функции, доступные из контекстного меню таблицы:

- *Отфильтровать по значению* — функция доступна при нажатии правой кнопкой мыши (долгом касании) в конкретной ячейке. Работает как фильтр, введённый вручную и применяется автоматически.
- *Сгруппировать по столбцу* — применяет группировку по колонке, из которой было вызвано контекстное меню.
- *Скопировать адрес ссылки* — копирует адрес ссылки для карточки, в строке которой было вызвано контекстное меню.
- *Скопировать текст ячейки* — копирует текст ячейки, по нажатию на которую было вызвано контекстное меню.
- *Открыть* — открывает карточку, на строке которой было вызвано контекстное меню.

- *Открыть в новой вкладке* — в новой вкладке открывает карточку, на строке которой было вызвано контекстное меню.
- *Открыть в новом окне* — в новом окне открывает карточку, на строке которой было вызвано контекстное меню.

Поиск в папках

Новый грид поддерживает поиск в папках. Если настроен полнотекстовый поиск, доступны следующие области поиска:

- В текущей папке.
- В текущей папке и подпапках.
- В найденном.
- Везде.

В некоторых локациях количество доступных областей поиска может быть ограничено.

При определённых условиях в гриде может отображаться область агрегации.

Справка по использованию грида

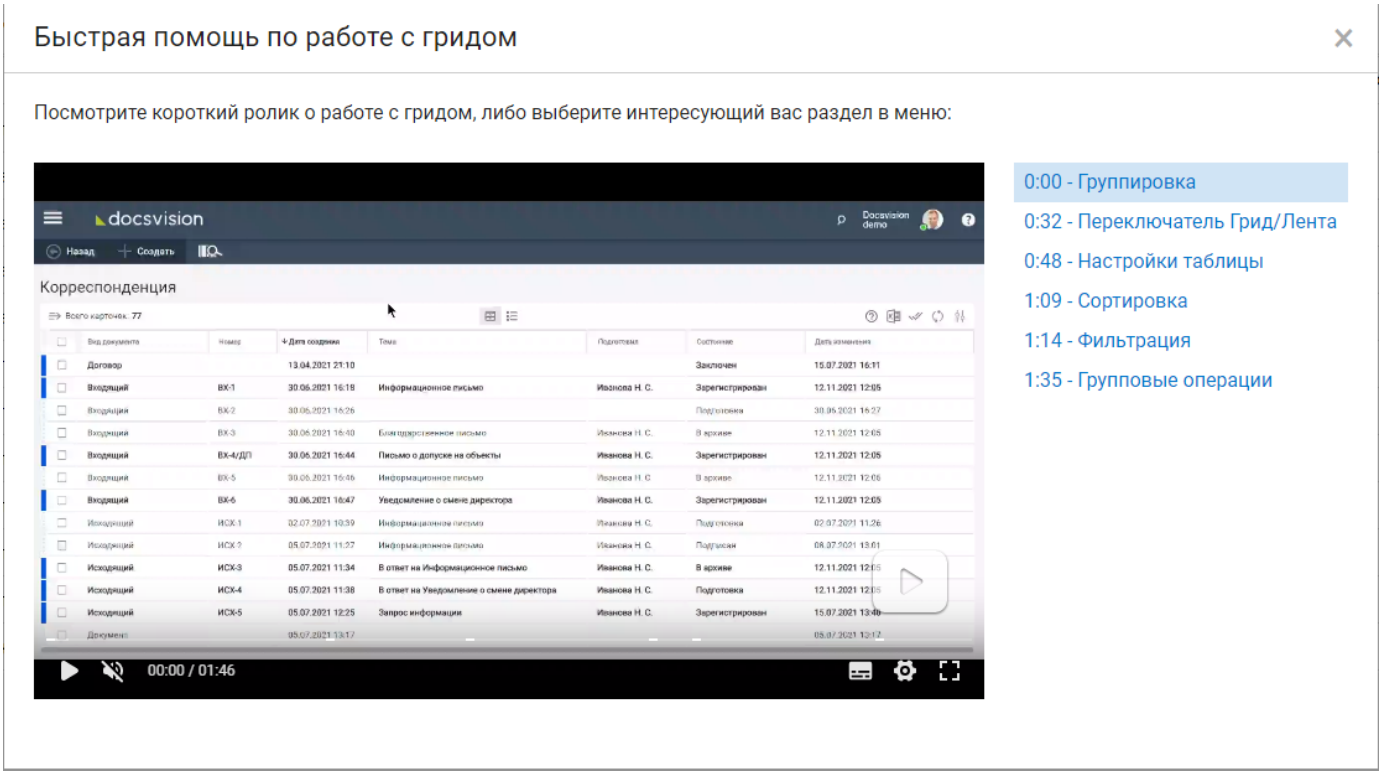


Рисунок 81. Справка по использованию грида

При нажатии на кнопку ⓘ на панели управления открывается ролик со

справкой по использованию грида. В ролике описываются механизмы взаимодействия пользователя с гридом.

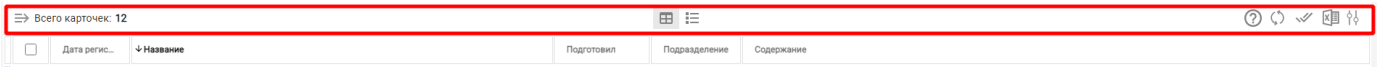
Панель инструментов нового грида

Панель инструментов отображает настройки нового грида, доступные пользователю. Настройки применяются и сохраняются для каждого пользователя индивидуально и сохраняются для текущего пользователя в текущем представлении текущей папки. Не все настройки могут быть доступны пользователям. Доступность настроек зависит от того, установлены ли свойства в ЭУ "Тулбар папки".

Панель инструментов располагается над заголовками колонок и растянута на всю ширину таблицы грида.

Панель инструментов по умолчанию содержит стандартные кнопки, описанные ниже, но также может содержать кнопки для команд, добавленных администратором.

Панель инструментов нового грида



Группировки и размер папки

Крайняя левая кнопка на панели инструментов позволяет отобразить или скрыть дополнительную область, позволяющую применить группировки.

В области "Размер папки" отображается общее количество карточек в папке в виде `Показано X`.

Если включены фильтр и/или группировка, в папке отображается количество карточек в виде: `Показано N из X`, где X — общее количество карточек в папке.

Если включена постраничная проверка прав, в папке отображается количество карточек в виде: `Показано N из X+`, где X — количество загруженных карточек.

Кнопка **Показать все** отображает все карточки в папке. Нажатие на кнопку не сбрасывает группировку, но сбрасывает фильтр.



Рисунок 82. Кнопка отображения области группировки и размер папки

Справка по использованию грида

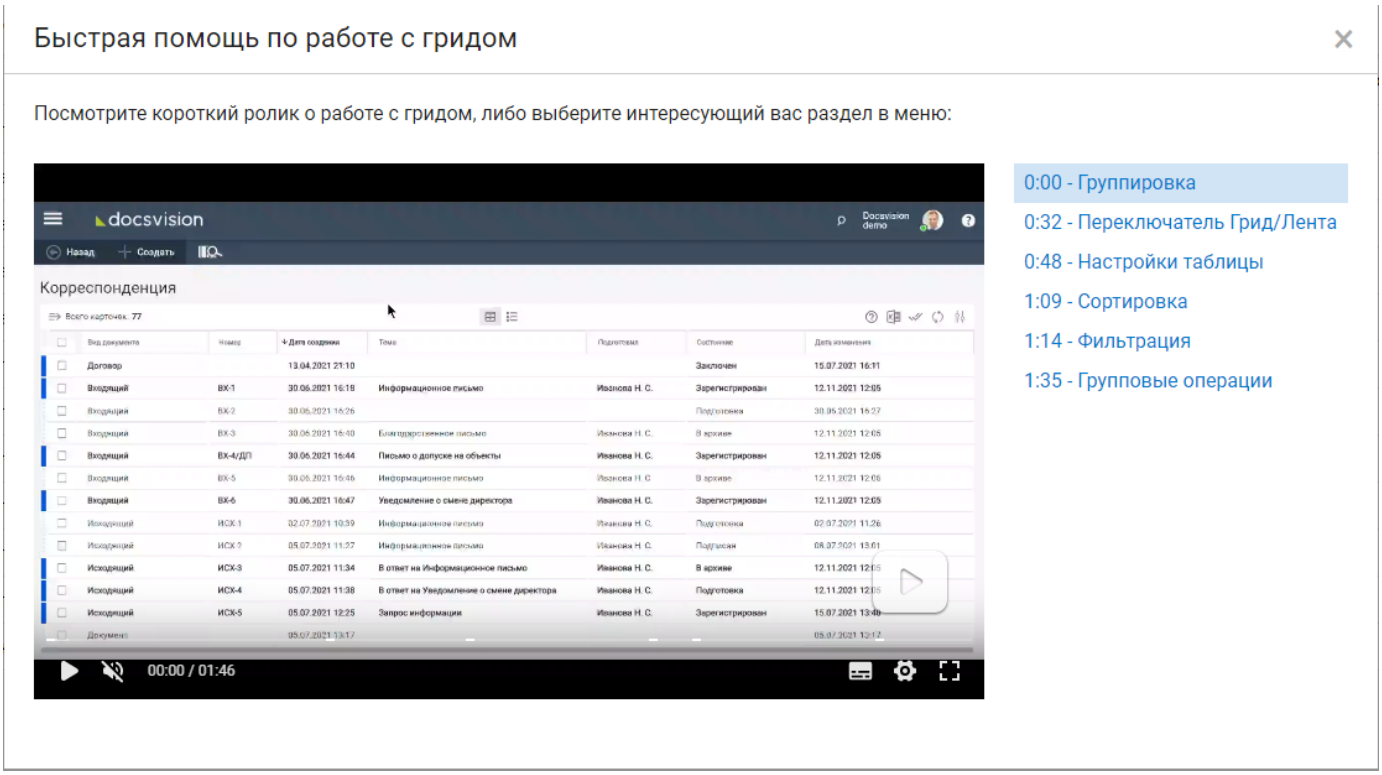


Рисунок 83. Справка по использованию грида

При нажатии на кнопку на панели управления открывается ролик со справкой по использованию грида. В ролике описываются механизмы взаимодействия пользователя с гридом.

Кнопка отображается во всех представлениях и не затрагивается командой "Сбросить настройки".

Обновить

Кнопка для обновления содержимого таблицы: . Нажатие на кнопку посылает запрос данных с сервера. Если в папке появились новые карточки, они будут отображены в представлении.

Лента — грид

Кнопки в середине панели инструментов являются переключателем режим отображения представления: . Для устройств с большим экраном (компьютер, ноутбук, планшет) по умолчанию выбирается режим "Грид" (левая кнопка), для мобильных телефонов — "Лента".

Прочитать всё

Кнопка прочитать всё — позволяет прочитать все карточки в папке одним нажатием кнопки.

Экспорт в Excel

Кнопка позволяет экспортировать представление таблицы с данными в Excel.

При экспорте будут выгружены все страницы представления (не только текущая). Если в представлении применена группировка, фильтрация и т.д. в Excel будут экспортированы текущие отображаемые на экране данные с учётом применённых настроек.

Следующие настройки представления будут включены в экспортируемый файл:

- Установленная сортировка.
- Условия фильтрации (экспортируются только отфильтрованные строки).
- Группировка.
- Правила заливки.
- Прочие пользовательские настройки.

> Сформированный файл `.xlsx` загружается на компьютер пользователя в стандартный каталог загрузки веб-браузера.
>
> Название файла соответствует названию представления, для которого была вызвана функция экспорта данных.

Если пользователь сбрасывает настройки представления, все пользовательские настройки представления будут отменены.

Настройки

Кнопка позволяет настроить внешний вид таблицы грида: представление, видимость и расположение колонок, перенос строк. Изменить возможности окна настроек через программу Конструктор Web-разметок нельзя. Все настройки сохраняются только для текущего пользователя в текущей папке с текущим представлением.

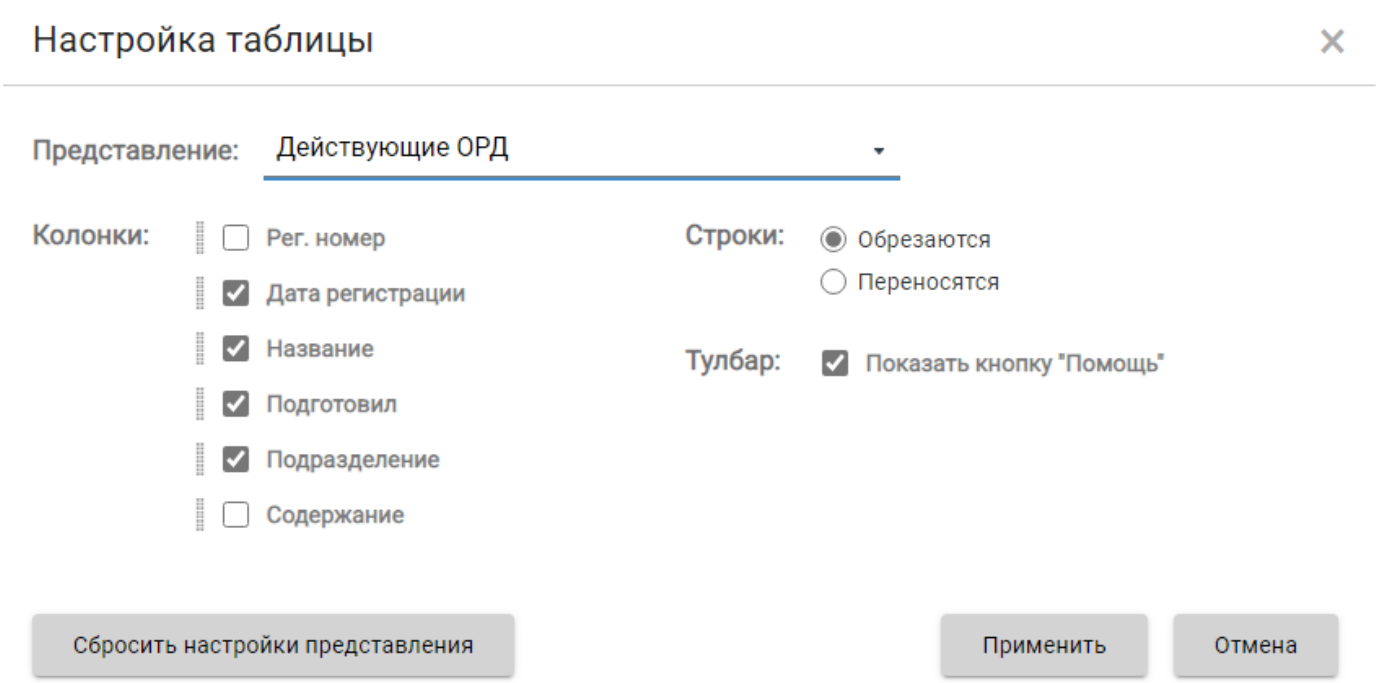


Рисунок 84. Окно настроек грида

Представление

По умолчанию используется представление настроенное в платформе для данной папки. Если представление не настроено, по умолчанию используется дайджест. Выбор доступных представлений определяется настройками Windows-клиент или PMA в свойствах данной папки.

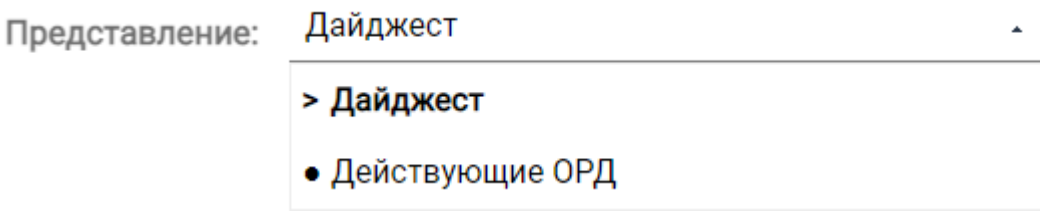


Рисунок 85. Настройки представления в гриде

Представление по умолчанию отмечено маркером. Представление, выбранное пользователем, отмечено жирным шрифтом и знаком >.

Колонки

По умолчанию для выбранного представления выводятся все колонки (установлены все флаги). Чтобы скрыть колонку, снимите флаг слева от нужного названия.

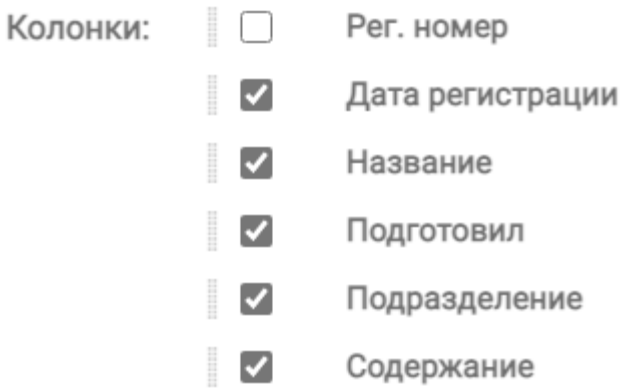


Рисунок 86. Настройки колонок в гриде

Колонки отображаются в порядке, определённом по умолчанию. Чтобы изменить порядок колонок, наведите курсор на заголовок колонки, нажмите левой кнопкой и перетащите его выше или ниже.

Строки



Рисунок 87. Настройки строк в гриде

В зависимости от выбранного значения, длинные строки в гриде буду переноситься или обрезаться. Перенос строк отключен по умолчанию.

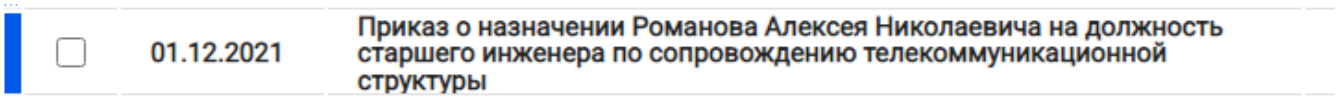


Рисунок 88. Строки переносятся

Когда перенос строк отключен, при наведении курсора на строку появляется полный текст во всплывающей подсказке.

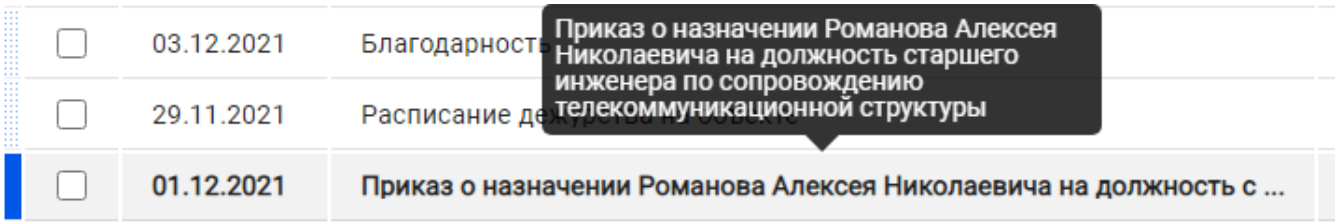


Рисунок 89. Строки обрезаются

Показывать кнопку "Помощь"

Кнопка **Помощь** отображается во всех представлениях, когда флаг установлен и не отображается нигде, когда флаг снят.

Настройка не является пользовательской и распространяется на все представления.

Сбросить настройки представления

Кнопка **Сбросить настройки представления** сбрасывает все пользовательские настройки для данного представления на настройки по умолчанию.

Следующие настройки будут сброшены при нажатии на кнопку:

- Видимость колонок.
- Порядок колонок.
- Перенос строк.
- Сортировки.
- Группировки.
- Ширина колонок.

Кнопка сбрасывает настройки для текущей папки, не путать с функцией "Вернуть настройки по умолчанию" в профиле пользователя.

Список последних выбранных сотрудников в ЭУ "Записи справочника сотрудников" очищаются автоматически при перезапуске службы Сервера Web-клиента.

Связь настроек и других функций грида

Нажмите применить, чтобы сохранить новые настройки.

После применения новых настроек сохраняются группировки, фильтры и сортировка, выполненные до изменения настроек.

Группировки, фильтры и сортировка по колонкам остаются прежними, даже если колонки скрыты из представления.

Мобильная версия грида

На мобильных устройствах грид имеет те же самые функции, но иной внешний вид.

Основные отличия в отображении дополнительной области и панели инструментов.

Одиночным касанием в любом месте строки можно открыть карточку.

Одиночным касанием индикатора прочитанности можно пометить карточку как прочитанную или не прочитанную.

Одиночным касанием флага слева от карточки можно вызвать панель групповых операций.

Нажатием и удержанием можно вызвать контекстное меню грида.

Дополнительная область

При нажатии на кнопку ⇉ дополнительная область отображается справа.

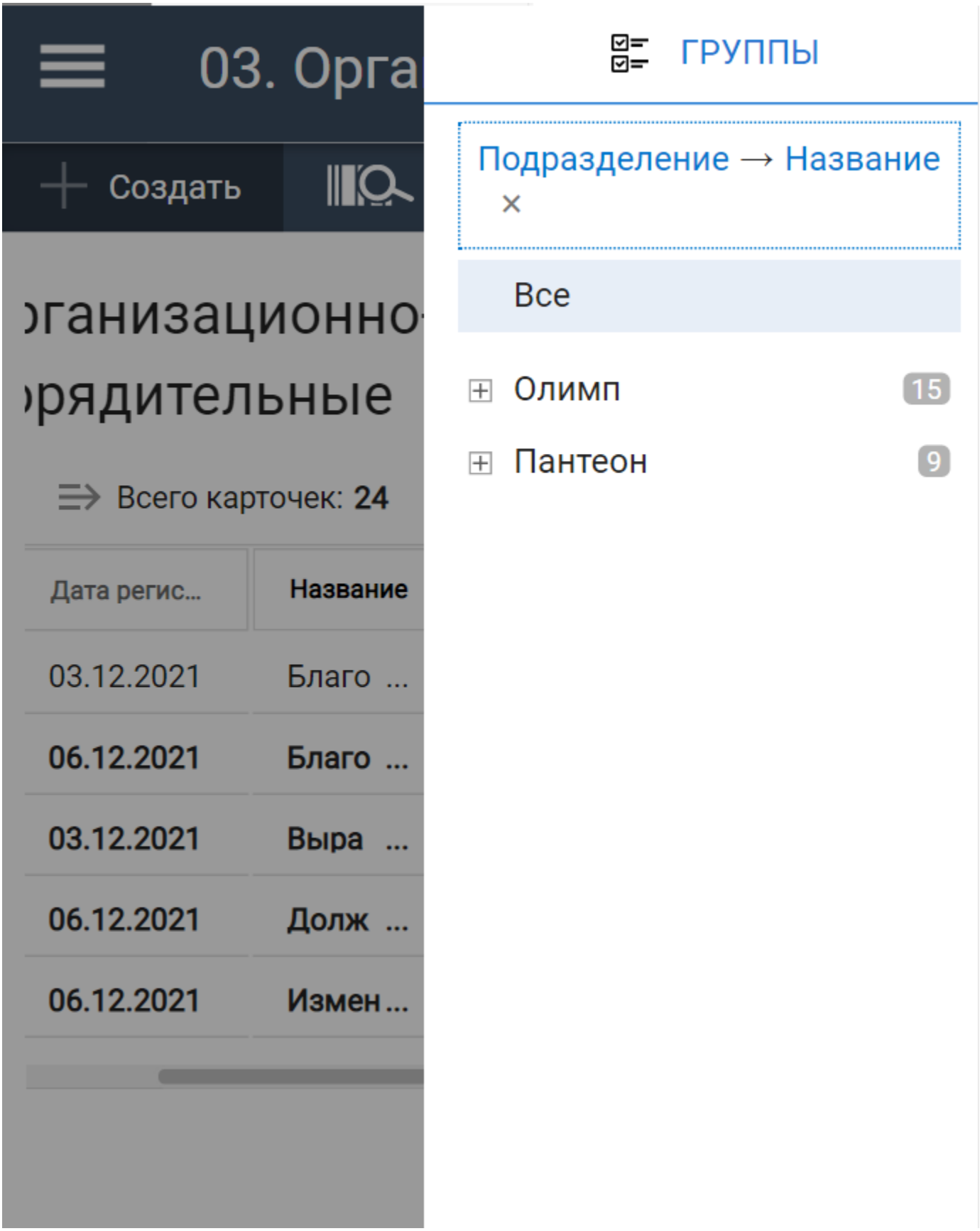


Рисунок 90. Дополнительная область в мобильной версии

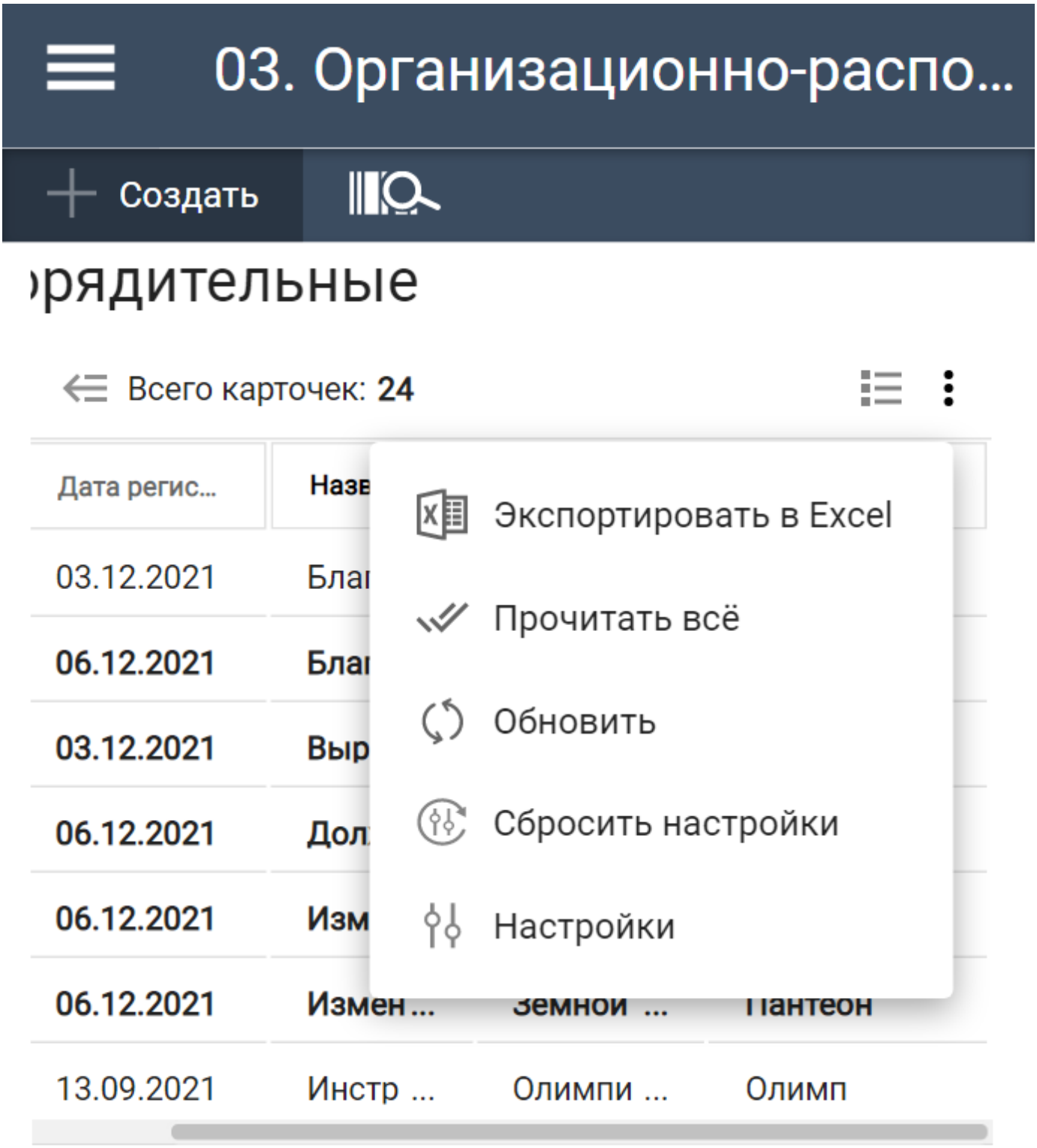


Рисунок 91. Панель инструментов в мобильной версии

Панель инструментов открывается нажатием на кнопку ⋮ и содержит те же функции, что и полная версия грида.

Размер папки

Индикация количества карточек в папке аналогична полной версии. Общее количество карточек отображается слева на панели управления в виде `Всего карточек: X`. Если применена группировка или фильтр, отображается `Показано: N из X` (например, 10 из 100). Кнопка "Показать всё" заменена крестиком: ✕.

Переключатель режима

Переключатель режима отображения находится справа, рядом с кнопкой отображения панели инструментов.

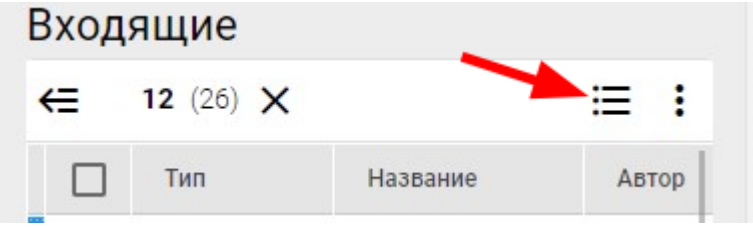


Рисунок 92. Переключатель в режим таблицы

Иконка переключателя зависит от текущего режима. Например, если текущий режим — таблица, отображается иконка ленты и наоборот.

Групповые операции

Кнопки групповых операций располагаются друг под другом.

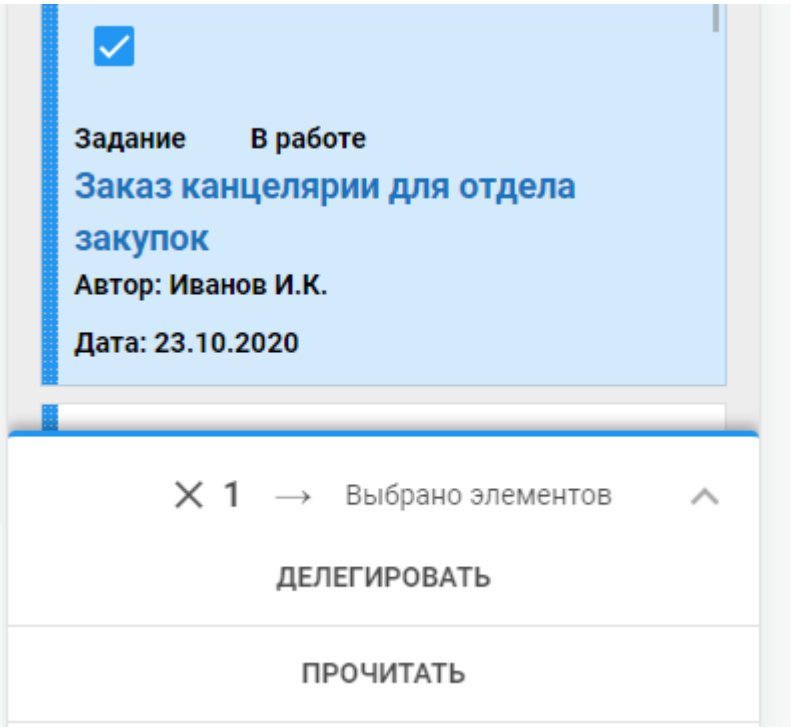


Рисунок 93. Кнопки групповых операций в мобильной версии

Панель групповых операций в мобильной версии изначально отображается распахнутой на высоту двух кнопок. Чтобы развернуть панель, нажмите на кнопку ^ и кнопку v, чтобы снова уменьшить панель до двух строк.

Фильтрация

Функция фильтрации (*Отфильтровать по значению*) доступна для вычисляемых колонок, но не работает на главной странице Web-клиента и в представлении *Дайджест*.

Доступность функции регулируется свойством "Фильтрация" ЭУ Карточки папки.

Работа с фильтрацией

Чтобы применить фильтр в представлении:

1. Наведите курсор на заголовок колонки.

   В режиме планшета нажмите один раз на заголовок колонки.

2. Нажмите кнопку .

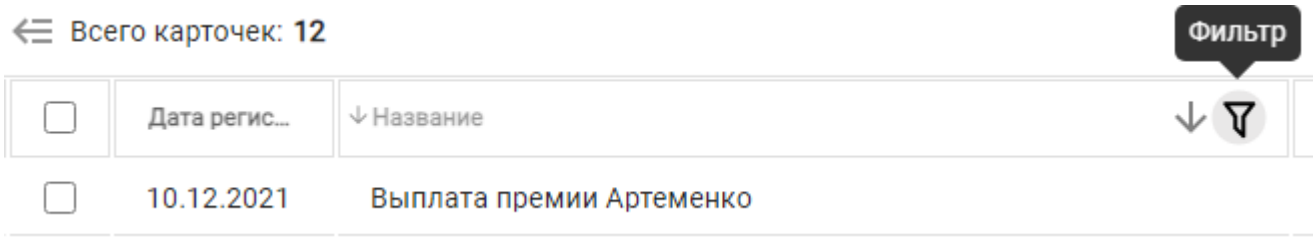


Рисунок 94. Кнопка фильтр в заголовке таблицы

3. Выберите нужный фильтр.

   Пользователь может найти нужный фильтр, начав вводить текст в поисковой строке. Совпадения подсвечиваются жёлтым во всех строках списка.

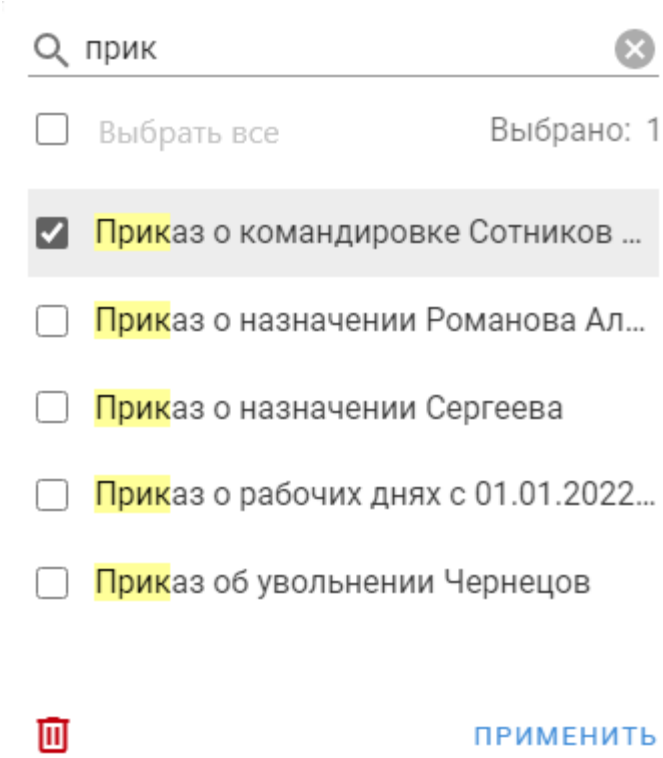


Рисунок 95. Поиск фильтров

4. Чтобы выбрать все значения, найденные по текущему запросу, установите флаг **Выбрать все**. Если снять флаг **Выбрать все**, выбор будет сброшен со всех значений.

   Флаг "Выбрать все" будет:

   - Установлен, если все найденные значения выбраны при вводе нового запроса.
   - Снят, если при вводе нового запроса выбраны не все найденные значения. При повторном открытии фильтра будут отображены все доступные значения, включая ещё не выбранные.
   - Не установлен, если выбор снят со всех значений, найденных по текущему запросу.

5. Чтобы выбрать нужное значение, установите напротив него флаг. Выбрать можно одно или несколько значений.

   Если значений много (больше 15), они загружаются поэтапно: сначала первые 15, затем при прокрутке частями загружаются остальные. Значения отображаются в порядке получения с сервера.

   Длинные названия фильтров обрезаются по ширине окна. При наведении на такое название появляется всплывающая подсказка с полным текстом.

6. Можно комбинировать несколько фильтров в разных колонках. Чтобы использовать несколько фильтров, примените фильтр к колонке, не сбрасывая предыдущие.



Если фильтр применён, в колонке отображается красный значок фильтра:

В колонке применён фильтр по названию



7. Чтобы изменить фильтр в колонке, нажмите на него ещё раз и выберите новое значение.
8. Чтобы сбросить фильтр, нажмите на кнопку . Чтобы сбросить все фильтры, нажмите на кнопку **Показать все**.

Фильтрация по датам

Если нажать кнопку  в колонке с датой, появится окно, позволяющее выбрать два варианта фильтрации: вкладки Дата и Журнал.

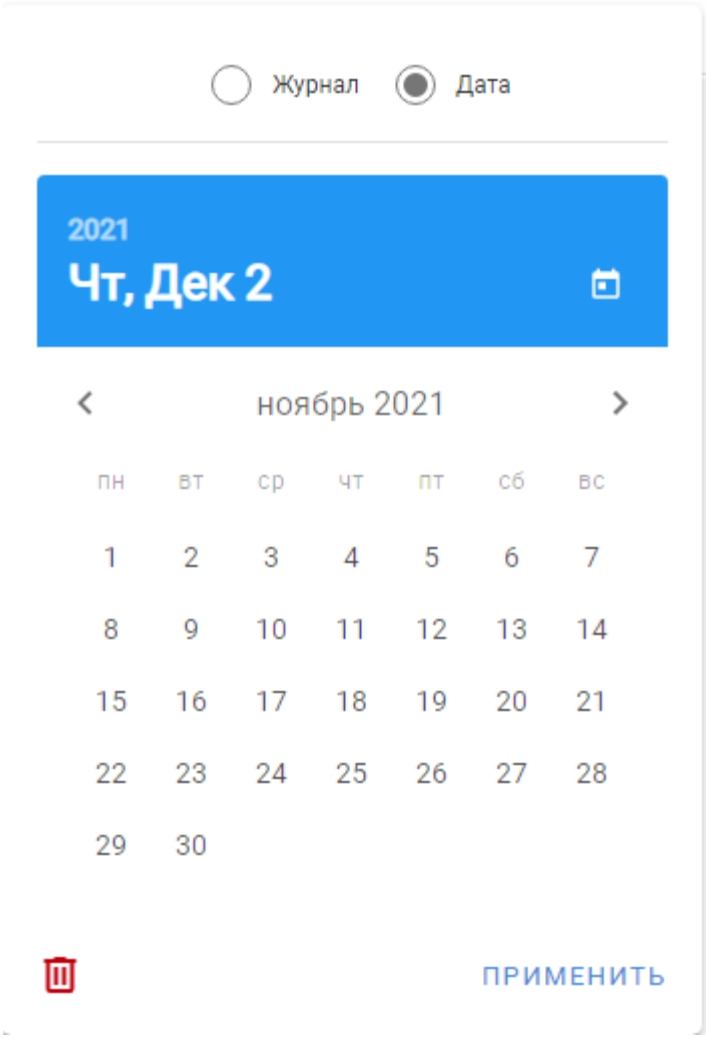


Рисунок 96. Использование фильтра по дате

Нельзя одновременно использовать фильтры из обеих вкладок. Например, вы выбрали фильтр на вкладке *Дата* и нажали кнопку **Применить**, затем перешли на вкладку *Журнал* и нажали кнопку **Применить**. В этом случае будет применён только фильтр с вкладки *Журнал*, фильтр на вкладке *Дата* будет сброшен.

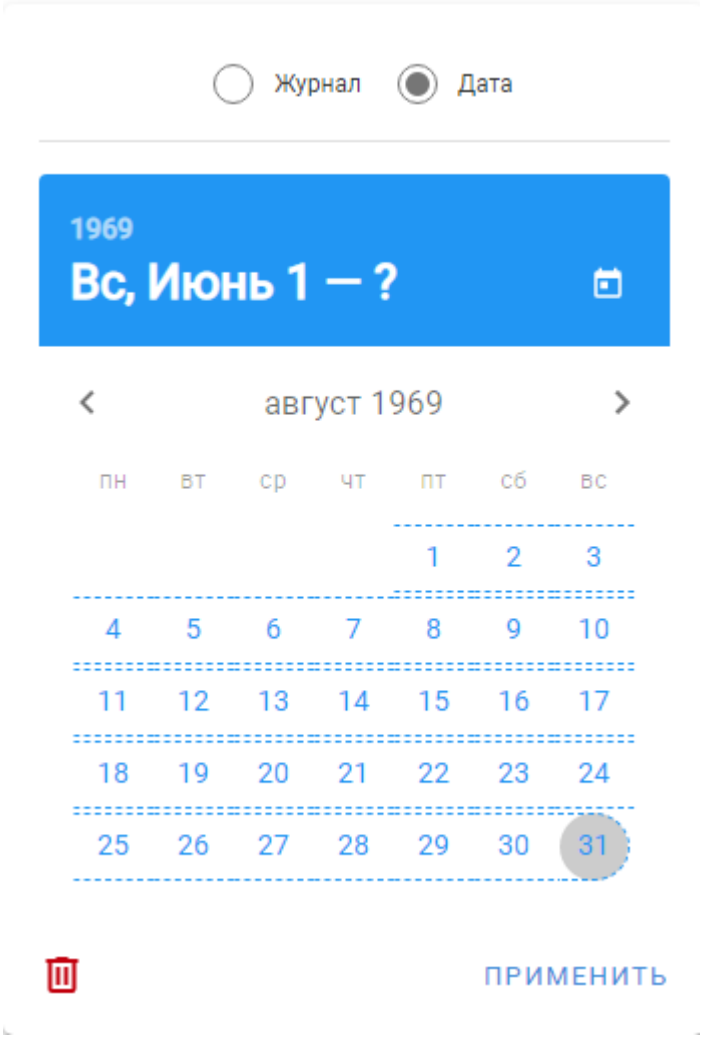
Вкладка "Дата"

В фильтре на вкладке *Дата* значение даты всегда отображается без времени, даже если в столбце время указано.

Вкладка *Дата* отображается первой при выборе фильтра.

При выборе данной вкладки открывается календарь, который позволяет выбрать конкретную дату, либо диапазон дат.

Выбор периода



По умолчанию в календаре выбрано сегодняшнее число.

- Чтобы выбрать другую дату, нажмите на число в календаре.
- Чтобы выбрать другой год, нажмите на отображаемый год.
- Чтобы выбрать диапазон дат, выберите два числа. Выберите дату начала периода, нажмите на неё, затем выберите дату окончания периода и нажмите на неё. Нажмите кнопку **Применить**, чтобы применить фильтр.

> Если вы выбрали дату окончания периода в прошлом (например, начало в 2020 году, а окончание в 2010), датой начала будет выбрана более ранняя дата (2010).

- Чтобы выбрать сегодняшнюю дату, нажмите на иконку календаря.

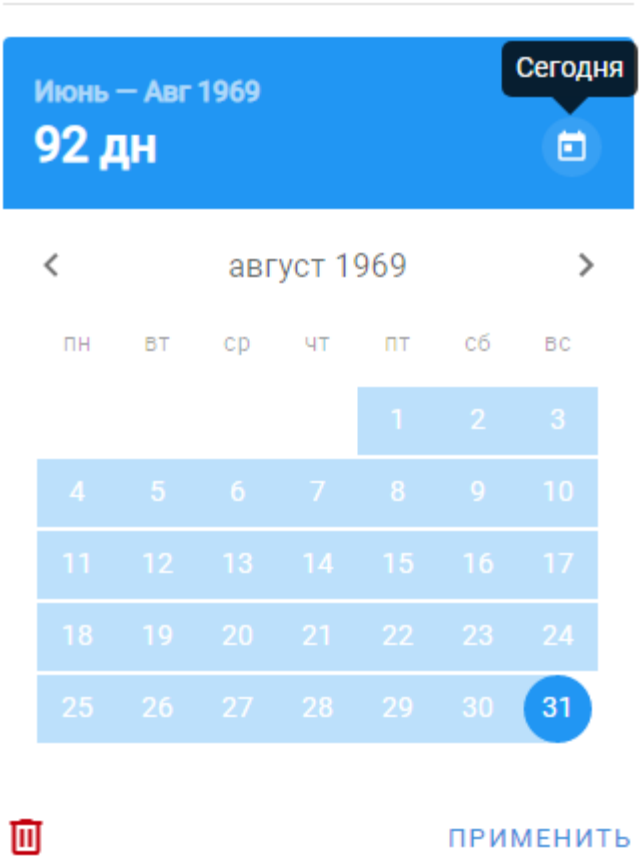


Рисунок 97. Выбранный период и кнопка "Сегодня"

Вкладка "Журнал"

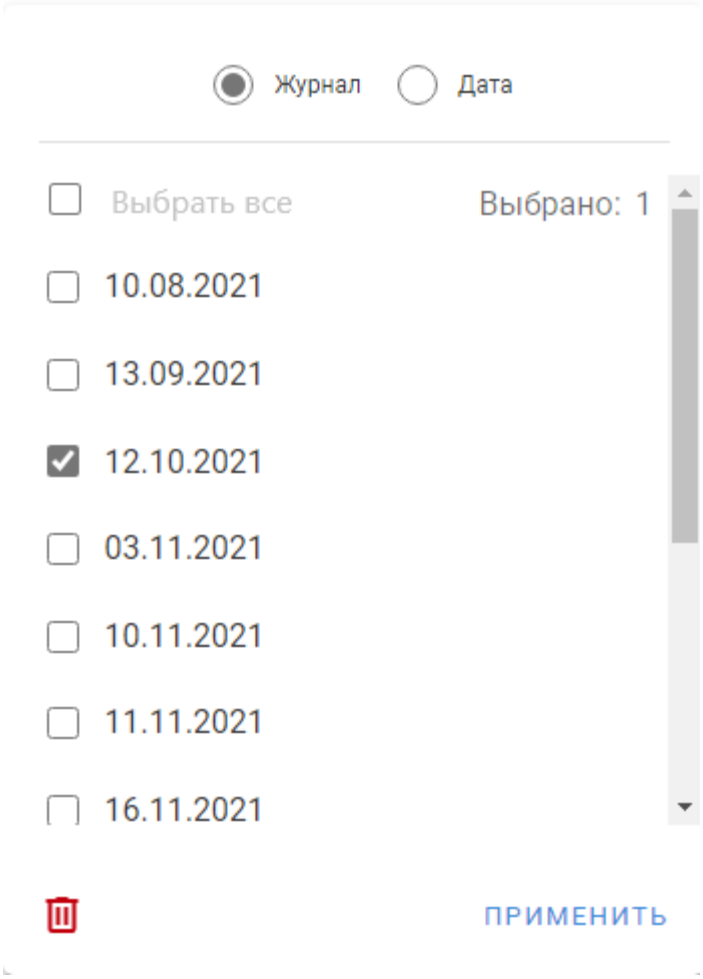


Рисунок 98. Вкладка "Журнал"

На вкладке *Журнал* отображается список уникальных дат, встречающихся в текущей папке. Список дат выводится в порядке, который получен с сервера. Если список слишком длинный, появляется вертикальная прокрутка.

Чтобы выбрать значение, установите напротив него флаг и нажмите кнопку **Применить**.

Индикация количества карточек

Если включены фильтр и/или группировка, в папке отображается количество карточек в виде: `Показано N из X`, где X — общее количество карточек в папке.

Если включена постраничная проверка прав, в папке отображается количество карточек в виде: `Показано N из X+`, где X — количество загруженных карточек.

Кнопка **Показать все** отображает все карточки в папке. Нажатие на кнопку не сбрасывает группировку, но сбрасывает фильтр.

Группировка

Функция группировки (*Сгруппировать по столбцу*) доступна для вычисляемых колонок, но не работает на главной странице Web-клиента и в представлении *Дайджест*.

Доступность функции регулируется свойством "Группировка" ЭУ Карточки папки.

Количество группируемых колонок не ограничено. При создании группировки сначала происходит группировка по первой колонке, затем, уже внутри групп — по второй и т. д.

Начиная с Web-клиента версии версии 6.1, сборка 1311, применённая группировка не сбрасывается при обновлении страницы, представления или при возврате из просмотра карточки по кнопке **Назад** или переходе на предыдущую страницу браузера. В таких случаях выбранные фильтр, группировка и группа сохраняются.

Если в карточках произошли изменения с последней группировки, при возврате количество карточек в группах будет изменено. Если условию группировки больше не соответствует ни одна карточка, пользователю будет выведено сообщение В выбранной группе больше нет карточек и кнопками **ОК** и **Показать все**. Нажатие на кнопку **ОК** откроет текущую выбранную группу, не имеющую карточек. Выбор **Показать все** сбрасывает применённую группировку, будут отображены все доступные карточки в текущем представлении.

Работа с группировками в представлении

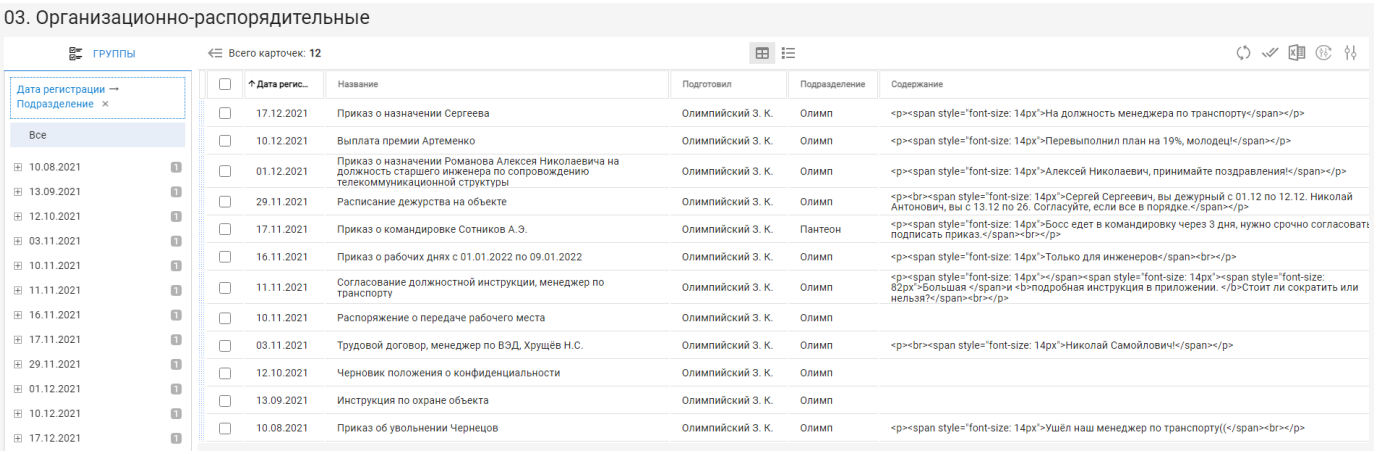


Рисунок 99. Грид с дополнительной областью группировки

Если группировка настроена в РМА:

- При открытии грида дополнительная область группировок отображается всегда вне зависимости от настроек панели управления грида.

- Удалять или добавлять группы можно, только если в настройках панели управления грида разрешена группировка.

Если группировка разрешена в настройках ЭУ "Карточки папки", но не настроена в РМА:

1. Группировка отображается в дополнительной области слева от грида. Дополнительная область по умолчанию скрыта и может быть отображена нажатием кнопки на панели управления гридом.

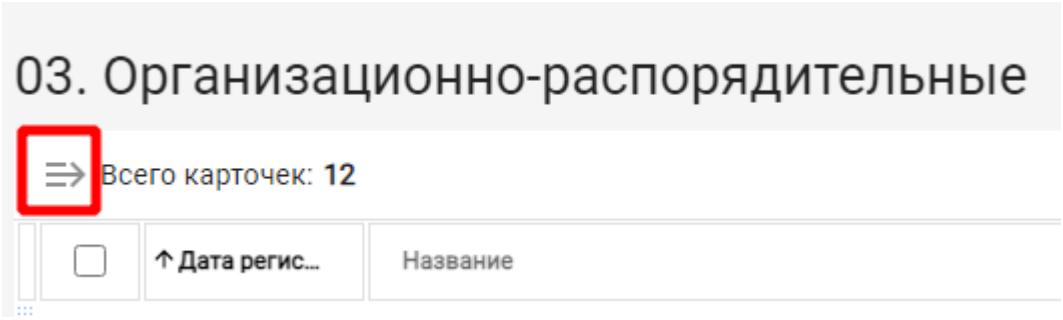


Рисунок 100. Кнопка отображения области группировок

2. Чтобы применить группировку, перетащите заголовок колонки из области грида в дополнительную область. В дополнительной области отобразится название группы — имя колонки представления. Перетащить можно один или несколько заголовков.

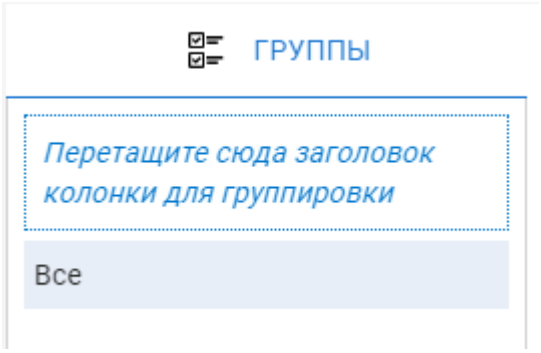


Рисунок 101. Область группировки

3. Напротив каждой группы отображается количество карточек. Чтобы перейти с одной группы на другую, нажмите на строку группы в дополнительной области. При этом текущий набор карточек в гриде будет заменён карточками выбранной группы.

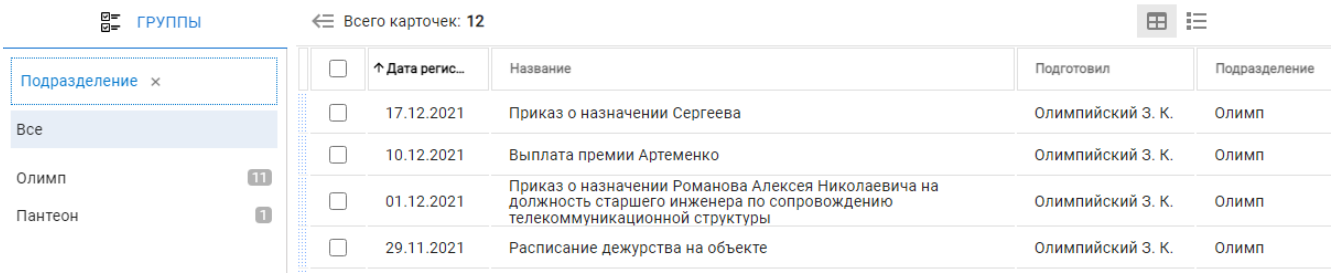


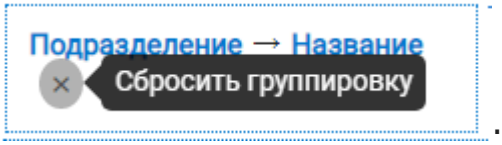
Рисунок 102. Группировка по столбцу "Подразделения"

4. При выбранной группировке колонки, кроме сгруппированных, могут скрываться.

> Колонки будут скрыты, когда группировка настроена в РМА.
>
> Отменить скрытие колонок можно в РМА: флаг **Представления** › **Представление** › **Установки** › **Не скрывать группируемые колонки**.

5. Чтобы сбросить группировку, нажмите на крестик в области группировок.

Кнопка сброса группировки в дополнительной области



.

6. Чтобы убрать только одну колонку из группировки, перетащите её заголовок из дополнительной области в свободную область Web-клиента.

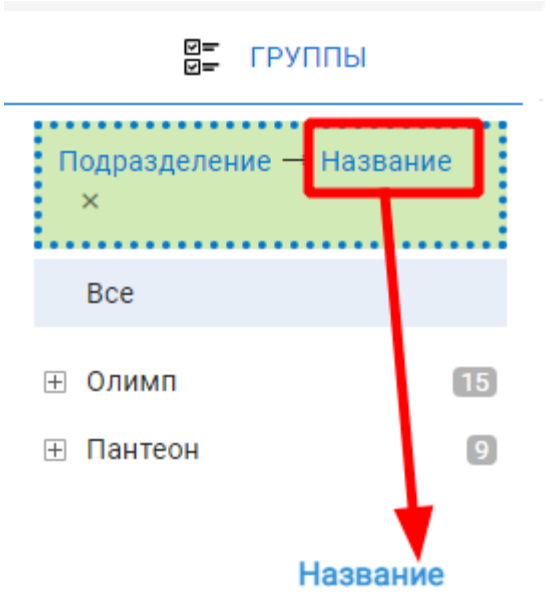


Рисунок 103. Разгруппировка

Если удалить одну из колонок группировки, в представлении колонка вернётся в последнее сохранённое положение.

Если удалить центральную колонку (например, в группе из трёх), она будет встроена обратно в грид. При этом группировка по оставшимся колонкам сохранится.

Взаимодействие с другими функциями грида

Сортировка

1. Если применяется группировка по колонке с сортировкой, сортировка колонки переносится на группы. Внутри каждой группы сортировка может быть настроена отдельно.
2. Если группировка применяется к колонке без сортировки, группы сортируются по возрастанию (от меньшего к большему, от А до Я).
3. Для колонки, встроенной обратно в грид, будет сохранена последняя сортировка. Например, если было 4 столбца и сортировка по 2 столбцу, то при возврате в грид 5 столбца, сортировка продолжает быть по 2 столбцу.

> Колонки будут скрыты и возвращены в грид, только если группировка включена в PMA.

4. При полном сбросе группировки карточки отсортированы по последней примененной сортировке.

Фильтры

Если осуществляется группировка по колонке, к которой был применен фильтр, то при группировке фильтр с этой колонки сбрасывается. Для остальных колонок фильтры остаются.

Поиск в папке

Если настроена группировка, поиск в папке осуществляется по группам. То есть после поиска в каждой группе выводятся только карточки, соответствующие критериям поиска

Индикация количества карточек

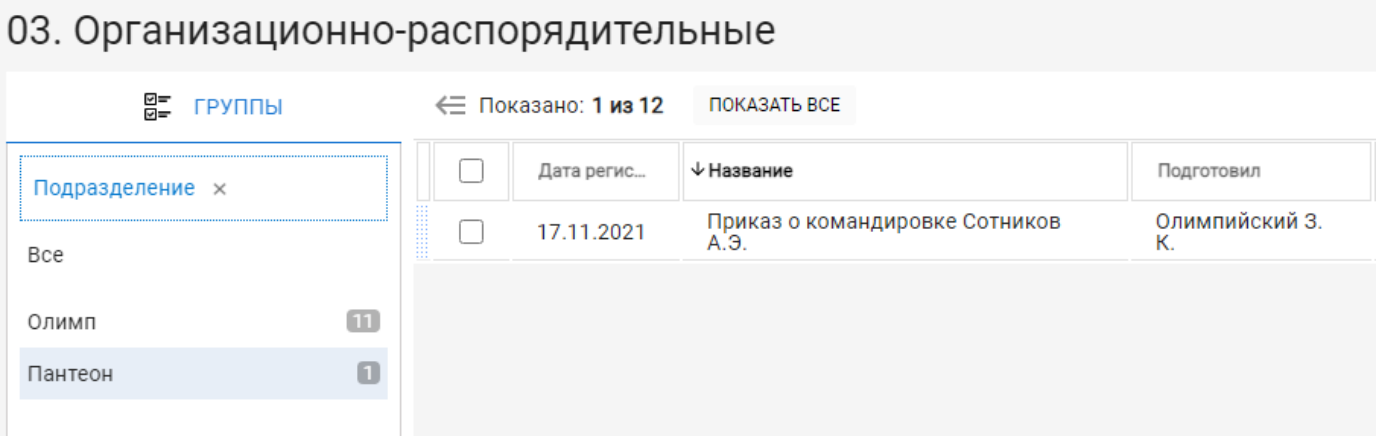


Рисунок 104. Количество карточек в группе

Если включены фильтр и/или группировка, в папке отображается количество карточек в виде: `Показано N из X`, где X — общее количество карточек в папке.

Если включена постраничная проверка прав, в папке отображается количество карточек в виде: `Показано N из X+`, где X — количество загруженных карточек.

Кнопка **Показать все** отображает все карточки в папке. Нажатие на кнопку не сбрасывает группировку, но сбрасывает фильтр.

Постраничная проверка прав

1. Если группировка была включена по умолчанию в папке с постраничной проверкой прав, данные папки могут загружаться дольше.
2. Если пользователь самостоятельно применяет группировку в папке с настроенной постраничной проверкой прав, операцию необходимо подтвердить.
    - Если пользователь подтверждает действие, загружаются все данные папки.

- Если пользователь не подтверждает действие, группировка не будет применена и данные папки не будут загружены.

3. Если пользователь переключает представление по умолчанию на представление с группировкой и постраничной проверкой прав, действие необходимо подтвердить.
   - Если пользователь подтверждает действие, загружаются все данные папки.
   - Если пользователь не подтверждает действие, группировка не будет применена и данные папки не будут загружены.

Сортировка

Базовая сортировка

Доступность функции регулируется свойством "Сортировка" ЭУ Карточки папки.

Сортировку можно применить к любому столбцу грида, но только к одному.

По умолчанию используется сортировка, настроенная в представлении. Однако, значок сортировки по умолчанию не отображается. Если сортировка по умолчанию не настроена, карточки сортируются по дате изменения.



Сортировка недоступна в мобильной версии Web-клиента.

Чтобы применить сортировку:

1. Наведите курсор на заголовок колонки грида и нажмите на него.
2. Данные в представлении будут отсортированы в восходящем порядке (от А до Я, от меньшего к большему). Чтобы инвертировать порядок сортировки, нажмите на заголовок столбца ещё раз. При этом стрелка сортировки поменяет направление.



Если сортировка была применена, перед заголовком колонки появится маленький значок порядка сортировки. Если нажать на кнопку сортировки ещё раз, порядок будет инвертирован.

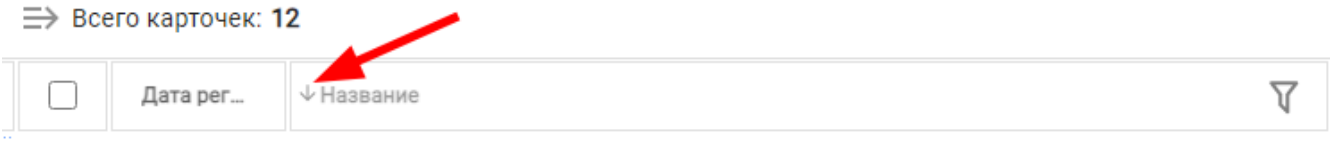


Рисунок 105. Индикатор сортировки по возрастанию и кнопка инвертирования сортировки

3. Если вы нажмёте на значок сортировки в другой колонке, предыдущая сортировка будет сброшена.
4. Если вы нажмёте кнопку **Сбросить настройки**, сортировка будет сброшена на стандартные значения.

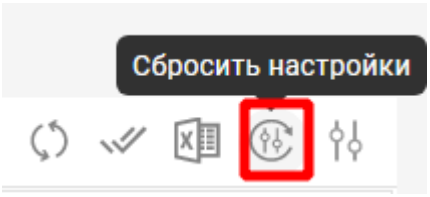


Рисунок 106. Кнопка "Сбросить настройки"

Плагин сортировки

В локации результаты поиска при отображении грида в режиме таблицы и при отображении грида в режиме ленты в любой локации используется плагин сортировки.

Плагин отображается как значок в панели инструментов грида. При наведении на значок появляется выпадающий список. Выберите из выпадающего списка сортировку, чтобы применить её.

Порядок сортировки можно изменить повторным нажатием на индикатор в выпадающем списке. Сортировка в плагине применяется ко всем режимам отображения. То есть, если изменить сортировку в таблице грида, сортировка также изменится в ленте и наоборот.

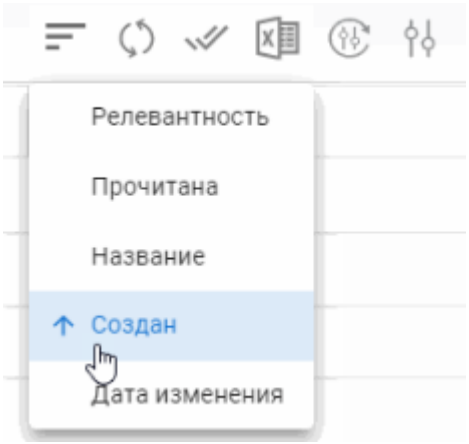


Рисунок 107. Выпадающий список сортировки

Если в локации *Результаты поиска* сортировка не задана, результаты будут отсортированы по релевантности при помощи плагина. Для остальных локаций сортировка в плагине соответствует сортировке применённой в режиме отображения "таблица".

Групповые операции в новом гриде

Новый грид поддерживает работу с групповыми операциями.

Доступные групповые операции нового грида:

- Прочитать.
- Копировать ярлыки карточек.
- Переместить ярлыки карточек.
- Удалить ярлыки карточек.
- Групповое делегирование.

Групповая операция — операция, которая применяется сразу к нескольким карточкам.

Если групповые операции недоступны, выводится соответствующее сообщение. Сообщение может быть изменено в программе Конструктор Web-разметок, в свойстве Текст, если нет групповых операций.

Особенности групповых операций в Web-клиенте

Особенности работы с групповыми операциями в Web-клиенте:

- Групповая операция является синхронной. Во время выполнения групповой операции другие операции недоступны.
- Групповые операции выполняются последовательно.
- Выполнение групповой операции можно прервать. При этом невыполненные операции не будут завершены.
- Групповая операция выполняется в веб-браузере. Страница Web-клиента, в которой выполняется групповая операция, должна оставаться открытой до завершения, иначе выполнение будет прервано.
- При попытке выполнить групповую операцию происходит проверка. Проверяется возможность выполнить данную операцию.

Работа с групповыми операциями

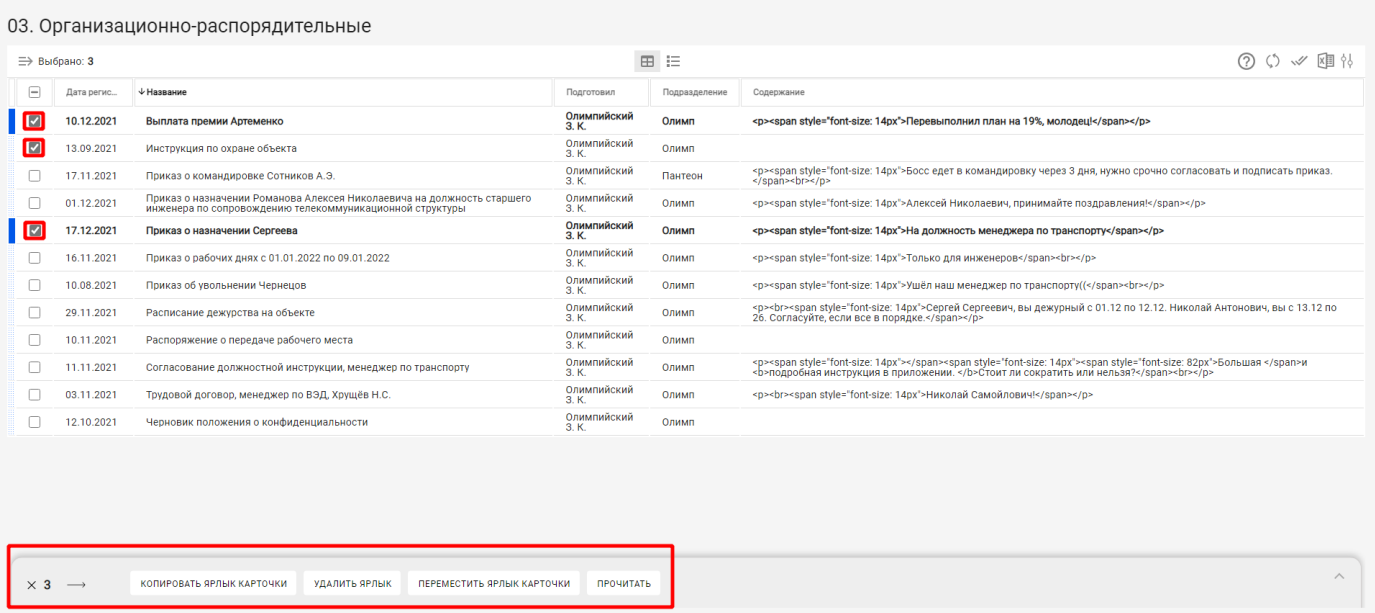


Рисунок 108. Панель групповых операции в гриде

1. Чтобы открыть новые групповые операции, выберите одну или несколько карточек в папке при помощи флага.



В мобильной версии коснитесь и удерживайте палец на нужной карточке до появления строки групповых операций.

Коснитесь других карточек, чтобы добавить их в групповую операцию. Коснитесь повторно, чтобы снять выбор.

Количество выбранных карточек будет указано в строке групповых операций.

2. В появившейся панели групповых операций выберите нужную операцию.
3. Если для выбранных карточек недоступны групповые операции, будет отображаться сообщение, заданное в свойстве "Текст, если нет групповых операций" элемента управления `Узел для групповых операций`.

Прочитать

Чтобы прочитать несколько карточек:

1. Выберите нужные карточки, поставив флаги слева от заголовка строки.
2. Нажмите кнопку **Прочитать** на панели групповых операций.
3. Дождитесь завершения групповой операции.

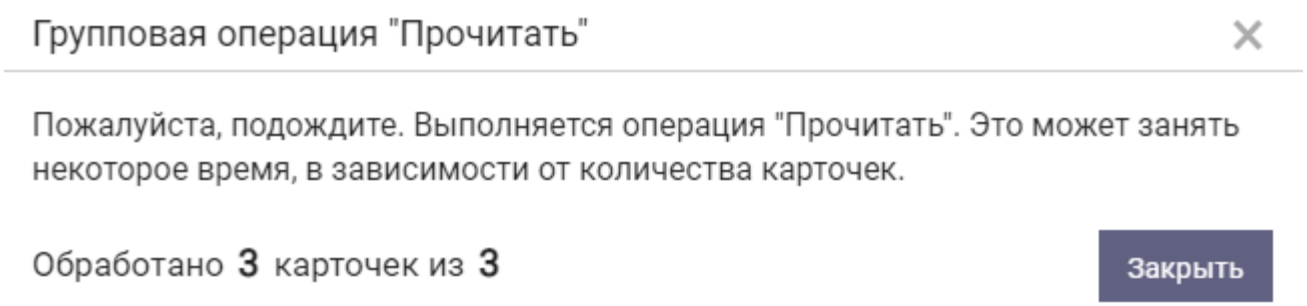


Рисунок 109. Выполнение групповой операции "Прочитать"

Копировать ярлыки карточек

Чтобы копировать ярлыки карточек из одной папки в другую:

1. Выберите нужные карточки, поставив флаги слева от заголовка строки.
2. Нажмите кнопку **Копировать ярлык карточки** на панели групповых операций.
3. В появившемся окне выберите папку, в которую нужно копировать ярлыки.
4. Нажмите кнопку **Копировать ярлык карточки**.
5. Дождитесь завершения групповой операции.

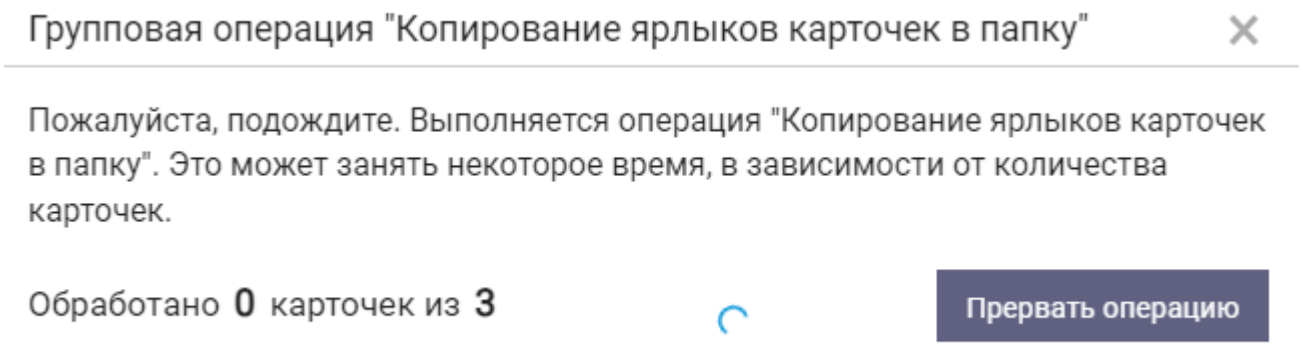


Рисунок 110. Копирование ярлыков карточек

Переместить ярлыки карточек

Чтобы переместить ярлыки карточек из одной папки в другую:

1. Выберите нужные карточки, поставив флаги слева от заголовка строки.
2. Нажмите кнопку **Переместить ярлык карточки** на панели групповых операций.
3. В появившемся окне выберите папку, в которую нужно переместить ярлыки.

4. Нажмите кнопку **Переместить ярлык карточки**.
5. Дождитесь завершения групповой операции.

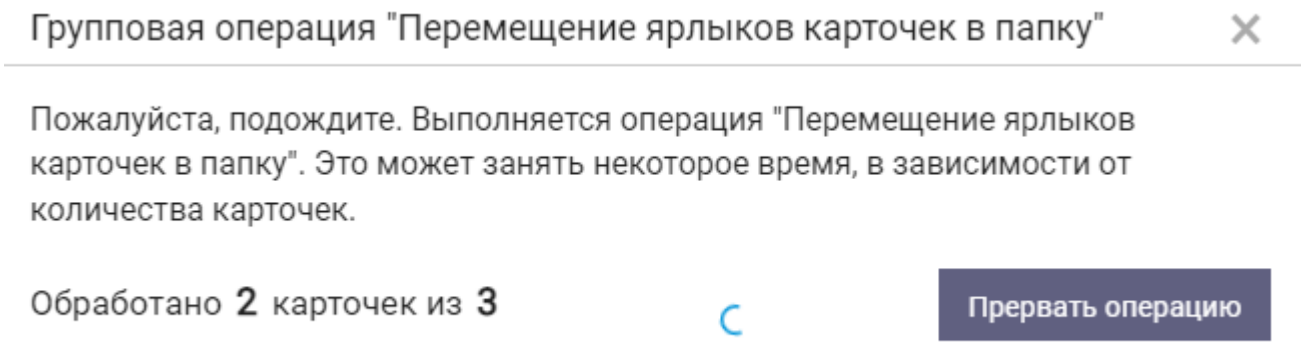


Рисунок 111. Премещение ярлыков карточек

Удалить ярлык

Чтобы удалить ярлыки нескольких карточек:

1. Выберите нужные карточки, поставив флаги слева от заголовка строки.
2. Нажмите кнопку **Удалить ярлык** на панели групповых операций.
3. Подтвердите удаление ярлыков.

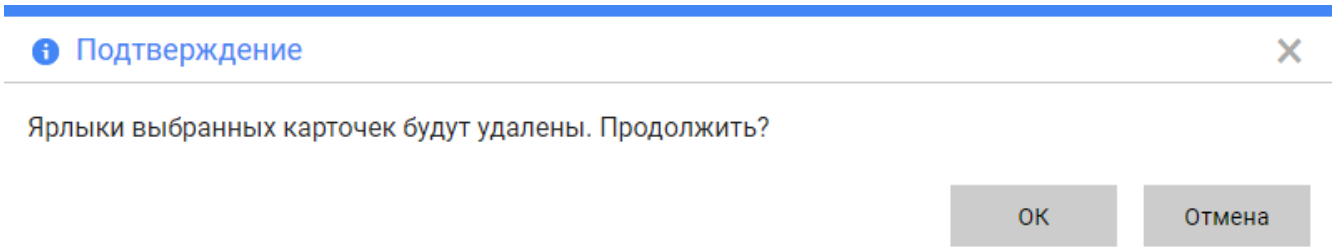


Рисунок 112. Подтверждение удаления ярлыков

4. Дождитесь завершения групповой операции.

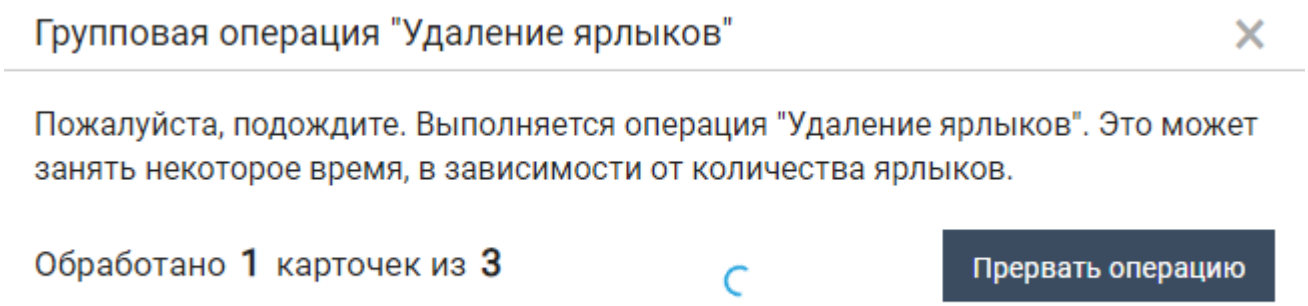


Рисунок 113. Выполнение групповой операции "Удалить ярлык"

Удалить карточки

При удалении карточек проверяются права пользователя на удаление этих карточек по ролевой модели и по логике Docsvision (например, запрещено удалять карточки, связанные сильными ссылками). В случае, если у пользователя нет прав на удаление одной или нескольких карточек, будет выдано сообщение об ошибке с объяснением причин, почему удаление невозможно.

Чтобы удалить несколько карточек:

1. Выберите нужные карточки, поставив флаги слева от заголовка строки.
2. Нажмите кнопку **Удалить карточки** на панели групповых операций.
3. Подтвердите удаление карточки.

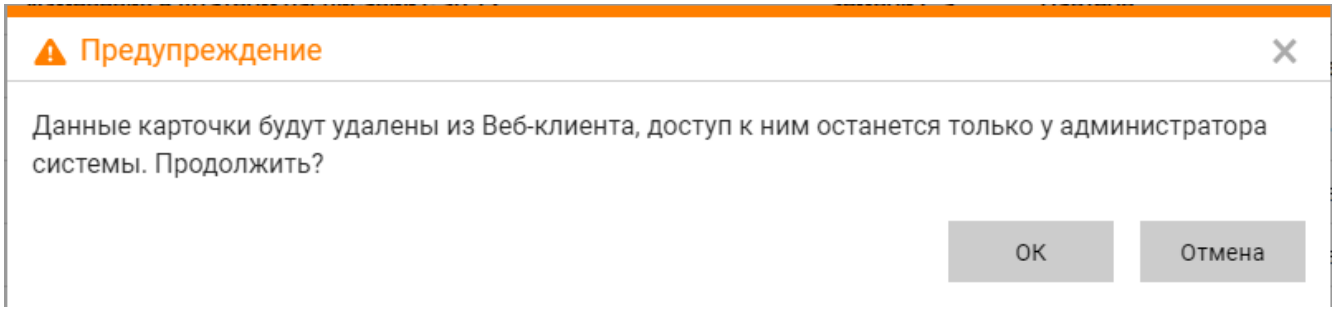


Рисунок 114. Подтверждение удаления карточки

4. Дождитесь завершения групповой операции.

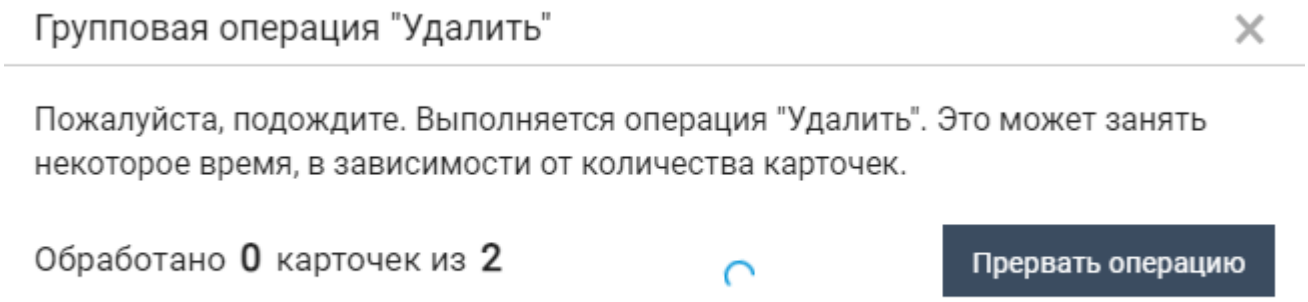


Рисунок 115. Выполнение групповой операции "Удалить карточки"

Карточки удаляются в корзину или безвозвратно в зависимости от настроек в файле конфигурации Web-клиента.

Скачать файлы карточки

Чтобы скачать файлы из нескольких карточек:

1. Выберите нужные карточки, поставив флаги слева от заголовка строки.
2. Нажмите кнопку **Скачать файлы** на панели групповых операций.
3. Дождитесь завершения групповой операции.

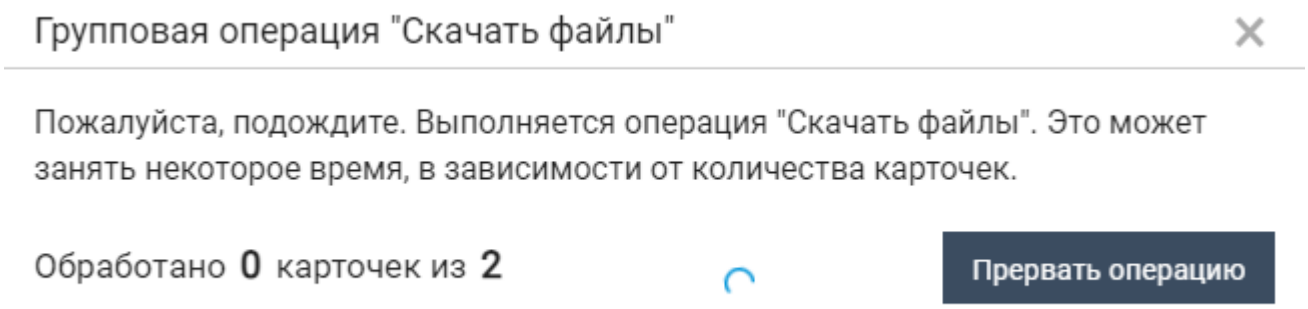


Рисунок 116. Выполнение групповой операции "Скачать файлы карточки"

Если в процессе выполнения групповой операции возникли ошибки, информация будет выведена в окне выполнения.

Из карточки могут быть скачаны основные файлы, дополнительные или файлы с подписями в зависимости от заданных свойств ЭУ групповой операции. Файлы каждой карточки будут скачаны одним архивом.

Групповое делегирование

Чтобы делегировать несколько заданий:

1. Выберите нужные карточки, поставив флаги слева от заголовка строки.
2. Нажмите кнопку **Делегировать** на панели групповых операций.

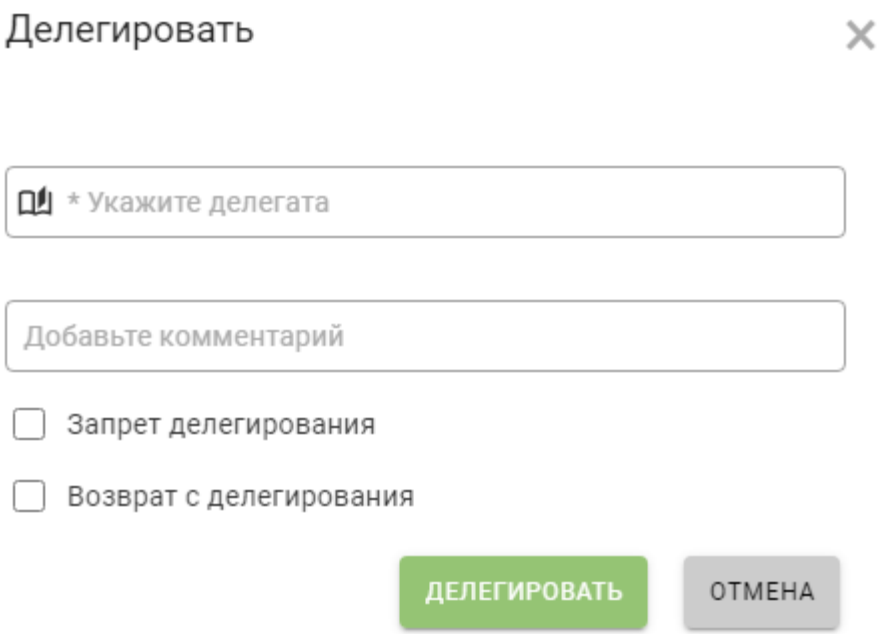


Рисунок 117. Делегирование задания

3. Выберите сотрудника, которому делегируются задания.
4. Добавьте комментарий к делегированию.
5. Чтобы запретить повторное делегирование заданий, установите флаг `Запрет делегирования`.
6. Чтобы вернуть себе задания после их завершения делегатом, установите флаг `Возврат с делегирования`.
7. Нажмите кнопку **Делегировать**, чтобы запустить процесс делегирования заданий.

Состояние выполнения групповой операции отображается в диалоговом окне.

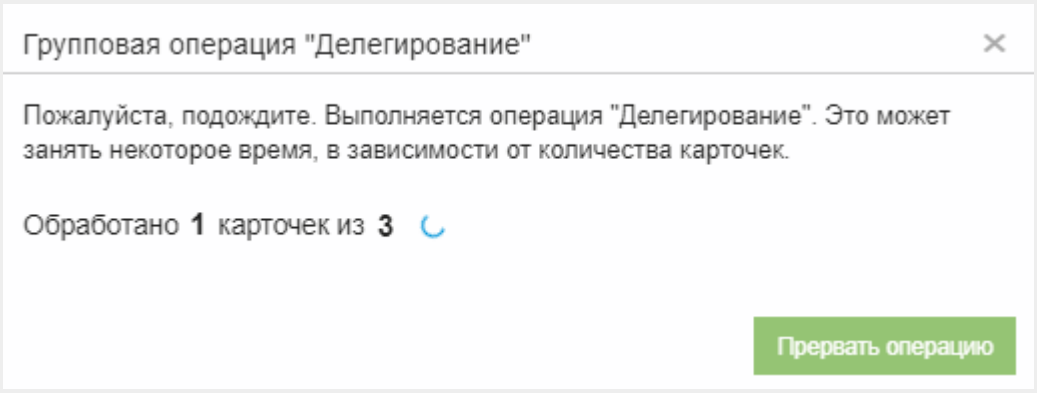


Рисунок 118. Диалоговое окно групповой операции "Делегирование"

Чтобы прервать выполнение операции, нажмите кнопку **Прервать операцию**.

Если групповая операция завершилась без ошибок, диалоговое окно будет автоматически закрыто.

Если в процессе выполнения групповой операции возникли ошибки, информация о них будет показана в диалоговом окне.

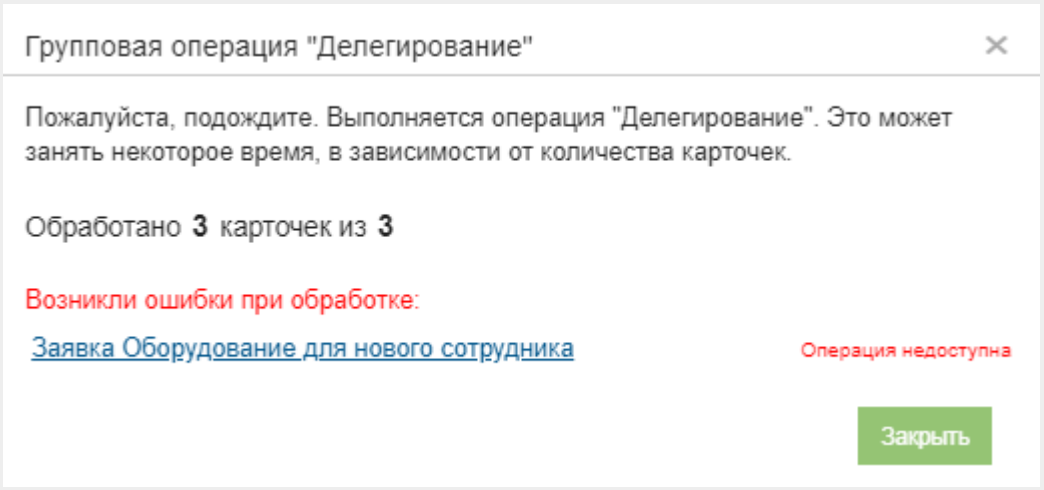


Рисунок 119. Диалоговое окно групповой операции "Делегирование" со списком ошибок

Для каждой карточки с ошибкой отображается ссылка на карточку и причина ошибки.

После исправления ошибок групповое делегирование для требуемых карточек может быть выполнено повторно.

8. Чтобы выйти из режима групповых операций, нажмите кнопку ✕ на панели групповых операций.

Лента

При нажатии правой кнопки на панели управления переключается режим отображения грида. При переключении на режим отображения "лента" окно позиционируется на самой верхней карточке в окне.

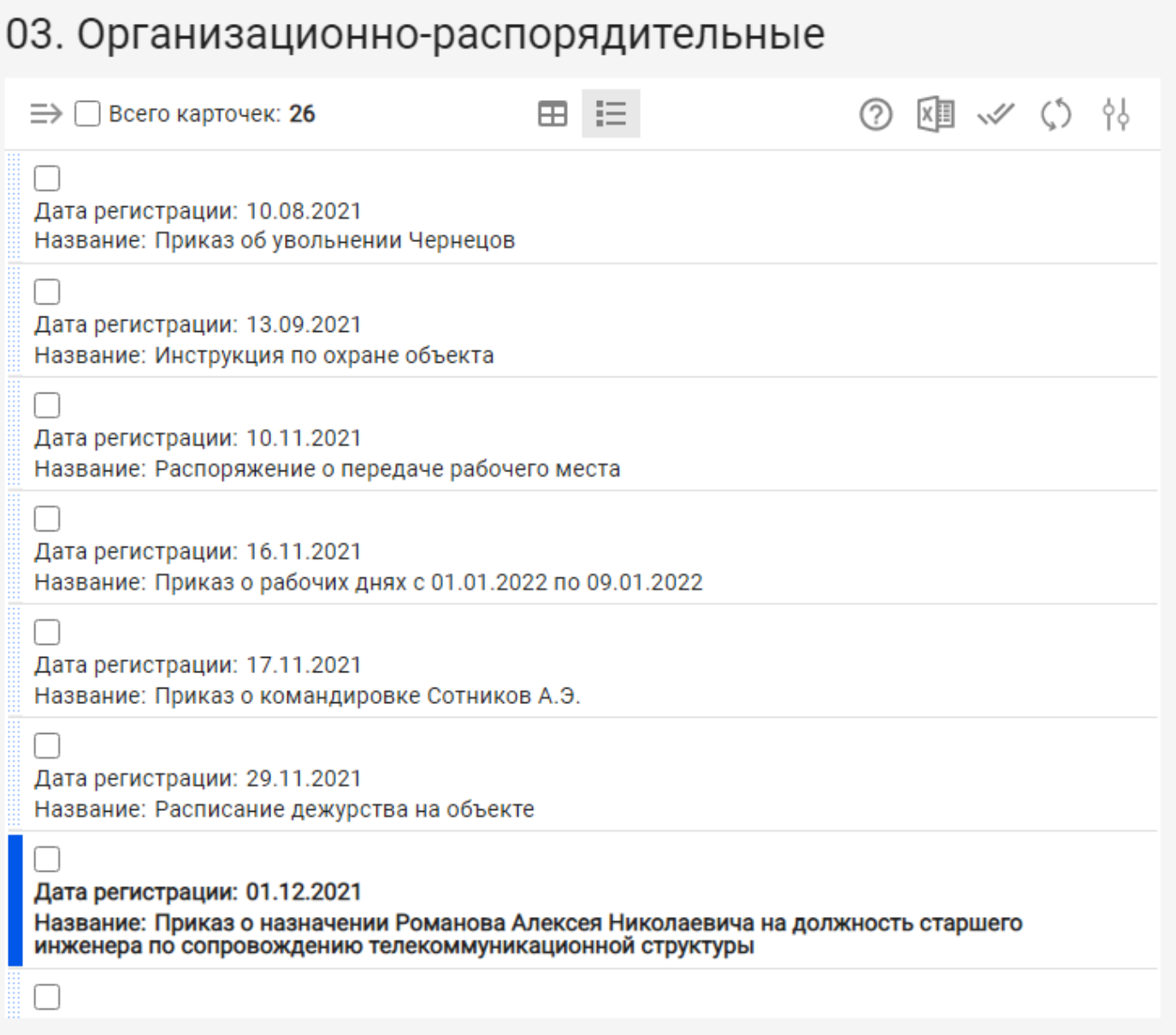


Рисунок 120. Общий вид ленты

Как и в табличном режиме отображения, каждая строка ленты представляет отдельную карточку.

Карточки

В левой части строки отображается индикатор прочитанности карточки. Если в строке текст выделяется жирным и слева строки отображается синий индикатор, значит, карточка не прочитана. Чтобы прочитать карточку, нажмите на индикатор один раз. Чтобы отметить карточку как непрочитанную, нажмите слева от заголовка карточки.

В каждой строке отображается флаг для выбора карточки и активации групповых операций.

Слева на панели инструментов отображается флаг для выбора всех карточек в папке и количество карточек.

В центре панели инструментов отображается кнопка переключения режима отображения грида: .

Сортировка

Также на панели инструментов отображается значок сортировки . По нажатию на значок появляется выпадающий список, в котором можно задать направление сортировки.

При смене сортировки данные в ленте меняют порядок.

Настройки

Окно "Настройки ленты" в целом аналогично настройкам таблицы, за исключением возможности переноса строк.

Настройка *Колонки* в режиме ленты имеет свои особенности. Информация, которая отображается в колонках в режиме таблицы, представлена в виде строк карточки.

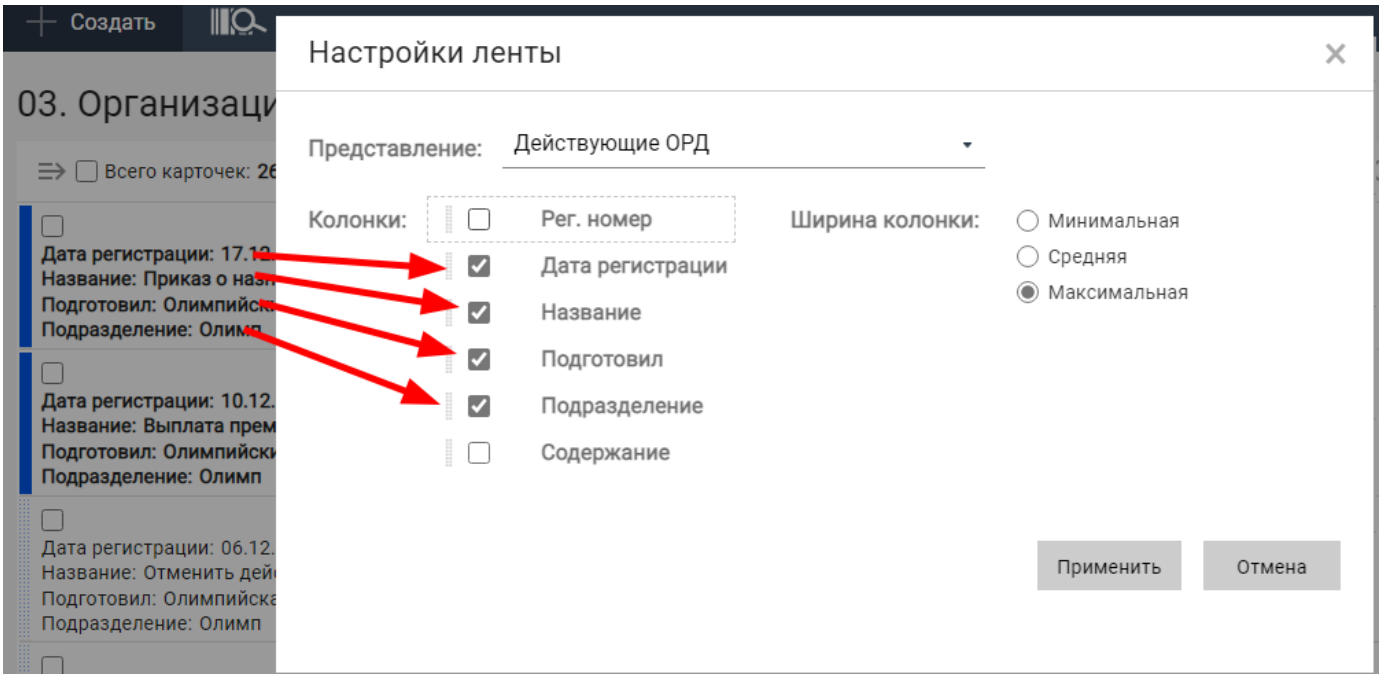


Рисунок 121. Настроенные "колонки" в ленте

По умолчанию для отображения выбираются три первых колонки, например *Рег. номер*, *Дата регистрации*, *Название*.

Все колонки добавляются в карточку в ленте друг под другом. Чтобы изменить их порядок, поменяйте расположение заголовков.

Настройки колонок для ленты не переносятся в табличное отображение. Если в ленте что-то поменяли, в том же представлении грида настройки остаются неизменными.

В правой части окна настроек ленты располагаются настройки ширины колонки:

- При выборе пункта **Минимальная** все данные ленты будут отображены в несколько узких колонок (значение по умолчанию).

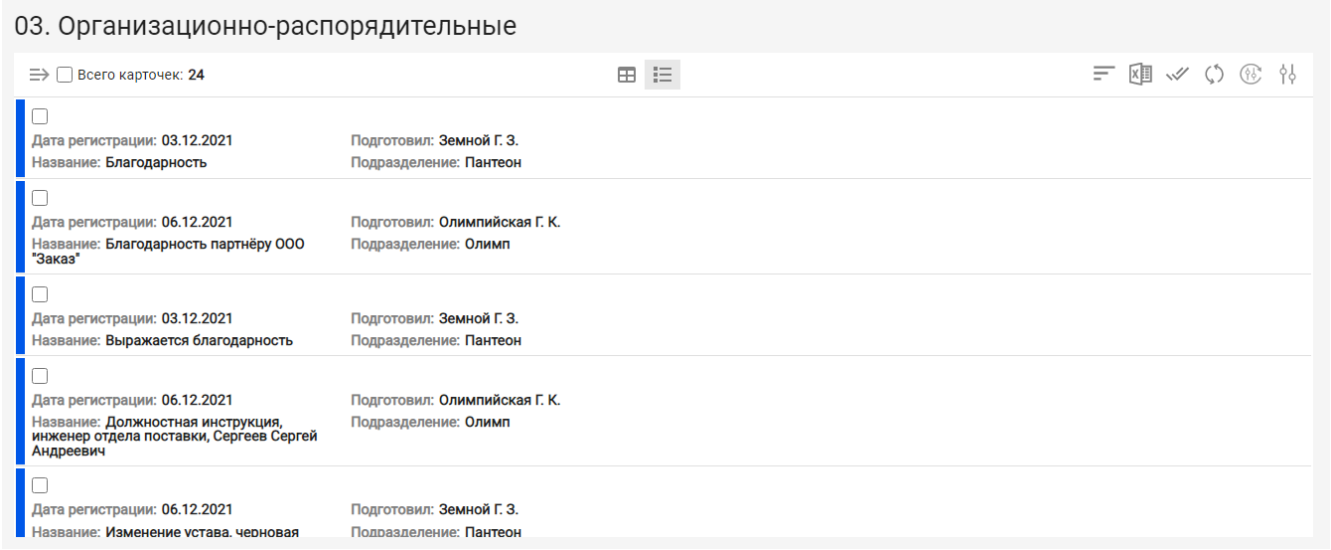


Рисунок 122. Минимальная ширина колонки

- При выборе пунктов **Средняя** и **Максимальная** данные ленты будут отображены в несколько широких колонок, а поля данных будут распределяться по области ленты более равномерно.

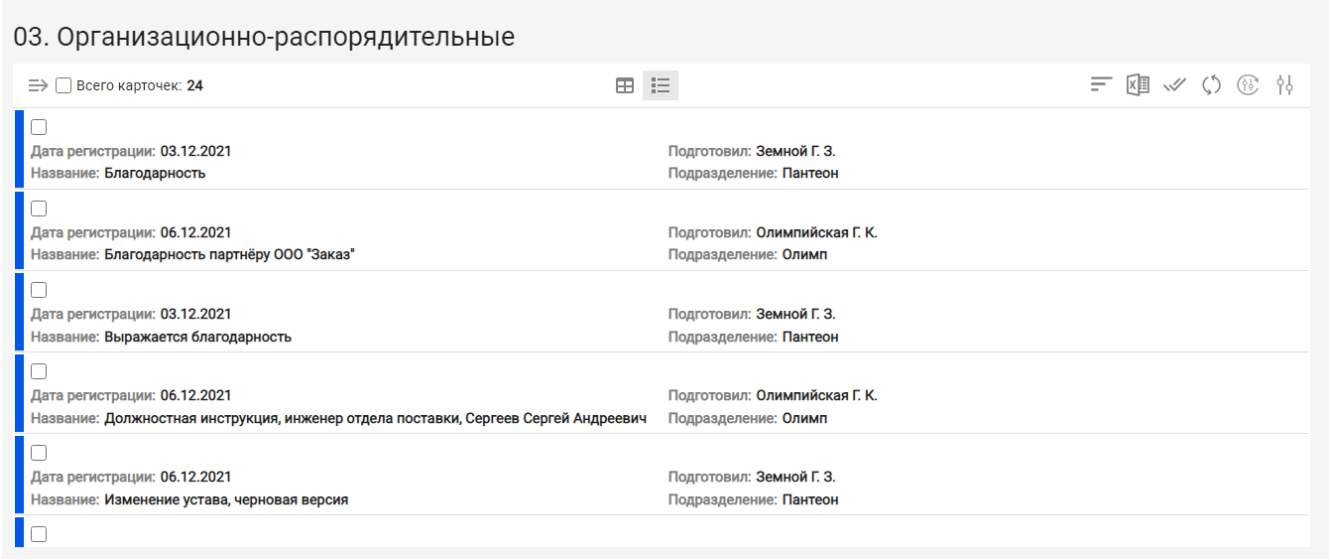


Рисунок 123. Максимальная ширина колонки

Текст, который не помещается в одну строку, переносится всегда.

Экспорт

При экспорте ленты в Excel результирующий файл всё равно отображается в виде таблицы, но с теми полями и в том порядке, что были выбраны в окне *Настройки ленты*.

Прокрутка

Прокрутка в ленте работает аналогично прокрутке в таблице.

Контекстное меню

Функции, доступные из контекстного меню ленты:

- *Скопировать адрес ссылки* — копирует адрес ссылки для карточки, в строке которой было вызвано контекстное меню.
- *Открыть* — открывает карточку, на строке которой было вызвано контекстное меню.
- *Открыть в новой вкладке* — в новой вкладке открывает карточку, на строке которой было вызвано контекстное меню.
- *Открыть в новом окне* — в новом окне открывает карточку, на строке которой было вызвано контекстное меню.

Цитаты во вложениях

Для отображения цитат во вложениях требуется иметь опцию *Docsvision Модуль Интеллектуальный поиск* лицензии Docsvision. Если опция отключена, цитаты не

будут отображаться в режиме ленты.

Если используется полнотекстовый поиск Elasticsearch, в режиме ленты отображается список вложенных файлов и их количество.

Если включён полнотекстовый поиск, режим ленты поддерживает отображение цитат из файла, в котором обнаружено соответствие поисковому запросу.

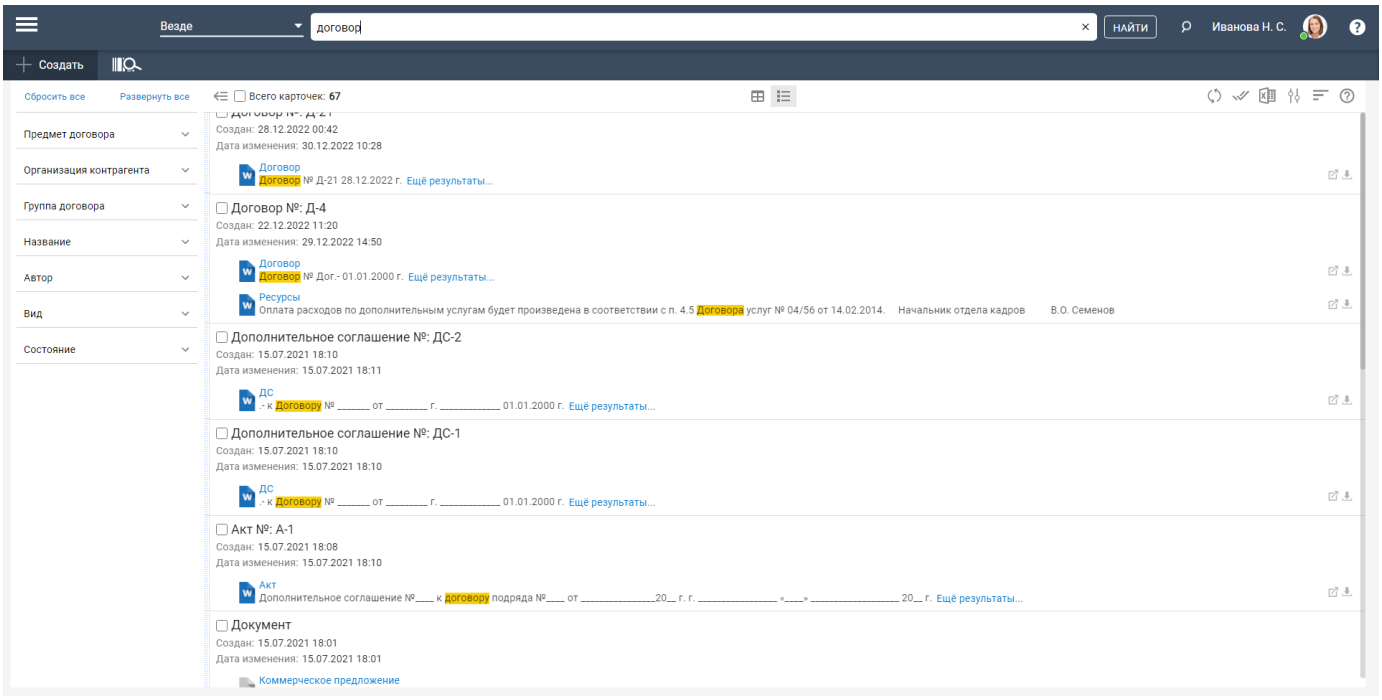


Рисунок 124. Поиск по вложениям

Если в файлах карточки было найдено совпадение поисковому запросу, в результатах поиска выводится карточка со списком файлов. В результатах поиска отображается максимум два файла.

Если в карточке больше двух файлов, появляется ссылка с текстом "Ещё файлы…". Нажмите на этот текст, чтобы показать все файлы карточки, в которых было найдено совпадение.

На каждый файл отведена одна строка — первая, в которой было найдено совпадение. Если в файле совпадений больше одного, справа от цитаты появляется ссылка с текстом "Больше…".

Чтобы уменьшить список файлов до двух строк, нажмите на текст "Меньше…".

Кнопки в окне найденных совпадений в файле:

- **Отмена** — закрывает окно.

- **Просмотреть файл** — открывает окно предварительного просмотра файла.
- **Открыть** — (кнопка отображается только для файлов Microsoft Word и Excel), открывает файл в программе на компьютере.
- **Скачать** — загружает файл в папку загрузок браузера.



Цитаты из файла упорядочены по релевантности.

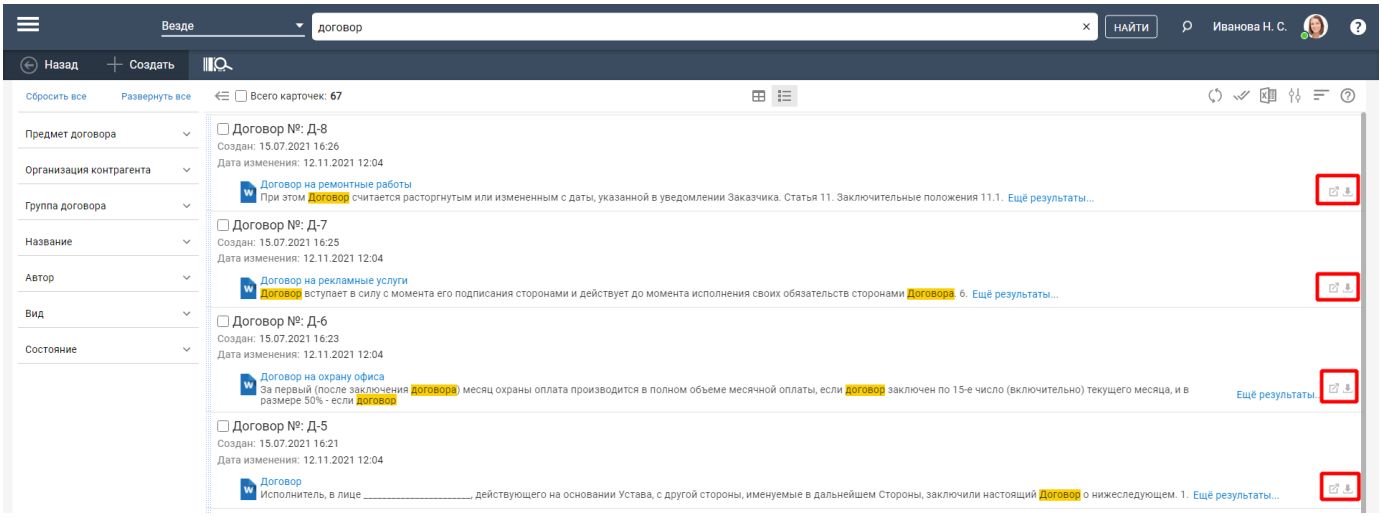


Рисунок 125. Кнопки напротив результатов поиска

Справа от выводимого результата отображаются кнопки, позволяющие скачать: или открыть (для предварительного просмотра) файл: . Также файл можно открыть, нажав на его название. Открывать или скачивать можно только файлы офисного формата (Word или Excel).



Цитаты во вложениях могут быть объединены с агрегацией в гриде.

Агрегация

Работа с функцией агрегации может быть ограничена администратором, см. пункт "Настройки агрегации".

Новый грид позволяет настроить агрегацию строк. Агрегация — сбор суммарных данных — позволяет получать численные результаты по данным, содержащимся в строках таблицы. Например, в представлении может отображаться количество документов определённого вида или количество документов, отправленных определённому пользователю.

Агрегаты отображаются в виде флагов в дополнительной области.

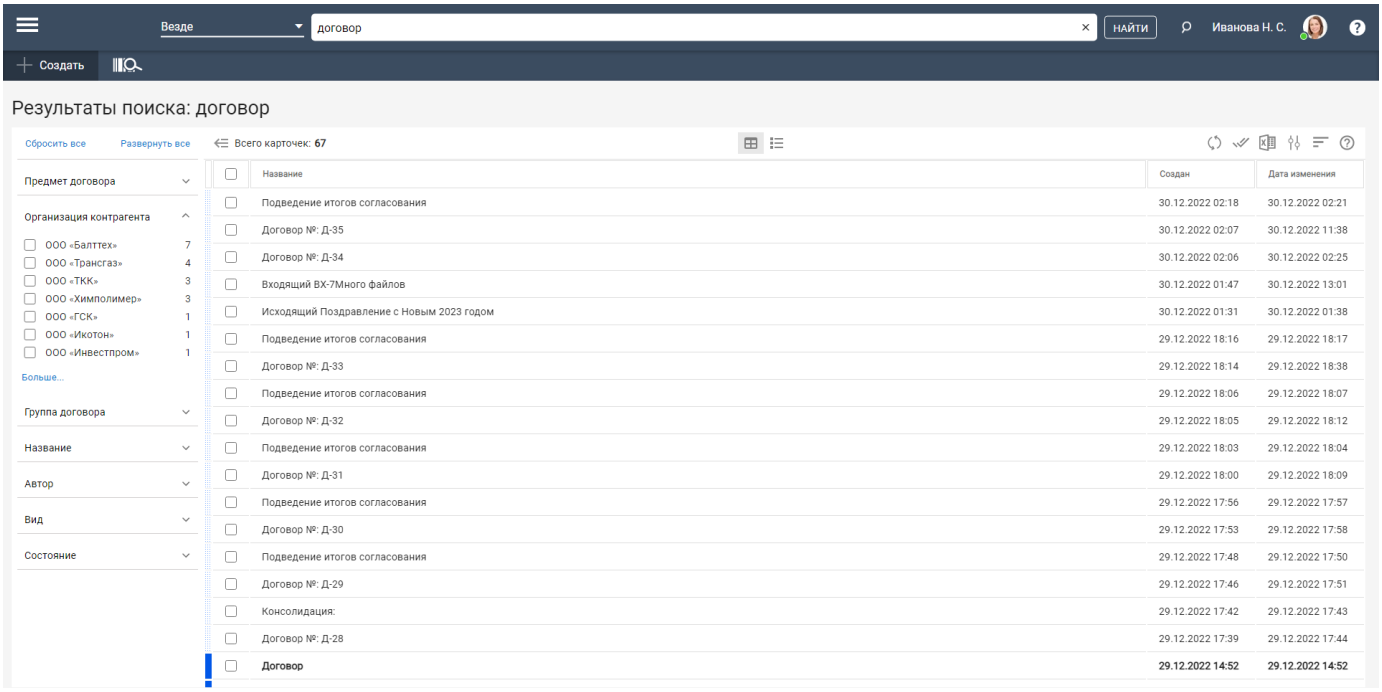


Рисунок 126. Дополнительная область с настроенными агрегатами

Агрегаты работают, только когда подключен и настроен полнотекстовый поиск Elasticsearch.

Агрегаты не отображаются в дополнительной области, если:

- В лицензии отсутствует опция *Docsvision Модуль Интеллектуальный поиск*.
- Elasticsearch включён, но не настроены агрегаты в программе Консоль настройки Docsvision.
- Elasticsearch включён, агрегаты не найдены.
- Elasticsearch выключен.
- Использование агрегатов ограничено администратором, см. пункт "Настройки агрегации".

Настройка агрегации при использовании Elasticsearch описана в документации модуля Платформа.

Как работает агрегация

Агрегаты отображаются в виде флагов в дополнительной области представления. Чтобы применить один или несколько агрегатов, установите напротив них флаги, а затем нажмите кнопку **Показать**. Кнопка может не отображаться в зависимости от настроек администратора.

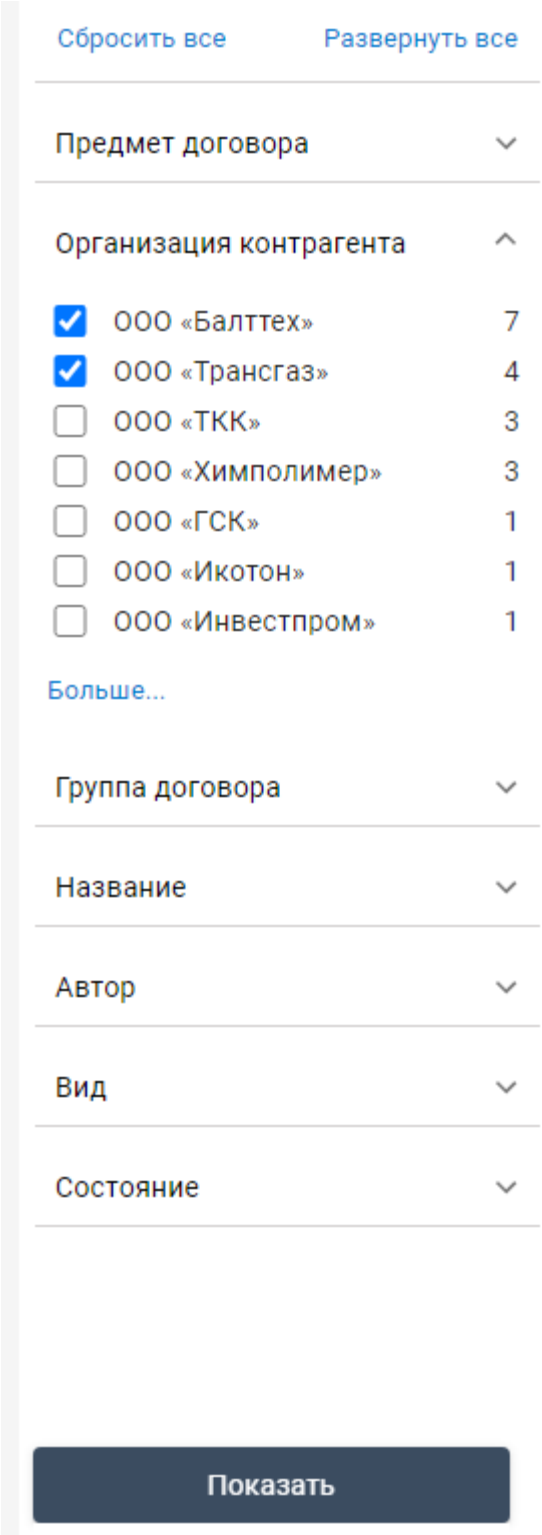


Рисунок 127. Кнопка "Показать"

По умолчанию после выполнения поиска выводятся все результаты поиска и найденные агрегаты, но ни один флаг не установлен. Когда хотя бы в одном агрегате выставляется флаг, результаты поиска фильтруются с учётом выбранных агрегатов.

Для выбранных агрегатов могут быть доступны другие агрегаты или категория

агрегатов. Если другие агрегаты или категория агрегатов неприменимы, их строки становятся неактивными.

При выборе агрегата, помимо основного фильтра по значению агрегата, автоматически добавляется фильтр по типу карточек, для которых в настройках агрегата установлены поля. При применении нескольких агрегатов будут отображены только те типы карточек, которые присутствуют в настройках всех выбранных агрегатов.

Агрегаты считаются только для типов карточек, для которых задан этот агрегат. То есть счетчики могут быть не совсем достоверны: в поиске может отображаться 2 элемента с встречающимся названием, но реальное количество будет равно 1. Такое поведение обусловлено настройкой фасетов Elasticsearch.



Поиск с использованием агрегатов не учитывает проверку прав. Значения количества карточек в агрегатах могут отличаться от фактического количества карточек в представлении.

Пользователь, выполняющий поиск с агрегатами, может получить статистику по карточкам, на которые у него нет прав.

Отображение агрегатов в представлении

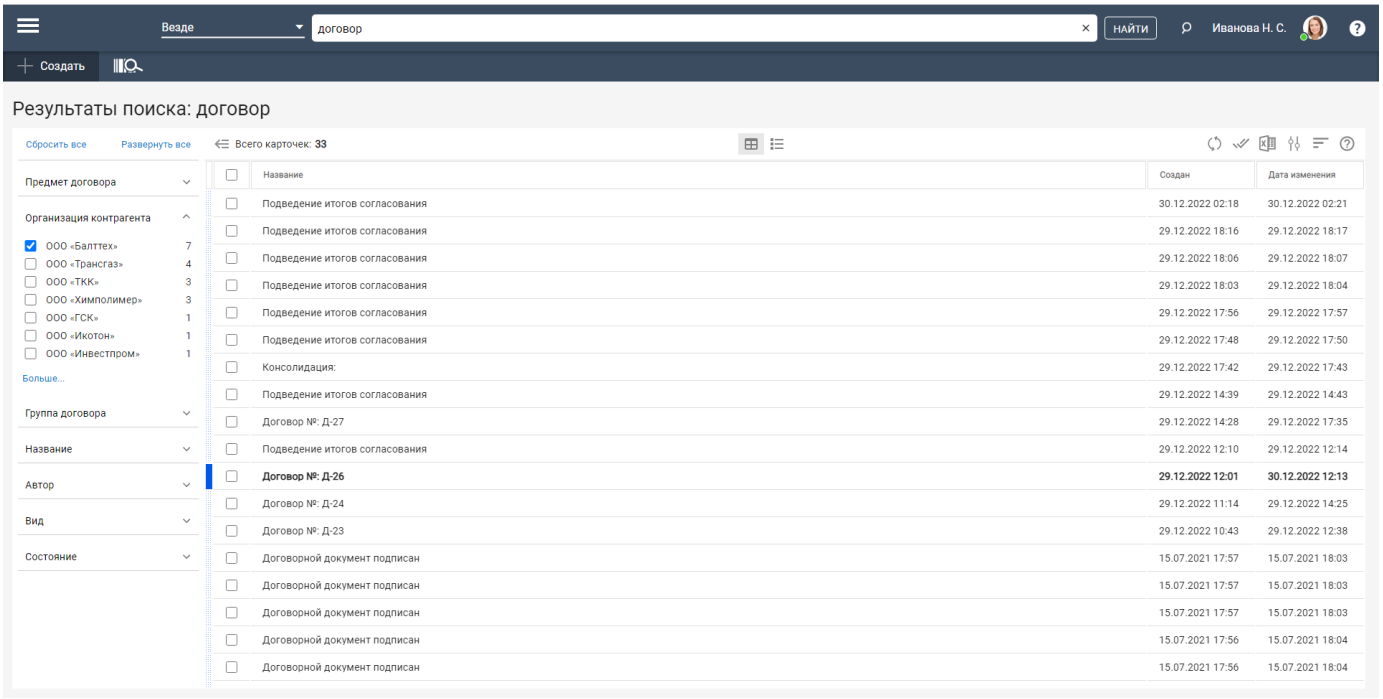


Рисунок 128. Представление с применённой агрегацией

В представлении выводятся все агрегаты для найденного типа карточки, за исключением агрегатов, в котором не нашлось ни одного значения. Агрегаты

отображаются в дополнительной области грида слева.

Агрегат с пометкой *главный* выводится первым в списке, все остальные агрегаты выводятся в порядке, указанном в Консоли настроек Docsvision. Если количество различных вариантов в агрегате превышает 7 шт., появляется ссылка-кнопка **Ещё**. Нажмите на **Ещё**, чтобы отобразить следующие 7 агрегатов. Когда все результаты загружены, вместо кнопки **Ещё** отображается кнопка **Скрыть**, которая скрывает результаты до исходных 7 вариантов.

Результаты агрегатов отсортированы по количеству найденных совпадений от большего к меньшему.

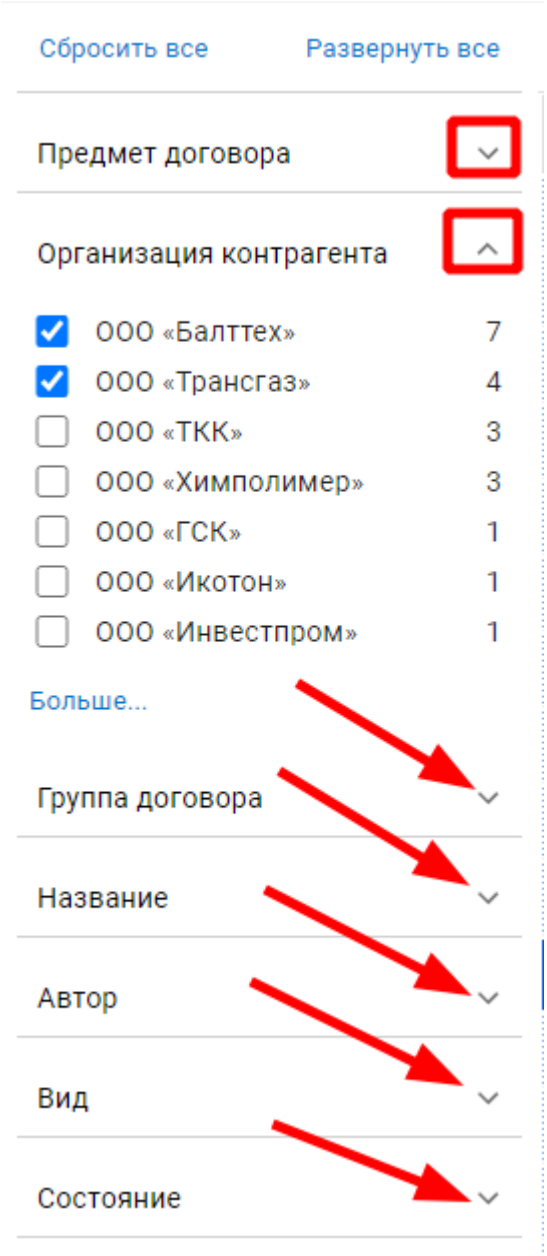


Рисунок 129. Кнопки свернуть/развернуть агрегаты

Каждый агрегат имеет кнопку **Свернуть/Развернуть**.

Новые настройки агрегации могут быть применены только в случае полной перенастройки полнотекстового поиска. Для настройки агрегации сперва нужно отключить БД от полнотекстового поиска, удалить индексы, изменить настройки агрегатов, затем настроить всё заново.

Изменить текущее представление папки

Представление папки определяет то, как отображаются данные карточки при просмотре папки. С особенностями отображения данных в мобильной версии можно ознакомиться в пункте Область представления папки.

Чтобы изменить представление папки:

1. Перейдите в любую папку.
2. Нажмите кнопку , чтобы открыть окно настроек грида.
3. Выберите представление из списка доступных.

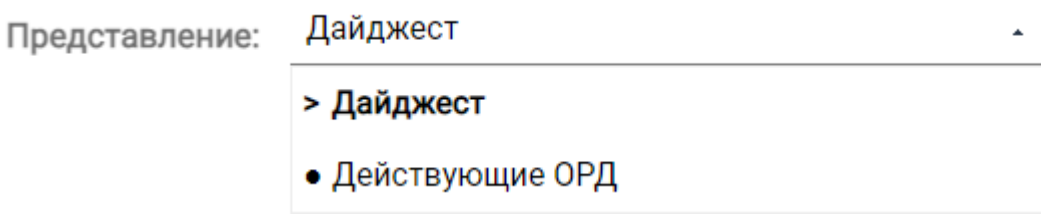


Рисунок 130. Выбор представления папки

Если в списке отсутствует представление, добавленное администратором, нажмите кнопку на панели управления грида.

Экспортировать данные представления в Excel

Пользователь может экспортировать данные представления в файл формата `.xlsx` (Microsoft Excel).

Что включено в экспорт

При экспорте будут выгружены все страницы представления (не только текущая).

Следующие настройки представления будут включены в экспортируемый файл:

- Установленная сортировка.

- Условия фильтрации (экспортируются только отфильтрованные строки).
- Группировка.
- Правила заливки.
- Прочие пользовательские настройки.

Выгрузка представления

Чтобы экспортировать представление:

1. Выберите папку.
2. Выберите представление папки.
3. Если требуется, настройте сортировку и фильтр (недоступно для представления *Дайджест*).
4. Нажмите кнопку  на панели инструментов представления.

> Сформированный файл `.xlsx` будет загружен на компьютер пользователя в стандартный каталог загрузки веб-браузера.
>
> Название файла соответствует названию представления, для которого была вызвана функция выгрузки данных.

Работа с карточками

Основные понятия для работы с карточками

Разметка карточки

Работа пользователя с данными карточек в Web-клиенте осуществляется с помощью настраиваемых форм, которые в системе Docsvision называются *разметками*.

Разметки карточек могут различаться в зависимости от типа и вида карточек, их состояния, роли пользователя при работе с ними, способа открытия и типа пользовательского устройства.

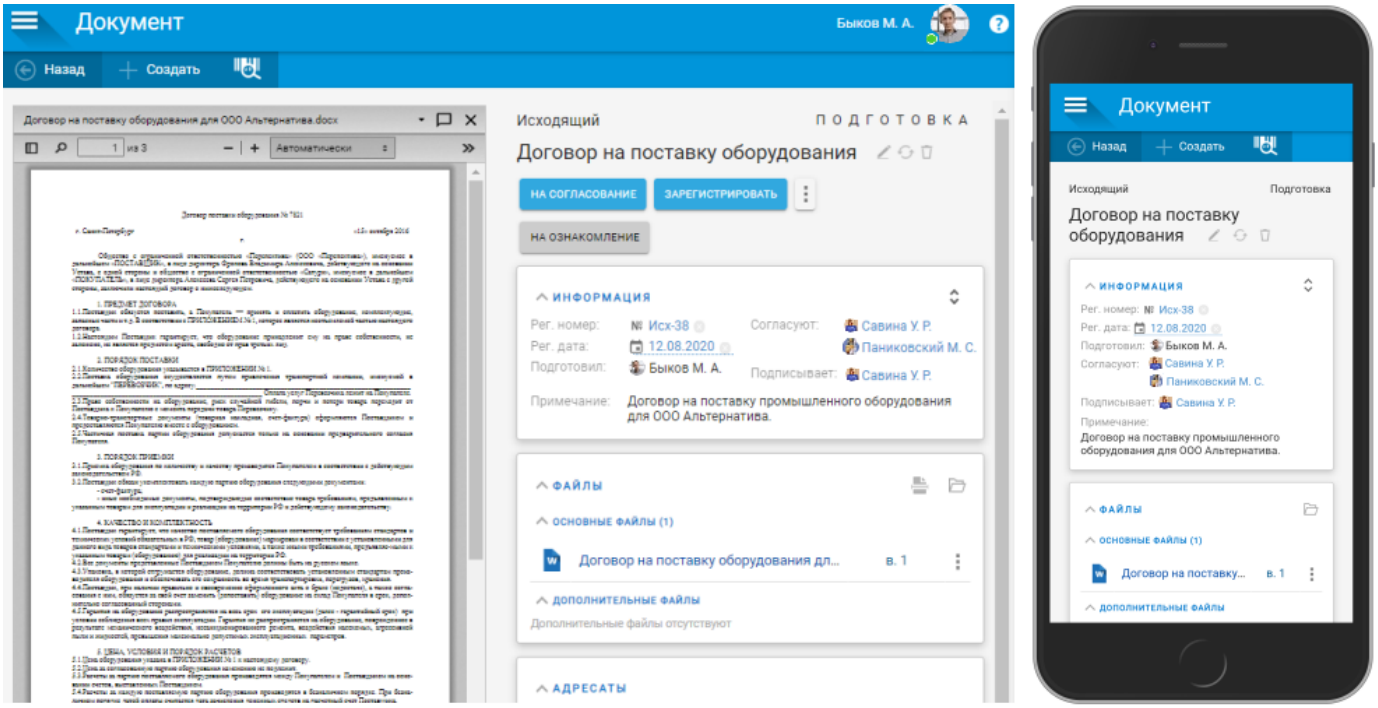


Рисунок 131. Пример разметки исходящего документа в полной и мобильной версиях

Разметка карточки может содержать различные элементы (управления), предназначенные для ввода и отображения информации.

Способ взаимодействия с элементом управления зависит от текущего режима редактирования.

Количество элементов управления в карточке может отличаться, на это влияет режим открытия карточки.

Состояние карточки

Состояние карточки определяет текущий этап работы с ней и может влиять на возможность выполнения определённых операций и внешний вид карточки.

Состояние отображается в верхней части карточки в режиме просмотра.

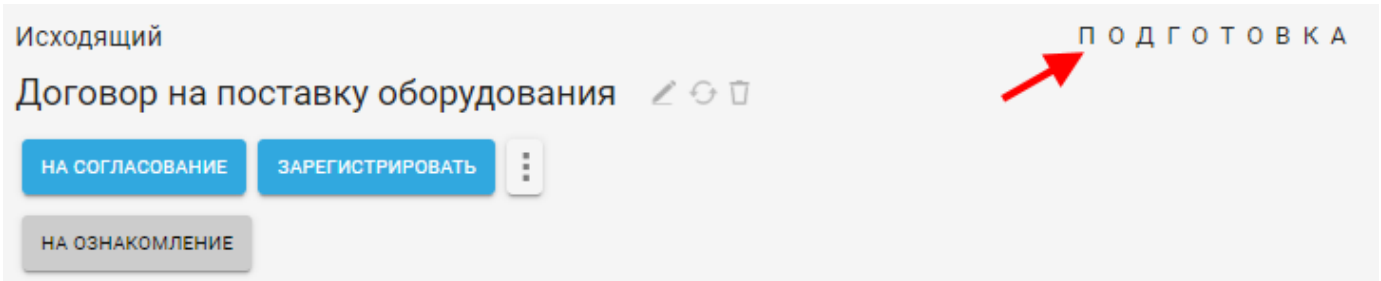


Рисунок 132. Состояние карточки "Документ"

Режимы редактирования карточки

Способ работы с элементом управления зависит от его типа и режима редактирования, указанного в его настройках.

Доступны три режима редактирования:

Только чтение

Используется только для отображения текущего значения без возможности изменения.

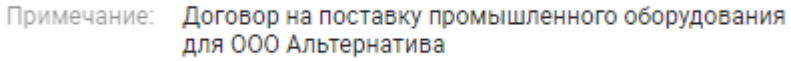


Рисунок 133. Элемент разметки в режиме редактирования "Только чтение"

Редактирование

Используется для ввода или изменения значения.

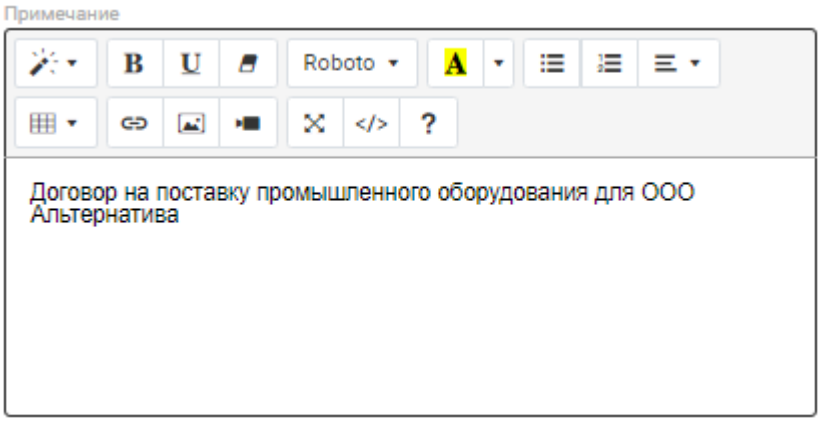


Рисунок 134. Элемент разметки в режиме редактирования "Редактирование"

Редактирование по месту

Совмещает два других режима. В элементе управления отображается текущее значение без возможности прямого изменения.

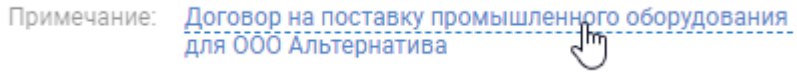


Рисунок 135. Элемент разметки в режиме редактирования "По месту"

Изменить значение любого элемента можно нажатием на текущее значение. После нажатия элемент перейдёт в режим редактирования.

> Исключение — элемент управления `Текст` с HTML-форматированием, данный элемент редактируется в отдельной форме.

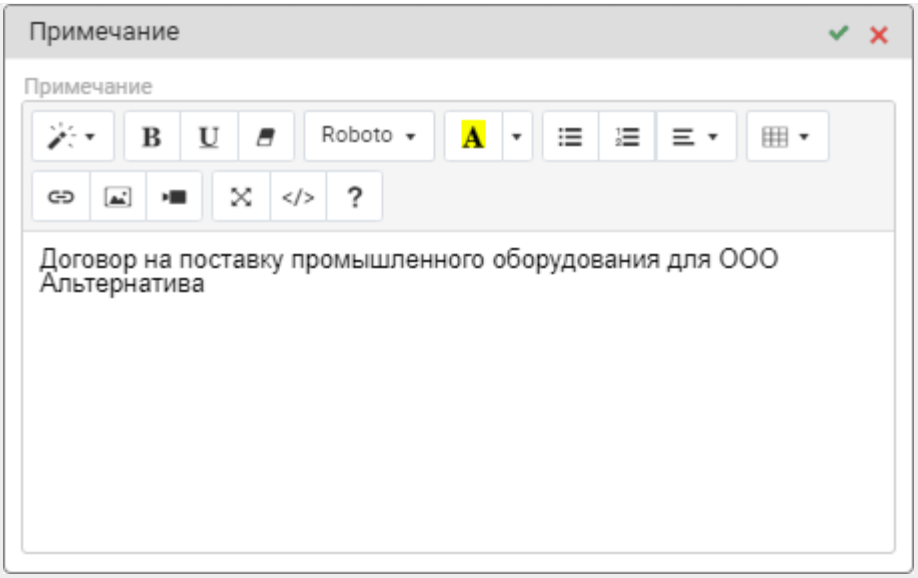


Рисунок 136. Форма изменения значения элемента управления

Изменение осуществляется аналогично изменению в режиме ***Редактирование***.

Чтобы сохранить изменения, нажмите на пустую область или кнопку ✔ (для ЭУ Текст).

Операции с карточками

После ввода карточки в систему пользователи могут изменять её данные и выполнять операции, предусмотренные последовательностью работы с карточкой.

Изменение данных карточки доступно, только если пользователь обладает правами на редактирование и состояние карточки позволяет её изменять.

Доступность операций определяется состоянием карточки и правами пользователя.

В полной версии Web-клиента список операций отображается в верхней части экрана, а в мобильной — в нижней. Часть доступных операций может быть скрыта. Панель со всеми доступными операциями отображается при нажатии на кнопку ⋮.

Область с операциями карточки далее будет называться *Панель операций*.

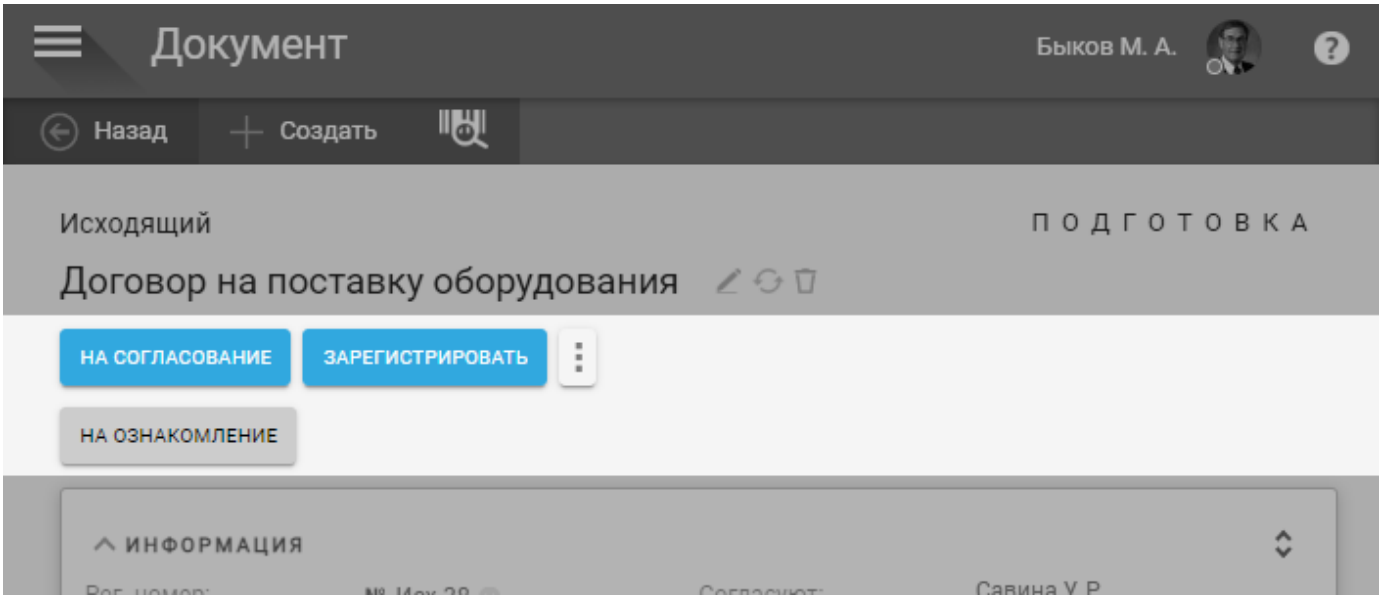


Рисунок 137. Панель операций карточки Документ

Панель операций включает два набора команд:

Основные операции с карточкой

Всегда отображаются первыми. Данные операции предусмотренные бизнес-логикой карточки: отправка на согласование, регистрация, отправка задания на исполнение и т. п.

Дополнительные операции

Отправка документа на ознакомление, создание входящего к исходящему, просмотр истории (доступен ограниченному кругу пользователей) и др.

Операции с карточками рассмотрены подробно в соответствующих разделах.

Режимы открытия карточки

Открыть карточку в режиме чтения

Карточки Web-клиента всегда открываются в одном из двух режимов открытия: *чтение* или *редактирование*.

Чтобы открыть карточку в режиме чтения

1. Откройте папку, содержащую карточку.
2. Нажмите на ссылку карточки.

Карточка будет открыта в режиме чтения.

Режим чтения карточки

Нажатие на карточку откроет её в режиме чтения. В данном режиме недоступно изменение большинства полей карточки, а доступные изменения осуществляются в режиме редактирования По месту.

Открыть карточку в режиме редактирования

Чтобы открыть карточку в режиме редактирования

1. Откройте папку, содержащую карточку.
2. Нажмите на ссылку карточки.
3. Нажмите на кнопку ✎ в строке с названием карточки.

Карточка будет открыта в режиме редактирования.

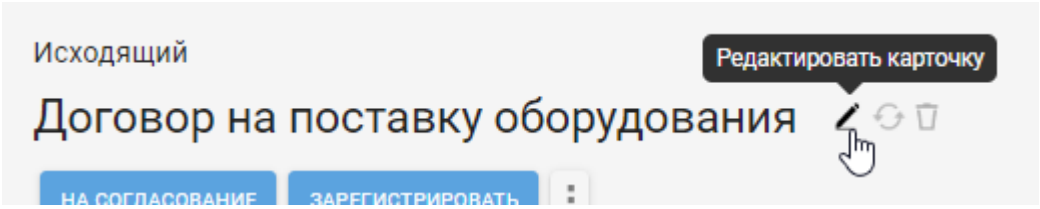


Рисунок 138. Кнопка редактирования карточки

При открытии карточки в режиме редактирования Web-клиент не устанавливает блокировку автоматически. Блокировка может быть установлена при открытии карточки для редактирования, например, в Windows-клиенте. Блокировку также можно реализовать в собственном решении на базе Web-клиента.

Режим редактирования карточки

В карточке, открытой в режиме редактирования, доступны основные возможности изменения карточки. Создание карточки также осуществляется в режиме редактирования.

В зависимости от режима открытия карточки набор элементов в разметке может отличаться.

Недоступность редактирования

Кнопка ✎ будет недоступна, если редактирование карточки невозможно в следующих случаях:

- У пользователя отсутствуют права на редактирование карточки.
- Карточка находится в состоянии, которое запрещает редактирование.

Заблокированная другим пользователем карточка также не может быть открыта для редактирования. При попытке открыть заблокированную карточку будет показано сообщение: `Карточка уже заблокирована пользователем⋯`.

Создание новой карточки

1. Нажмите кнопку **Создать** панели меню. Будет открыто окно выбора способа создания карточки.

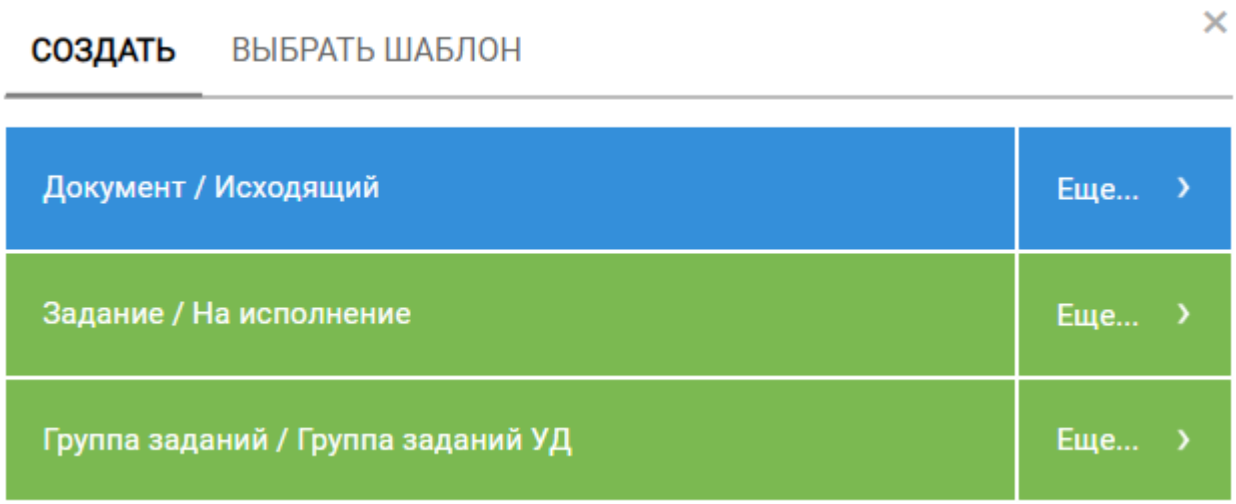


Рисунок 139. Создание карточки

2. Перейдите на вкладку *Создать*.
3. Щелкните по строке создаваемого вида карточек.

   Для каждого типа карточек (через /) указывается создаваемый по умолчанию вид карточек, который может быть назначен администратором Web-клиента.

   Чтобы создать карточку другого вида, нажмите кнопку **Ещё...** в строке создаваемого типа и выберите нужный вид из открывшегося дерева.

   После выбора вида будет открыто окно создания новой карточки соответствующего типа или вида. Подробности в пунктах:

   - Ввод документа в систему
   - Создание задания

- Создание группы заданий



Обратите внимание, что в случае принудительного закрытия браузера (например, при завершении работы ОС) с открытыми и несохранёнными карточками документов, их черновики будут сохранены в Windows-клиенте.

Создание карточки по шаблону

Чтобы создать карточку на основе существующей шаблонной карточки:

1. Нажмите кнопку **Создать** панели меню. Будет открыто окно выбора способа создания карточки.
2. Перейдите на вкладку *Выбрать шаблон*.

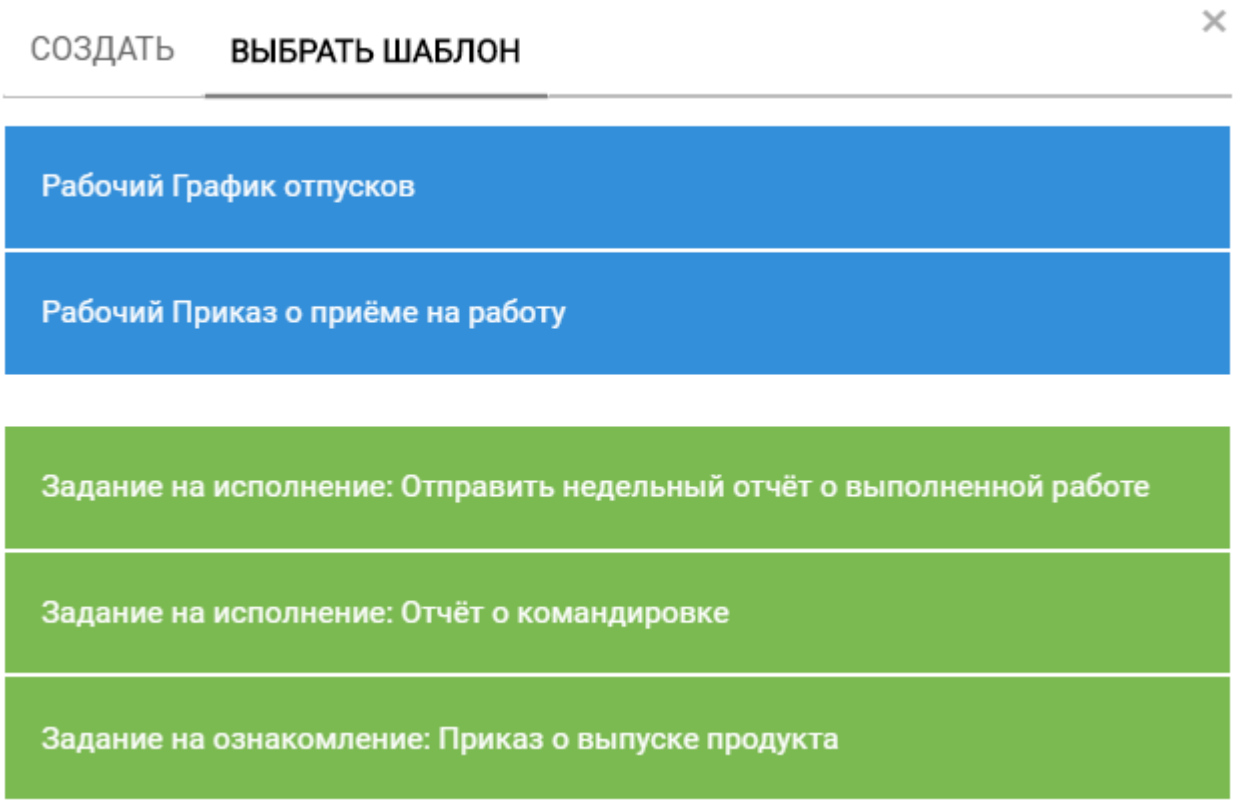


Рисунок 140. Создание карточки по шаблону

На вкладке *Выбрать шаблон* представлены шаблоны всех типов карточек.



Шаблоны карточек настраиваются в Windows-клиенте.

1. Щелкните по строке требуемого шаблона.

Будет открыто окно создания новой карточки соответствующего типа или

вида, для которой могут быть автоматически заполнены данные из шаблона.

Обновление содержимого карточки

1. Откройте карточку в режиме чтения.
2. Нажмите кнопку ↻ в строке с названием карточки.

Карточка будет перезагружена.

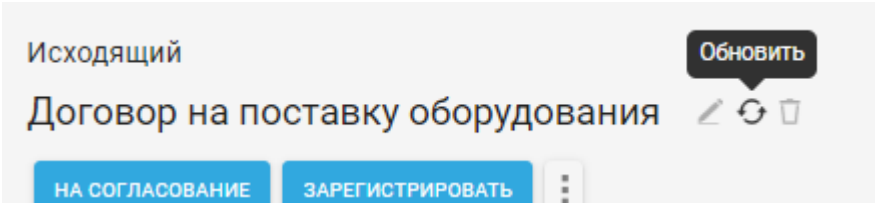


Рисунок 141. Кнопка обновления содержимого карточки

Удаление карточки

1. Откройте карточку в режиме чтения.
2. Нажмите кнопку 🗑 в строке с названием карточки.

Будет предложено подтвердить удаление.

3. Нажмите **ОК** для подтверждения.

После удаления будет открыта страница Дашборд.

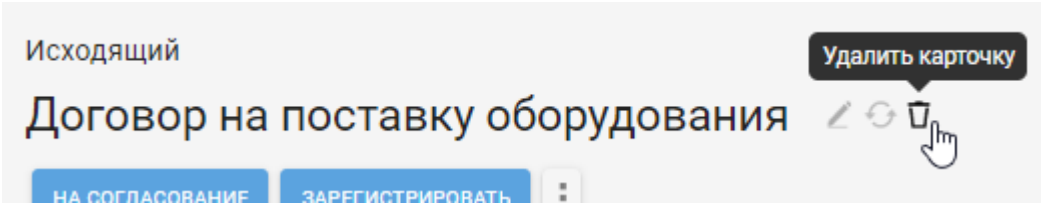


Рисунок 142. Кнопка удаления карточки

Недоступность удаления

Кнопка 🗑 будет недоступна, если удаление карточки невозможно в следующих случаях:

- У пользователя отсутствуют права на удаление карточки.
- Карточка находится в состоянии, которое запрещает удаление.
- Возможны случаи, когда кнопка отображается, но операция удаления недоступна, если карточка связана сильной ссылкой.

Заблокированная другим пользователем карточка также не может быть удалена. При попытке удалить заблокированную карточку будет показано сообщение: `Карточка уже заблокирована пользователем···`.

Удалить карточку можно также из папки при помощи групповых операций грида.

Карточки удаляются в корзину или безвозвратно в зависимости от настроек в файле конфигурации Web-клиента.

Просмотр истории изменений карточки

В истории карточки фиксируются все изменения карточки, а также выполняемые операции. Возможность просмотра истории доступна только пользователям из группы *Администраторы УД*.

Чтобы посмотреть историю изменений карточки, выполните следующие действия:

1. Откройте карточку в режиме чтения.
2. Нажмите кнопку **История** на Панели операций. Будет открыто диалоговое окно *История карточки*.

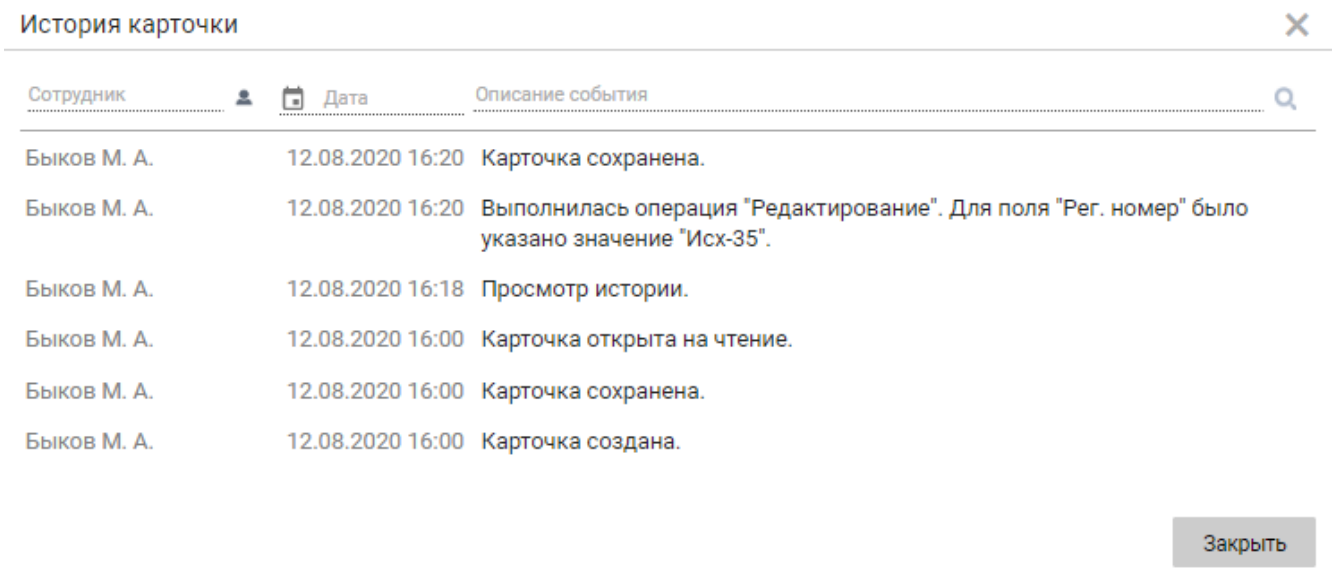


Рисунок 143. История карточки

События карточки сортируются по дате от самого недавнего к самому позднему.

Для каждого события указывается сотрудник, который инициировал событие, дата и описание.

Над списком событий отображаются фильтры. С помощью фильтров можно выбрать события, инициированные определённым пользователем, в указанную дату и/или с определённым описанием. Фильтры применяются автоматически при вводе или выборе значения.



При фильтре по дате время события не учитывается.

При фильтре по описанию может быть введена часть искомого текста описания.

Избранные карточки

Карточки можно добавить в избранные.

Чтобы добавить карточку в избранное:

1. Откройте любую карточку.
2. Найдите рядом с заголовком значок звёздочки.
3. Нажмите не значок.

   Значок заполнится жёлтым. Это означает, что карточка добавлена в избранное.

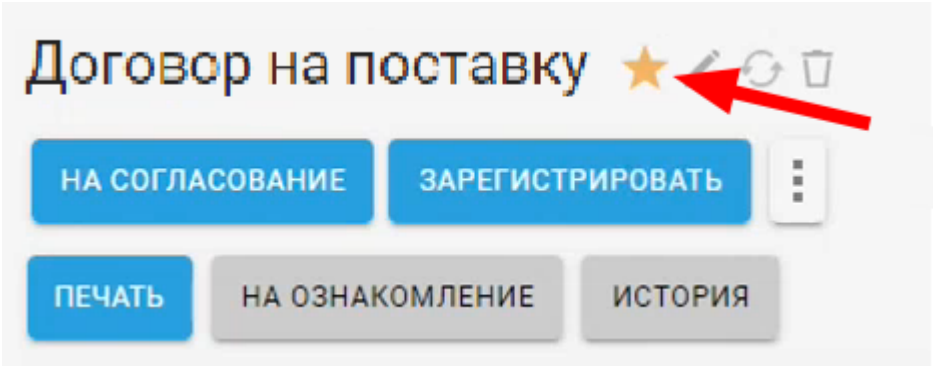


Рисунок 144. Значок "избранное"

4. Повторное нажатие на значок убирает карточку из избранного.

Папка "Избранное"

Все отмеченные звёздочкой  карточки сохраняются в папке *Избранное*. Перейти в папку можно из любого места, открыв ссылку в боковом меню.

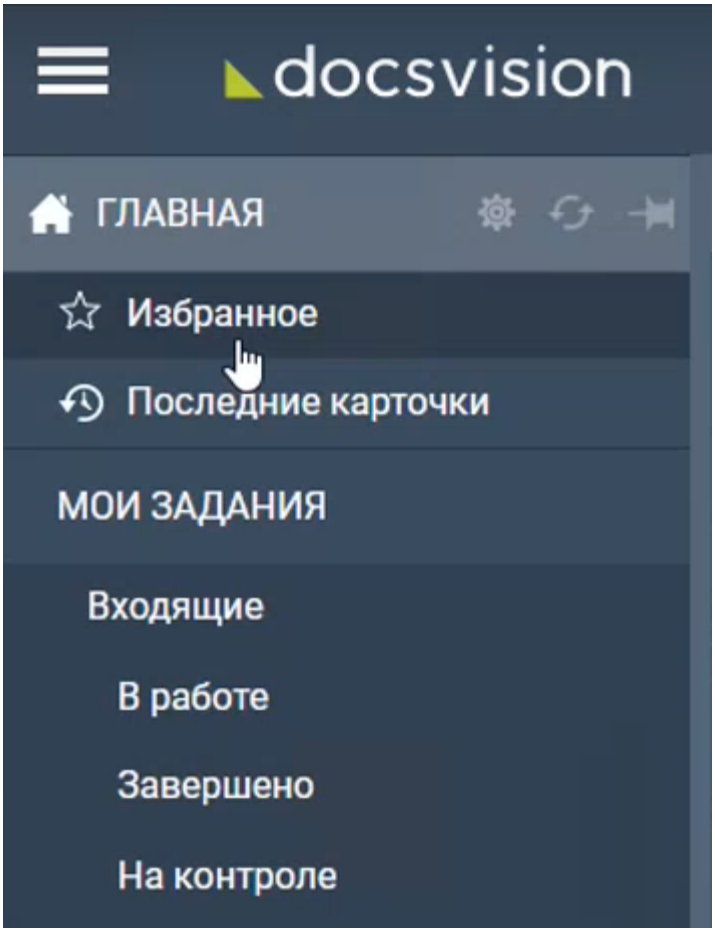


Рисунок 145. Папка "Избранное"

Открыть страницу сообщений карточки

Чтобы перейти на страницу сообщений карточки:

1. Откройте карточку в режиме чтения.
2. Нажмите кнопку в строке с названием карточки.

   Кнопка не отображается в следующих случаях:

   - Если текущий пользователь не является членом группы **Docsvision Management Console Administrators** на сервере Docsvision или сервере Консоли управления Docsvision.
   - Если карточка не поддерживает работу с сообщениями. Исключение — карточка Документа.

     Для карточки документа в отдельных вкладках будут открыты все страницы сообщений карточек согласования.

   - Если поле `MessagesId` карточки — пустое, то есть не было разослано

сообщений.

3. Будет открыта страница Сообщений карточки.

Работа с архивными карточками

Не следует путать архивацию карточек с отправкой документа в архив.

Любая карточка в любом состоянии считается архивированной, если в свойствах карточки установлен атрибут **Архивирована** (подробнее см. документацию модуля Windows-клиент, раздел "Перенос карточки в архив"). Архивировать карточку средствами Web-клиента невозможно.

Архивные карточки по умолчанию не могут быть изменены и не отображаются в представлениях. Пользователь может настроить отображение архивных карточек в настройках профиля пользователя.

Если отображение архивных карточек включено, такие карточки будут обозначаться специальным значком и отображаться в результатах поиска вместе с файлами.

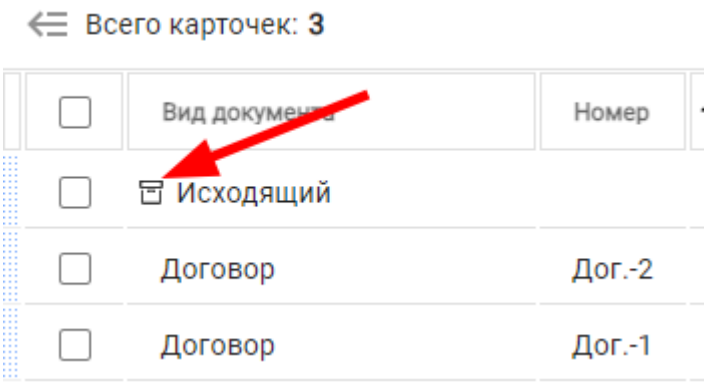


Рисунок 146. Индикация архивированной карточки

Использование поиска по карточкам

Для ввода поискового запроса используется Поле поиска.

Чтобы очистить результаты поиска, нажмите на крестик ✕ в строке поиска.

Пользователям Web-клиента доступны различные варианты поиска карточек.

Поиск по папке

Поиск по текущей папке используется, если отключен полнотекстовый поиск. В этом случае выполняется поиск карточек открытой папки, в которых

встречается указанная строка.

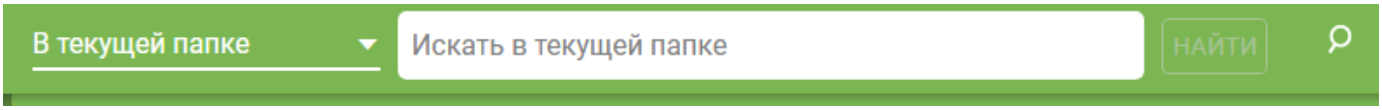


Рисунок 147. Поиск по папке

Чтобы осуществить поиск по открытой папке, введите в строке поиска один или несколько символов искомого слова. Поиск карточек будет выполнен после непродолжительной паузы. При повторном открытии данной папки будет применен ранее использованный поисковый запрос.

Чтобы сбросить поисковый запрос: нажмите крестик в строке поиска. В качестве альтернативы, вы можете стереть все символы, введённые в строку поиска, и нажать `Enter` или кнопку **Найти**.



Текущий поисковый запрос будет также сброшен вместе с пользовательскими настройками папки.

Полнотекстовый поиск



Результаты полнотекстового поиска всегда сохраняются при обновлении страницы представления.

Если включено автоматическое обновление представления, появляется всплывающее уведомление о том, что в папке есть новые карточки.

Полнотекстовый поиск от Microsoft SQL работает по **префиксу**. Соответственно, будут найдены только строки, где слово **начинается** с части введённой в поисковую строку.

Пользователь Web-клиента может осуществлять поиск карточек по содержимому с помощью полнотекстового поиска. Полнотекстовый поиск становится доступным при выполнении следующих условий:

- Полнотекстовый поиск настроен в Docsvision.

  Настройка полнотекстового поиска осуществляются по инструкции, приведённой в документации модуля Платформа.

- Данные карточек проиндексированы.

Полнотекстовый поиск карточек может выполняться среди карточек текущей

открытой папки или среди всех карточек. Область поиска выбирается в строке поиска при соблюдении следующих правил:

- На главной странице доступен только режим поиска среди всех карточек.
- На странице папки доступны варианты поиска:
  - По всем карточкам,
  - Карточкам текущей папки,
  - Карточкам папки и её подпапок.

Чтобы осуществить полнотекстовый поиск:

1. Введите в строке поиска часть слова или фразу искомого текста.
2. Выберите область поиска:

В текущей папке

Область поиска доступна для выбора только на страницах *стандартных* или *виртуальных* папок.

При выполнении поиска в дочерней папке, поиск в родительской папке выполнен не будет.

В поле поиска для данного варианта отображается подсказка: `Искать в текущей папке`.

Включая подпапки

Область поиска доступна для выбора только на страницах *стандартных* или *виртуальных* папок, у которых имеются дочерние папки.

В поле поиска для данного варианта отображается подсказка: `Искать в текущей папке, включая стандартные подпапки`.

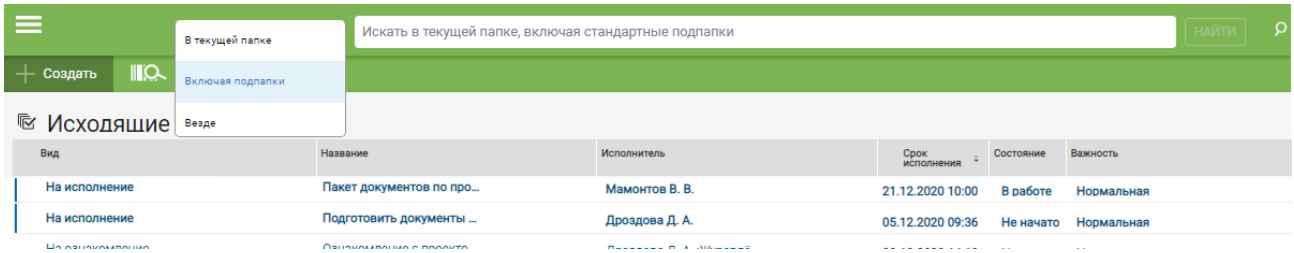


Рисунок 148. Выбор области поиска

В найденном

Область поиска доступна для выбора в папке *Поиск* при наличии ранее найденных карточек в данной папке.

В поле поиска для данного варианта отображается подсказка: `В найденном`. После поиска в найденном, результаты поиска отображаются как папка с заголовком `Результаты поиска: ⋯ → поисковый-запрос`.

Везде

Область поиска доступна для выбора на всех страницах Web-клиента.

При выборе данного варианта поиск карточек будет выполняться по всей базе данных.

В поле поиска для данного варианта отображается подсказка: `Везде`.

Результаты поиска будут отображаться как отдельная папка с представлением *Результаты поиска*. Результаты поиска сохранятся в том же виде, даже после открытия карточки или обновления страницы.

3. Нажмите клавишу `Enter` или нажмите кнопку **Найти**, чтобы отправить поисковый запрос на сервер.

Особенности полнотекстового поиска:

- Когда выбрана область поиска ***В текущей папке***, ***Включая подпапки***, результаты поиска отображаются как папка с заголовком `Название-текущей-папки (результаты поиска: поисковый-запрос)`.
- Если область поиска ***В текущей папке***, ***Включая подпапки*** или ***В найденном*** и поисковый запрос был удалён нажатием на крестик ✕ в строке поиска, строка поиска будет очищена и откроется исходная папка, из которой был запущен поиск.
- В папках с поисковыми запросами доступны обычные функции грида, включая групповые операции.
- Если в папке с применённым поисковым запросом применена группировка или фильтрация, кнопка **Показать все**, сбрасывает только фильтрацию или группировку, но не сбрасывает поисковый запрос.
- Поиск в папке с применённым фильтром или группировкой, отобразит только карточки, найденные в отфильтрованном или сгруппированном. Счётчик количества карточек будет пересчитан.
- Когда поисковая строка пустая, нажатие на кнопку **Найти** отображает всё содержимое папки.

- Если в настройках полнотекстового поиска не установлены ограничения на типы карточек, будет выполнен поиск по всем типам карточкам, иначе только по разрешённым типам.
- Поисковый запрос, используемый для поиска по текущей папке или папке с подпапками, сохраняется и будет использован при повторном открытии данной папки.

Чтобы сбросить поисковый запрос: нажмите крестик в строке поиска. В качестве альтернативы, вы можете стереть все символы, введённые в строку поиска, и нажать `Enter` или кнопку **Найти**.



Текущий поисковый запрос будет также сброшен вместе с пользовательскими настройками папки.

Администратор Docsvision может настроить полнотекстовый поиск с использованием одной из нескольких систем полнотекстового поиска. В зависимости от системы результаты поиска, а также требования к исходным данным могут отличаться. Например, при поиске с использованием системы Elasticsearch есть следующие особенности:

- Может потребоваться ввод большего числа знаков в словах поисковой фразы.
- Результаты поиска могут содержать записи, которые не полностью соответствуют поисковой фразе.
- В некоторых вариантах поисковой фразы Elasticsearch может не возвращать результаты.

Также от выбора системы полнотекстового поиска зависит возможность поиска по различным типам файлов:

- При использовании встроенной в Microsoft SQL Server системы поиска: `.pdf` и файлы поддерживаемого офисного формата;
- При использовании системы полнотекстового поиска Elasticsearch: `.pdf`, файлы поддерживаемого офисного формата (`.docx`, `.doc`, `.xlsx`, `.xls`, `.pptx`, `.ppt`, `.rtf`), `.html` и `.txt`.



Ознакомится с подробной инструкцией по настройке поисковой системы и особенностями работы каждой из них можно в документации модуля Платформа.

Папка "Результаты поиска"

Если выбрана область поиска ***Везде***, результаты поиска будут отображаться как отдельная папка *Результаты поиска*. Результаты поиска не исчезают после открытия карточки или обновления страницы.

В результатах поиска доступны все функции грида, включая групповые операции.



В папке *Результаты поиска*, дополнительная область всегда развёрнута.

- Если для пользователя доступна агрегация, в дополнительной области будут отображаться агрегаты. Группировки будут недоступны.
- Если для пользователя недоступна агрегация, агрегаты не будут отображаться в дополнительной области. Группировки будут доступны.

Когда применена фильтрация, кнопка **Показать все** возвращает к последнему поисковому запросу.

При включённом фильтре, поиск производится только среди отфильтрованного.

В результатах поиска отображаются только карточки, к которым у текущего пользователя есть доступ.

При отсутствии карточек, удовлетворяющих запросу, в представлении отображается текст: `Нет данных для отображения`.

Параметрический (атрибутивный) поиск

Параметрический поиск — вариант поискового запроса, для выполнения которого пользователь должен предварительно настроить параметры запроса с помощью предоставляемой панели настройки.

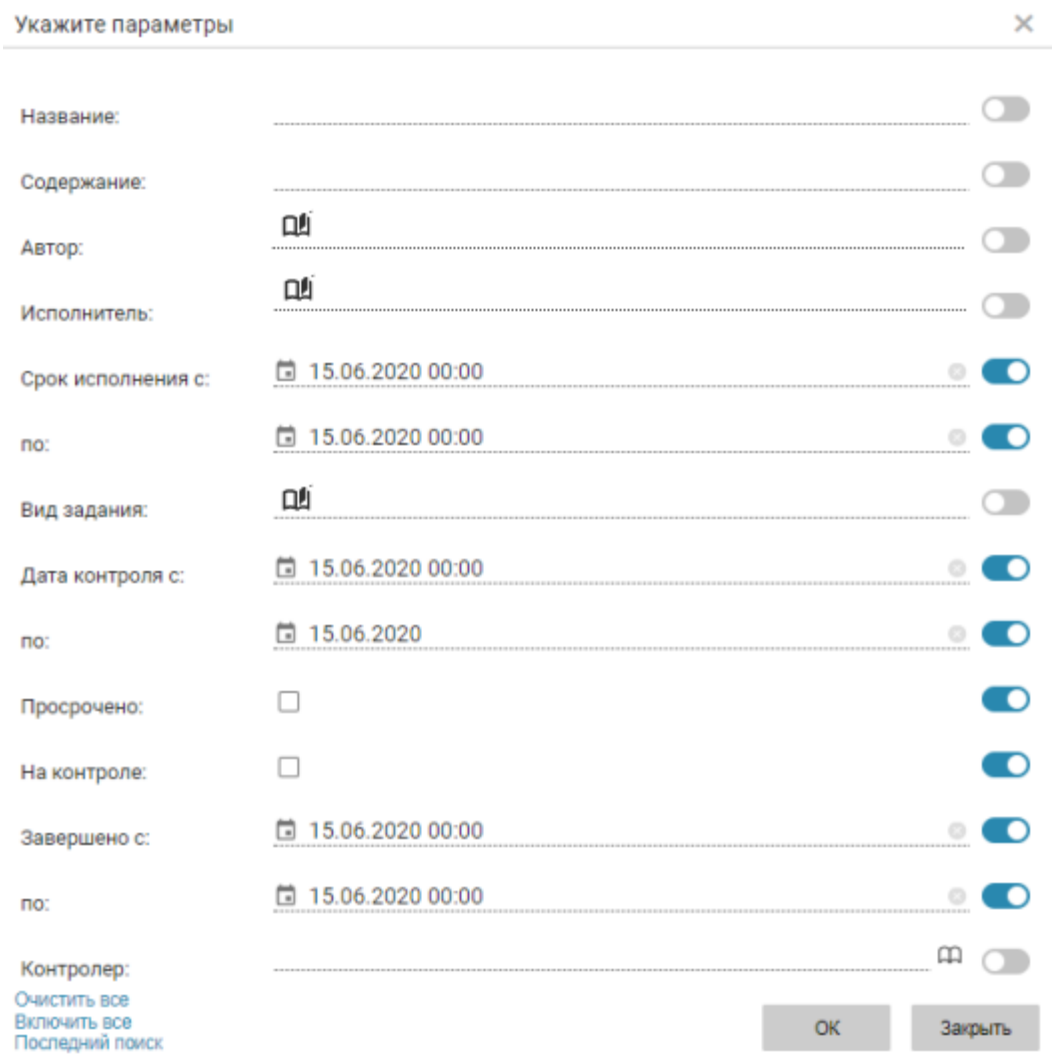


Рисунок 149. Панель настройки параметрического поиска

Панель настройки параметрического поиска содержит:

Список параметров запроса

Значение параметра устанавливается с помощью элемента управления, соответствующего типу параметра.

При наведении курсора мыши на название параметра будет отображена подсказка с условием поиска по данному параметру.

Переключатели использования параметра в запросе	• — параметр используется в запросе. • — параметр не используется в запросе.
Ссылки	• Очистить все. Очищает значения всех параметров и устанавливает все переключатели в значение **Не используется**. • Включить все. Устанавливает все переключатели в значение **Используется**. • Последний поиск. Устанавливает параметры и переключатели в значения, которые были использованы при последнем выполнении данного запроса на данном устройстве.
Ссылка на карточку	Поисковый параметр позволяет выполнять поиск по конкретной ссылочной карточке. Например, можно указать в параметре договор и найти дополнительное соглашение к нему, или родительское задание. Работа параметра настраивается при помощи ЭУ "Ссылка на карточку"
Кнопка Закрыть	Закрывает панель настройки.

Кнопка ОК	Выполняет запрос. Выполнить поисковой запрос можно также нажатием на клавиатуре кнопки `Enter`. Если при этом какое-либо поле находится в режиме выбора, например выбор даты и выбор из справочника, при первом нажатии будет выполнен выбор.

Чтобы настроить и выполнить параметрический запрос:

1. Раскройте в основном меню папку *Договоры* и выберите пункт *Поиск*.
2. Введите или выберите значения параметров, которые должны быть добавлены в поисковый запрос.

   При изменении значения параметра, переключатель использования будет автоматически включён.

   Если значение параметра, включённого в запрос, не выбрано, он будет участвовать в запросе с признаком `значение поля не указано`.
3. Установить переключатели использования параметров, которые не должны использоваться в запросе, в значение .
4. Нажать на кнопку **ОК**.

Поиск карточки по штрих-коду

Чтобы открыть карточку, зная её штрих код:

1. Откройте поле для ввода нажатием кнопки на панели меню.



Рисунок 150. Кнопка и поле для ввода штрих-кода

2. Введите в текстовое поле цифровое значение штрих-кода с помощью клавиатуры или сканера штрих-кодов.

> Найденная карточка будет открыта в новой вкладке веб-браузера.

Если карточка с указанным штрих-кодом не зарегистрирована в Docsvision, будет выдано сообщение: `Не удалось найти карточку с указанным штрихкодом`. Повторите поиск с корректным значением штрих-кода.

Потоковый поиск карточек по штрих-коду

Помимо одиночного режима поиска карточки по штрих-коду, Web-клиент поддерживает потоковый режим, при котором пользователь может открыть одновременно несколько карточек, предварительно указав их штрих-коды. Данный режим поиска должен быть настроен отдельно.

Чтобы выполнить поиск нескольких карточек:

1. Откройте поле для ввода нажатием кнопки  на панели меню.
2. Введите в текстовое поле цифровое значение штрих-кода с помощью клавиатуры или сканера штрих-кодов.

   В правой части поля поиска появится кнопка открытия карточек, в которой будет указано количество штрих-кодов в очереди поиска.



Рисунок 151. Поле ввода штрих-кода с кнопкой открытия карточек

3. Последовательно введите штрих-коды других искомых карточек.
4. Нажмите на кнопку **Открыть карточек: <Кол-во карточек>**.

   В веб-браузере будут открыты вкладки с найденными карточками.

5. Если карточка с указанным штрих-кодом не зарегистрирована в Docsvision, она будет содержать текст: `Не удалось найти карточку с указанным штрихкодом`.

   Повторите поиск с корректным значением штрих-кода.



Технические возможности ограничивают максимальное число вкладок, которые могут быть одновременно открыты в веб-браузере. Если при выполнении потокового поиска возникла ошибка вида `К сожалению, произошла ошибка. Пожалуйста, обратитесь`

> к администратору системы··· Не удалось завершить действие попробуйте выполнить поиск меньшего числа карточек.

Кнопка **X** *в поле ввода штрих-кода очищает:*

- Текущее значение в строке, если оно введено.
- Очередь поиска, если строка поиска пуста.

Работа с карточками типа "Документ"

Карточки типа *Документ* (далее "карточки документов") предназначены для работы с разного рода документами и файлами. Это могут быть входящие и исходящие документы, внутренние документы и др. Работа с документами выполняется с помощью карточек видов:

Входящий

Предназначены для хранения в системе Docsvision документов, поступивших от контрагентов.

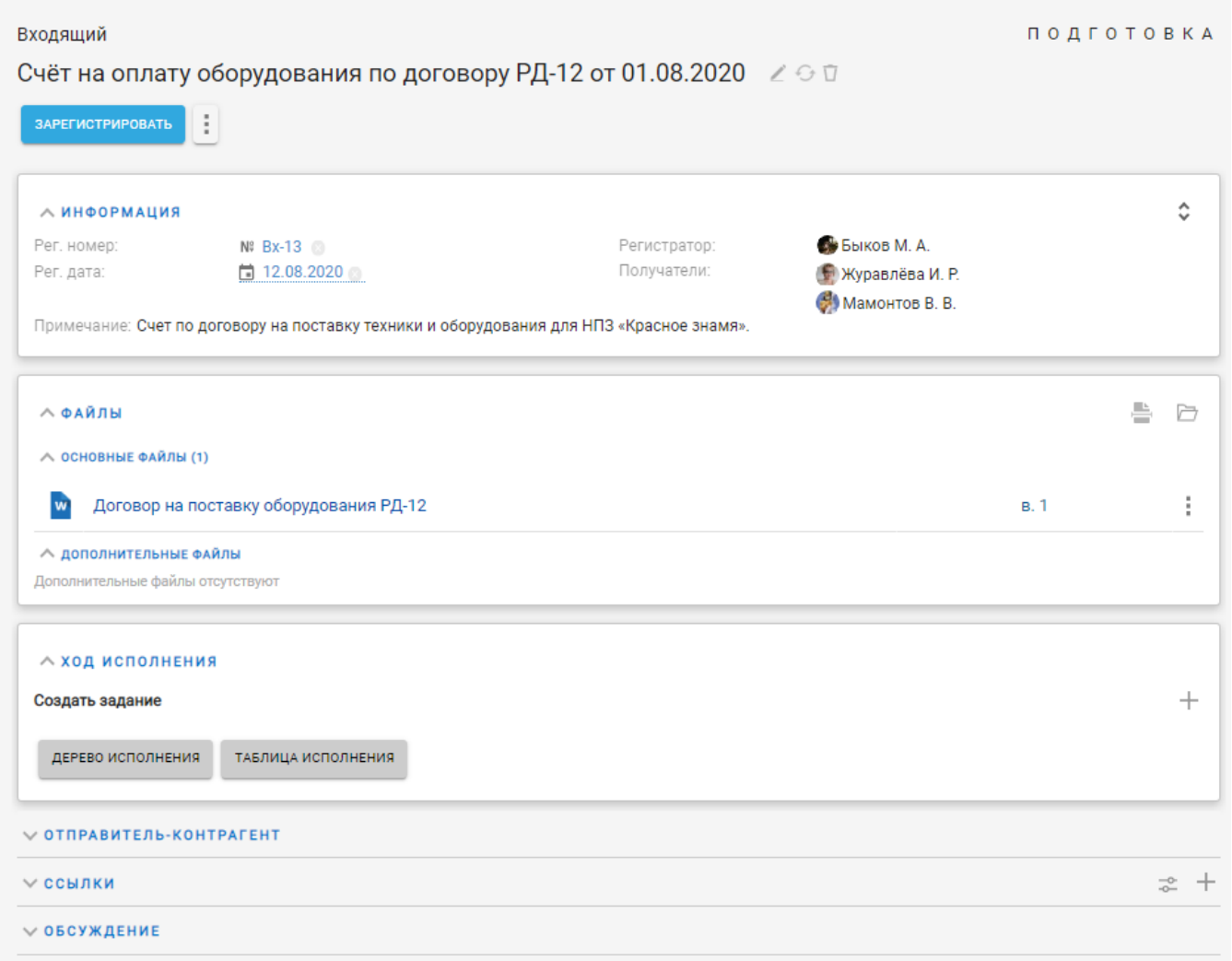


Рисунок 152. Пример входящего документа

Исходящий

Предназначены для хранения в системе Docsvision файлов корреспонденции, которая должна быть передана организациям-партнерам (контрагентам).

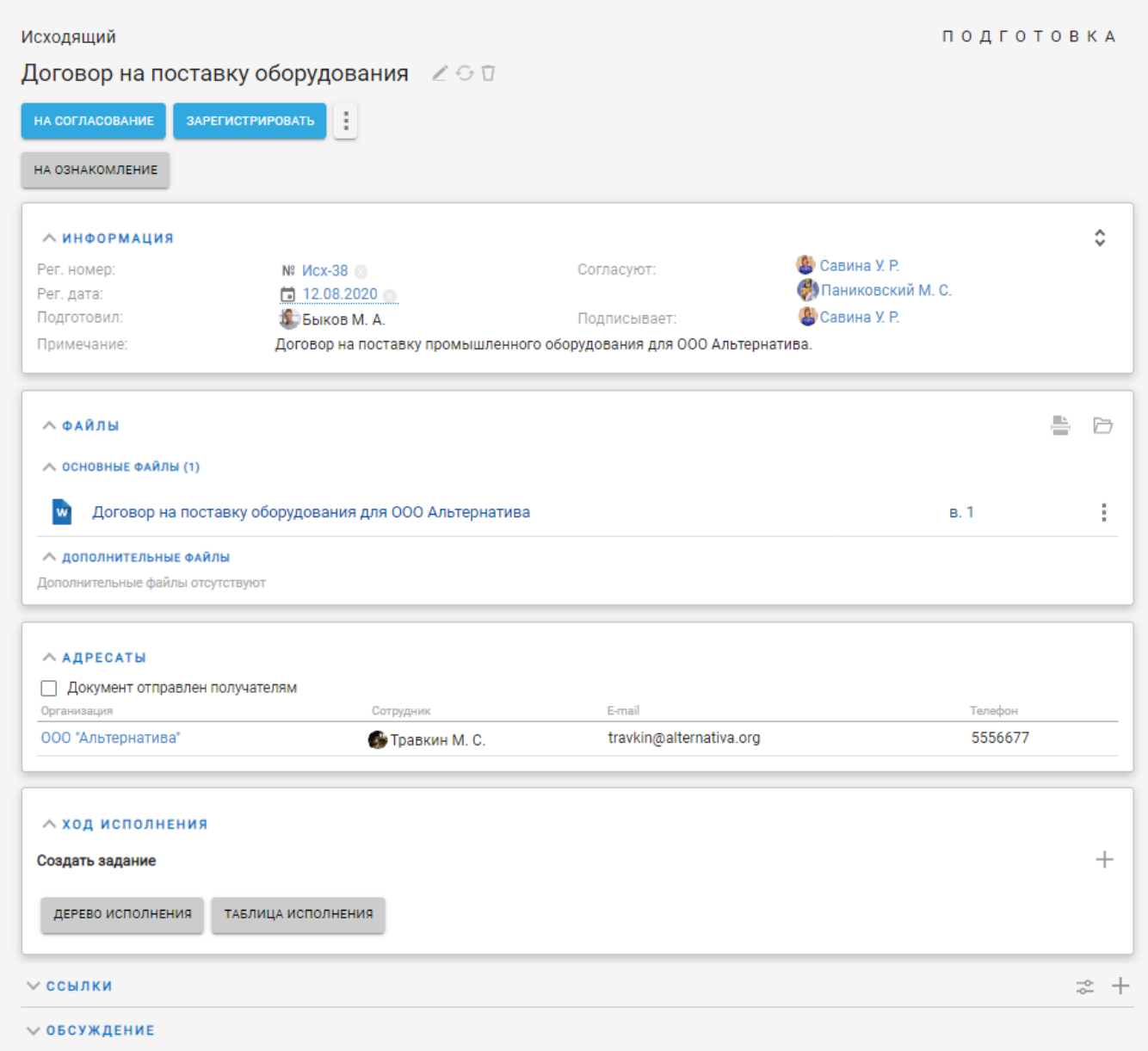


Рисунок 153. Пример исходящего документа

ОРД

ОРД — аббревиатура **О**рганизационно-**р**аспорядительный **д**окумент. Предназначены для документирования распорядительной деятельности учреждений и организаций.

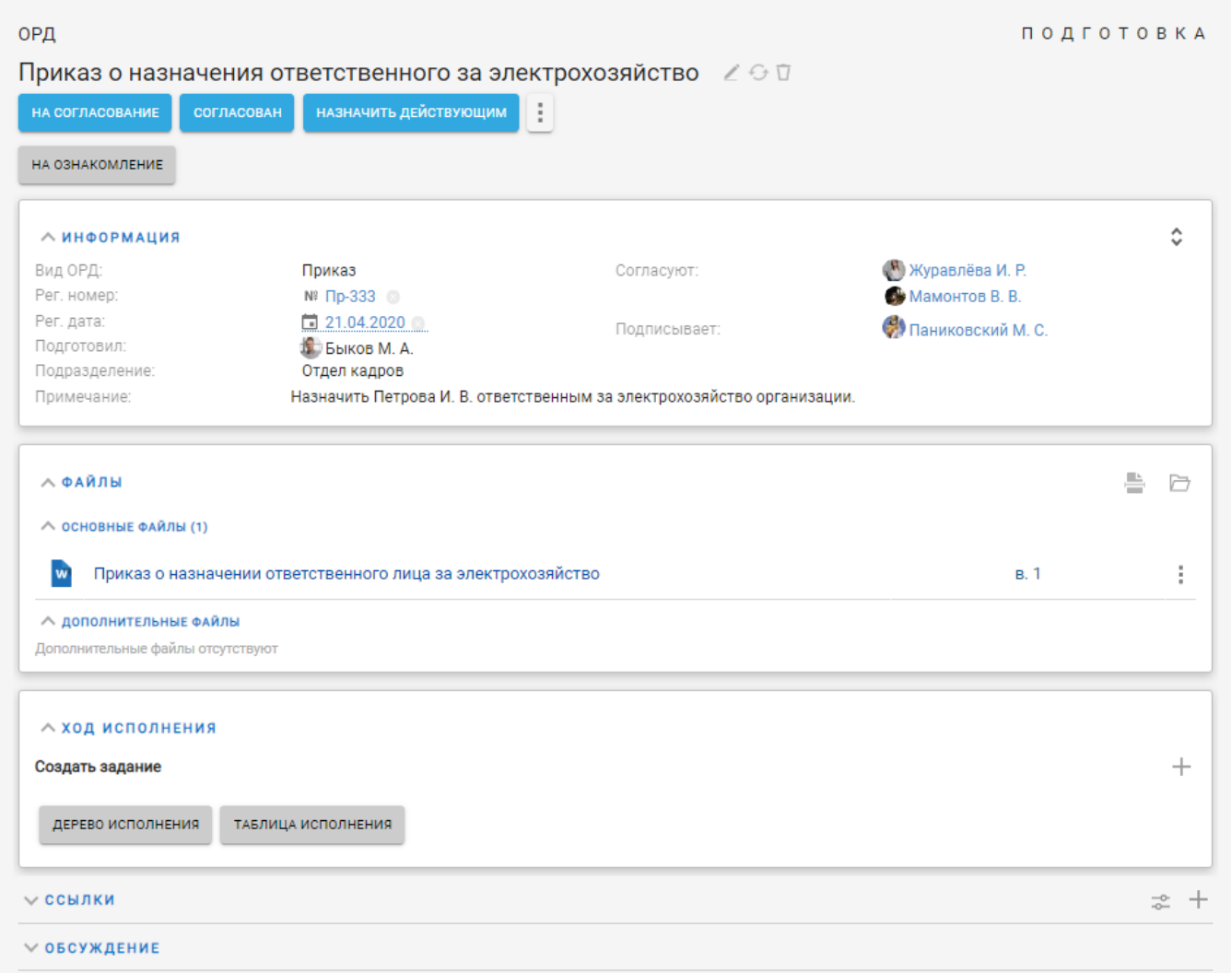


Рисунок 154. Пример организационно-распорядительного документа

Заявка

Предназначены для хранения в системе Docsvision файлов внутренней корреспонденции. Внутрення корреспонденция может быть использована для отправки различных заявок.

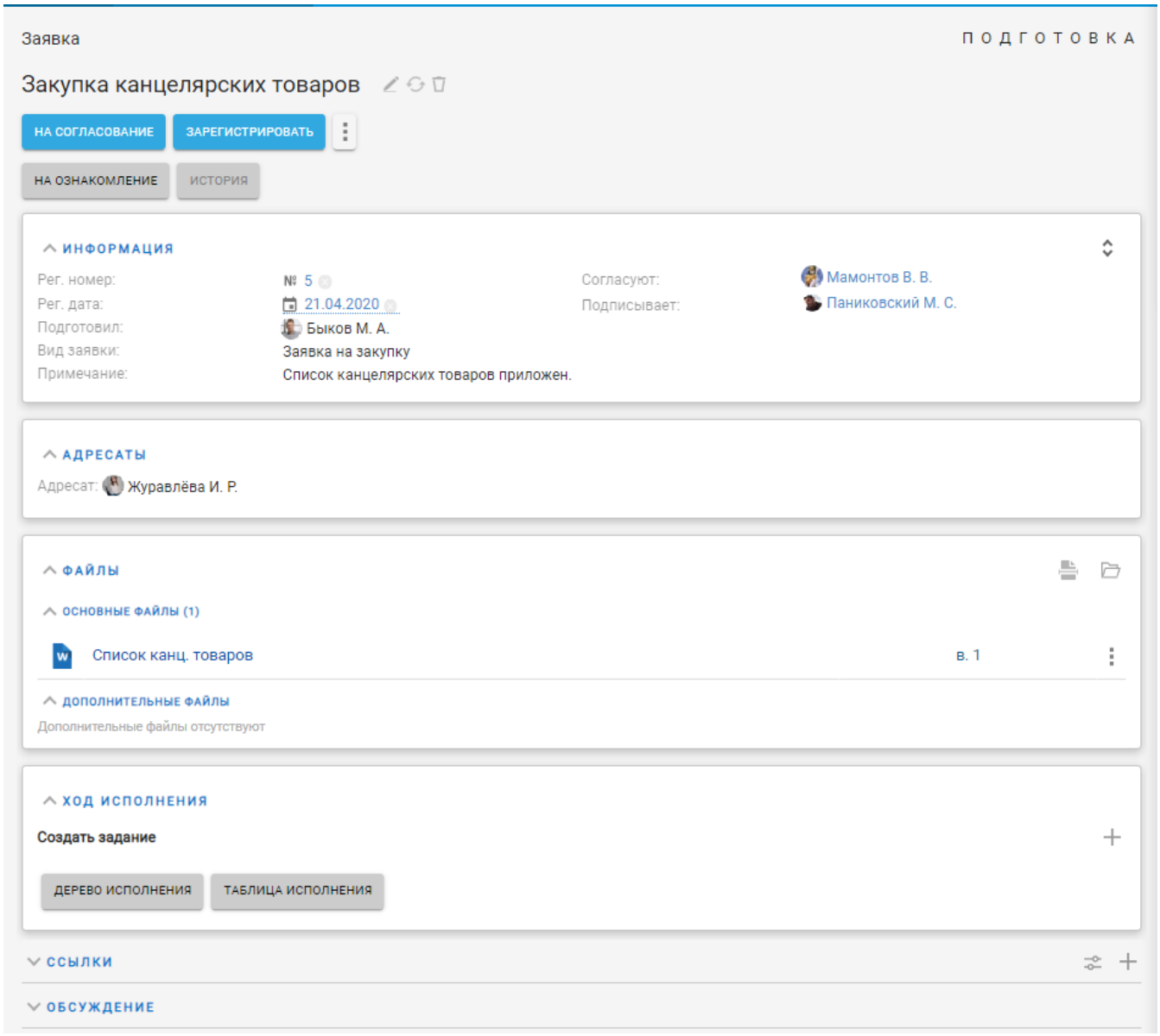


Рисунок 155. Пример заявки

Рабочий

Предназначен для выполнения произвольных задач документооборота. Использование документа предполагается внутри рабочих групп.

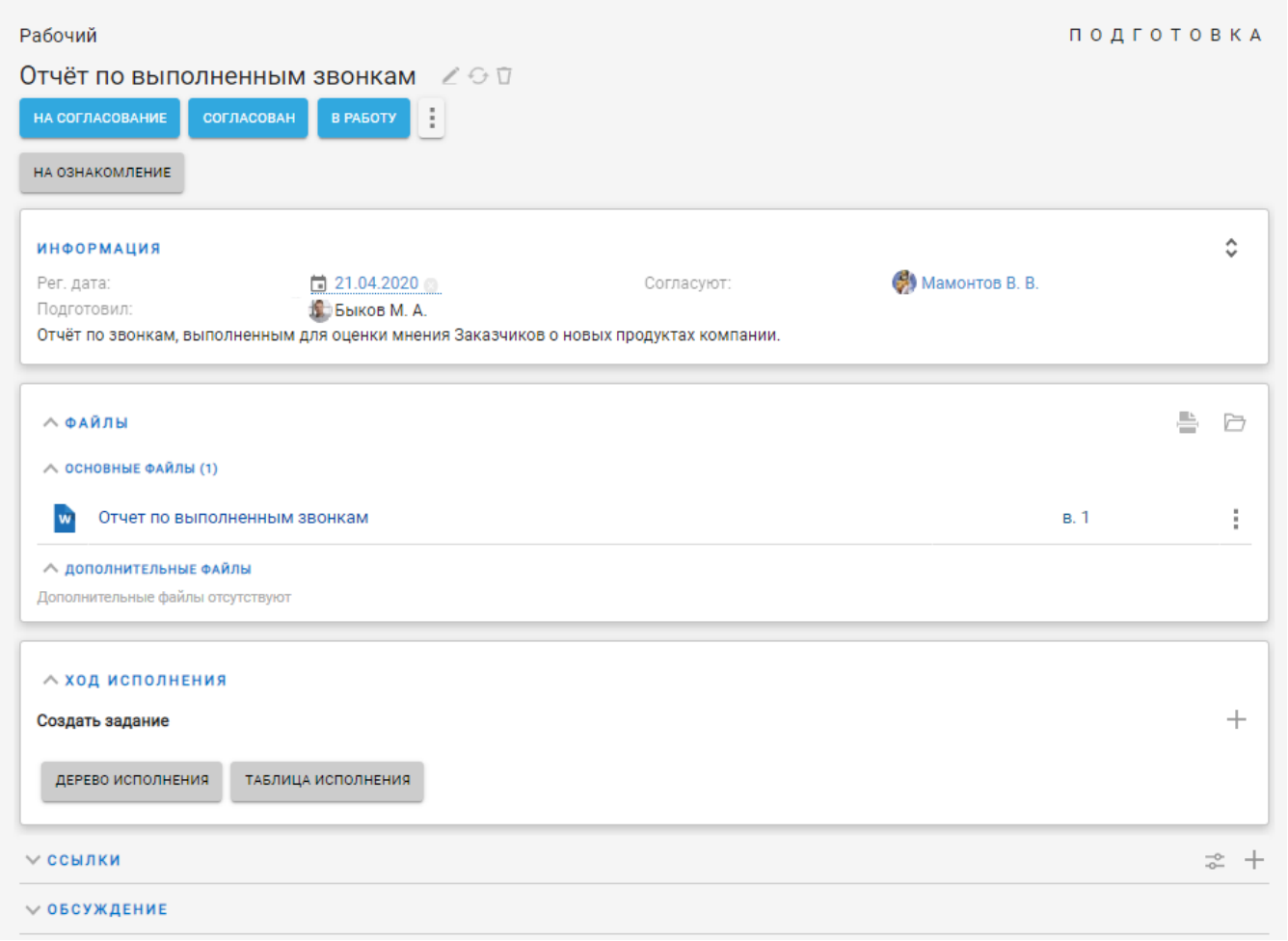


Рисунок 156. Пример рабочего документа

Служебная записка

Предназначен для ведения деловой переписки внутри предприятия. Деловая переписка может включать: создание докладных, объяснительных и прочих служебных записок.

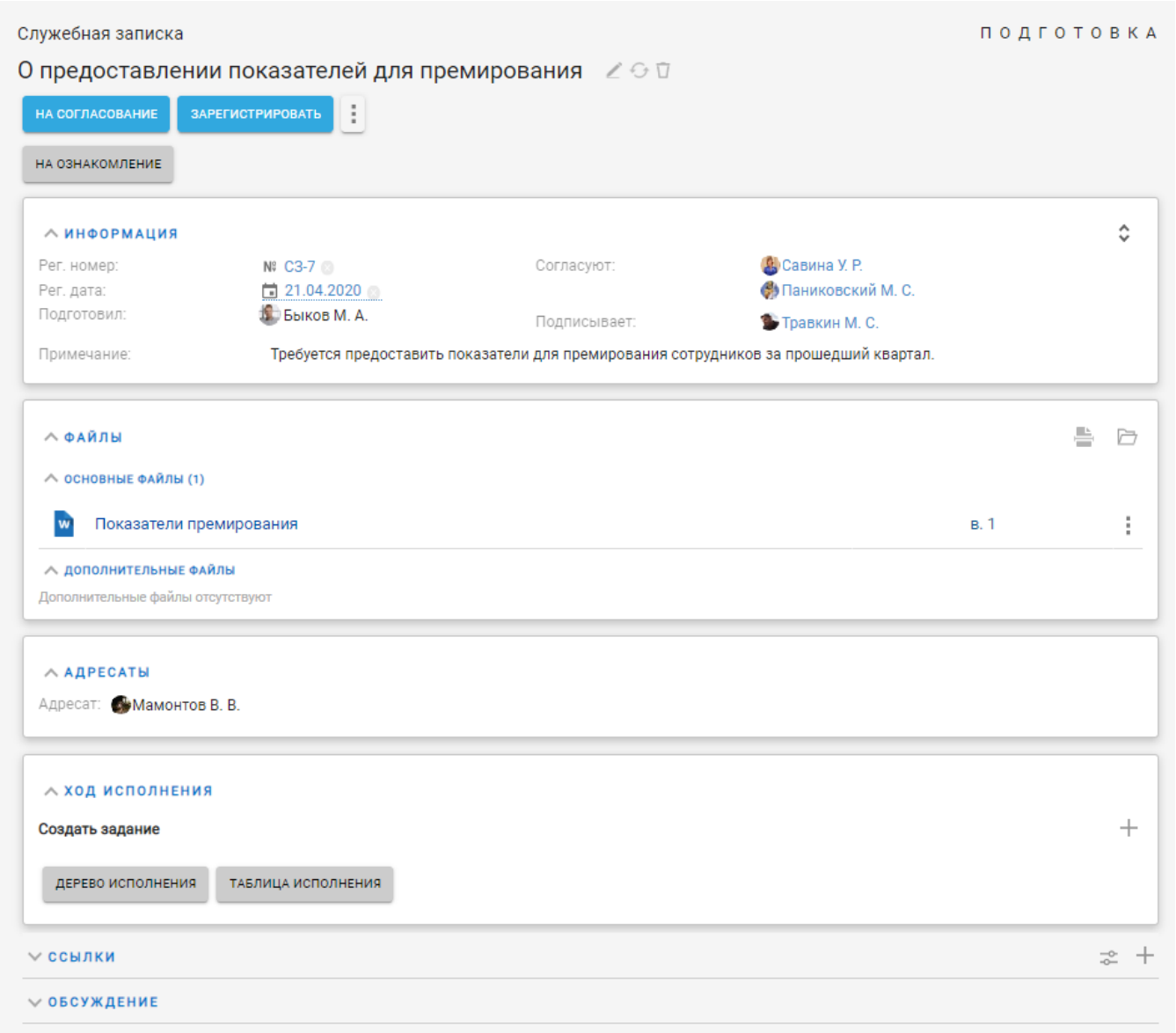


Рисунок 157. Пример служебной записки

На представленных выше изображениях скрыта область предварительного просмотра вложенных файлов.

Работа с документами в различных организациях может отличаться. Схематичное изображение типовых этапов работы с документами в Web-клиенте приведено в Приложении А. Основные сценарии работы с документами.

Создать документ

1. Создайте карточку одного из доступных видов документа.

   Будет открыто окно новой карточки.

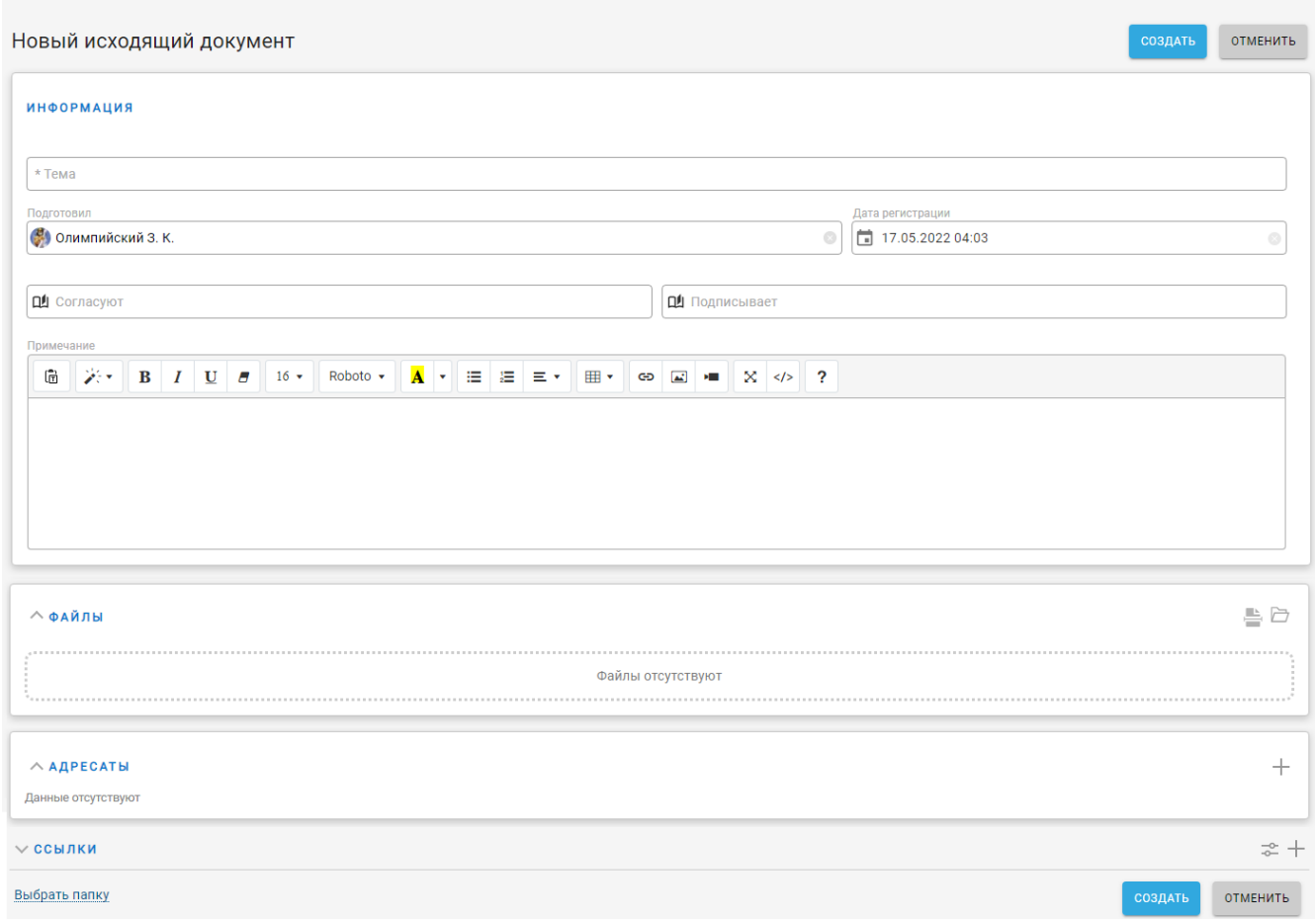


Рисунок 158. Карточка документа "Исходящий"

Если документ создан по шаблону, часть полей будет заполнена.

2. Заполните поля документа.

Перечень полей карточки зависит от вида документа. Перечень полей приведён в пунктах далее.

3. В блоке *Папка* выберите папку, в которую будет добавлен ярлык на карточку.

В зависимости от настроек Web-клиента в блоке *Папка* может быть по умолчанию выбрана текущая папка, а также ограничен список доступных для выбора папок.

Если папка не выбрана, карточка будет создана без размещения. Такие карточки можно найти с помощью поиска или виртуальной папки с соответствующим поисковым запросом, а также выбрать из списка последних карточек.

4. Нажмите кнопку **Создать**. Созданный документ будет открыт в режиме просмотра.

> Примеры карточек, открытых для просмотра, приведены в пункте Работа с документами.

Перечень полей исходящего документа

Тема

Название документа.

Обязательно для заполнения.

Подготовил

Сотрудник, подготовивший проект документа. По умолчанию выбирается текущий сотрудник.

Обязательно для заполнения.

Дата регистрации

Дата регистрации документа в организации. По умолчанию устанавливаются текущие дата и время.

Согласуют

Сотрудники, участвующие в согласовании документа.

Могут быть выбраны только сотрудники.

Подписывает

Сотрудники, участвующие в подписании документа при согласовании.

Могут быть выбраны только сотрудники.

Примечание

Дополнительная информация по документу.

Файлы

Файлы документа. Файлы можно добавить с текущего устройства или со сканера (см. условия в описании элемента Список файлов).

Организация

Контрагенты из *Справочника контрагентов*, которым должен быть отправлен документ после подготовки, согласования и регистрации.

Сотрудник

Сотрудник контрагента.

Перечень полей входящего документа

Тема

Название документа.

Обязательно для заполнения.

Регистрационный номер

Регистрационный номер документа. Может быть получен из списка свободных номеров. Ручной ввод номера не предусмотрен.

Дата регистрации

Дата регистрации документа в организации. По умолчанию устанавливаются текущие дата и время.

Регистратор

Сотрудник, который будет регистрировать документ. По умолчанию выбирается текущий сотрудник.

Обязательно для заполнения.

Описание

Дополнительная информация по документу.

Файлы

Файлы документа. Файлы можно добавить с текущего устройства или со сканера (см. условия в описании элемента Список файлов).

Получатели

Сотрудники организации, которые должны быть ознакомлены с документом.

Отправитель-контрагент

Информация об организации-отправителе документа.

Организация-отправитель

Контрагент, отправивший документ.

Сотрудник-отправитель

Ответственный сотрудник контрагента.

Номер отправителя

Номер документа, который был присвоен в организации-отправителе.

Вводится вручную.

Дата отправки

Дата регистрации документа в организации-отправителе.

По умолчанию устанавливаются текущие дата и время.

Перечень полей документа ОРД

Тема

Название документа.

Обязательно для заполнения.

Дата регистрации

Дата регистрации документа в организации. По умолчанию устанавливаются текущие дата и время.

Подготовил

Сотрудник, который подготовил проект документа. По умолчанию выбирается текущий сотрудник.

Обязательно для заполнения.

Подразделение

Организация или подразделение организации, для которого предназначен документ.

Вид ОРД

Подвид организационно-распорядительного документа. По умолчанию выбирает *Приказ* (первый элемент из списка видов ОРД).

Описание

Дополнительная информация по документу.

Файлы

Файлы документа. Файлы можно добавить с текущего устройства или со сканера (см. условия в описании элемента Список файлов).

Перечень полей заявки

Тема

Название документа.

Обязательно для заполнения.

Дата создания

Дата создания документа. По умолчанию устанавливаются текущие дата и время.

Инициатор

Сотрудник, который подготовил проект документа. По умолчанию выбирается текущий сотрудник.

Обязательно для заполнения.

Вид заявки

Подвид заявки.

Описание

Дополнительная информация по документу.

Файлы

Файлы документа. Файлы можно добавить с текущего устройства или со сканера (см. условия в описании элемента Список файлов).

Адресат

Сотрудники организации, которым адресован данный документ.

Перечень полей рабочего документа

Тема

Название документа.

Обязательно для заполнения.

Подготовил

Сотрудник, который подготовил проект документа. По умолчанию выбирается текущий сотрудник.

Обязательно для заполнения.

Дата создания

Дата создания документа. По умолчанию устанавливаются текущие дата и время.

Описание

Дополнительная информация по документу.

Файлы

Файлы документа. Файлы можно добавить с текущего устройства или со сканера (см. условия в описании элемента Список файлов).

Перечень полей служебной записки

Тема

Название документа.

Обязательно для заполнения.

Дата создания

Дата создания документа. По умолчанию устанавливаются текущие дата и время.

Инициатор

Сотрудник, который подготовил проект документа. По умолчанию выбирается текущий сотрудник.

Обязательно для заполнения.

Описание

Дополнительная информация по документу.

Файлы

Файлы документа. Файлы можно добавить с текущего устройства или со сканера (см. условия в описании элемента Список файлов).

Адресат

Сотрудники организации, которым адресован данный документ.

Зарегистрировать документ

Документы следующих видов могут быть зарегистрированы: *Входящий*, *Исходящий*, *Служебная записка* и *Заявка*. Возможность регистрации собственных видов документов определяется настройками.

Регистрируемый документ должен быть в состоянии:

- *Подготовка*. Для вида документа *Входящий*.
- *Подготовка*, *Согласован* или *Подписан*. Для видов документа *Исходящий*, *Служебная записка* и *Заявка*. После регистрации согласование данных видов документа станет невозможным.

В регистрируемом документе должны быть заполнены поля с регистрационным номером и датой регистрации.

Чтобы зарегистрировать документ:

1. Откройте карточку документа в режиме чтения.
2. На панели операций нажмите кнопку **Зарегистрировать**. Состояние документа изменится на Зарегистрирован.

Регистратору данный документ будет доступен в папке **Мои документы › Я — автор**.

Отправить документ на ознакомление

Чтобы отправить документ на согласование:

1. Откройте карточку документа в режиме чтения.
2. На панели операций нажмите кнопку **Отправить на ознакомление**. Будет открыта панель отправки на ознакомление.

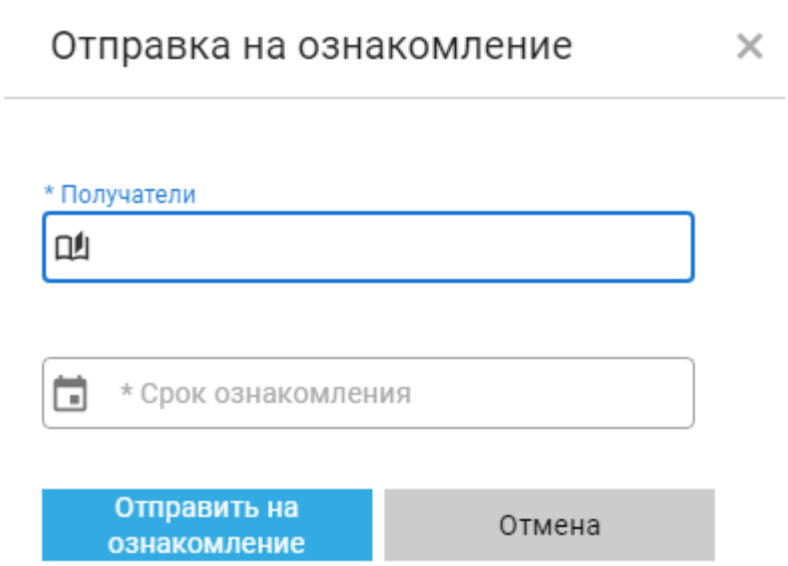


Рисунок 159. Панель отправки документа на ознакомление

3. В поле Получатели выберите организации, подразделения, сотрудников, группы, роли, которым данный документ будет отправлен на ознакомление.
4. Укажите срок ознакомления.
5. Нажмите кнопку **Отправить на ознакомление**.

   Выбранным организациям, подразделениям, сотрудникам, группам сотрудников будут разосланы задания *На ознакомление*.

Создать исходящий документ в ответ на входящий

Чтобы создать исходящий документ в ответ на входящий, выполните следующие действия:

1. Откройте карточку вида *Входящий* в режиме чтения.
2. На панели операций нажмите кнопку **Создать исходящий**. Будет открыта новая карточка исходящего документа.

   Часть данных карточки будет заполнена данными из входящего документа.

3. Заполните поля карточки и сохраните изменения.

   Созданная карточка будет добавлена в список ссылок в разделе *Ссылки*.

Создать задания и группу заданий по документу

Задание или группа заданий могут быть созданы при выполнении следующих условий:

- Карточка документа не должна быть заблокирована.
- Пользователь должен иметь достаточно прав на изменение данного документа.
- Пользователь должен иметь достаточно прав на создание задания в папке, в которой хранится данная карточка *Документ*.

Чтобы создать задание или группу заданий по документу:

1. Откройте карточку документа в режиме чтения.
2. В разделе *Ход исполнения* нажмите кнопку + и выберите вид создаваемого задания или группы заданий.

   Будет открыта карточка задания или группы заданий.

3. Заполните поля и отправьте на исполнение карточку Задания или Группы заданий.

   В созданную карточку будет добавлена ссылка на данный документ.

   Информация по созданному заданию или группе заданий будет отображаться в разделе *Ход исполнения* карточки Документ.

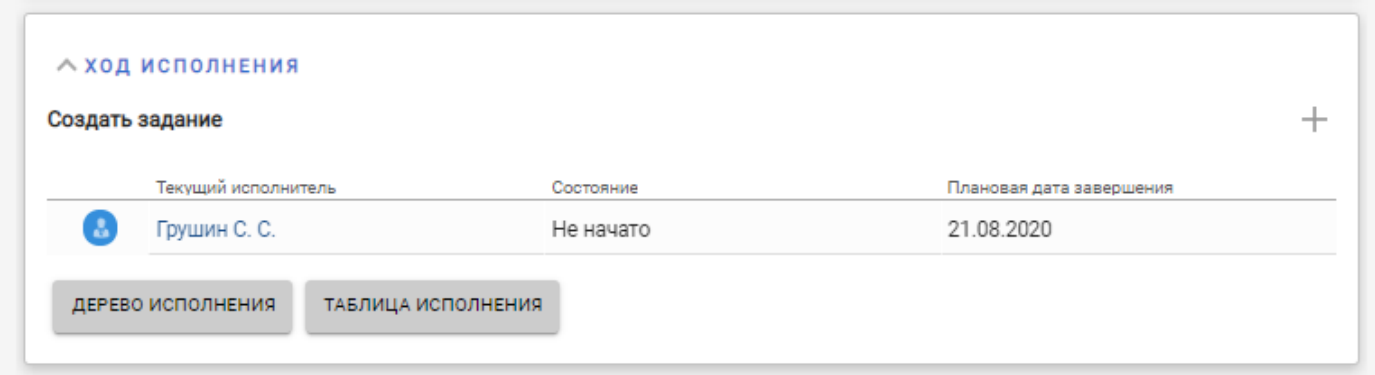


Рисунок 160. Информация по созданному заданию в документе

Чтобы перейти к заданию нажмите на имя исполнителя.

Чтобы перейти к заданиям исполнителей в группе заданий: нажмите на

значок ▾ слева от значка группы. Будет раскрыта группа заданий. Затем нажмите на имя исполнителя.



Значение в столбце *Текущий исполнитель* будет пустым, если задание не было отправлено исполнителю. Текущий исполнитель также не отображается для Группы заданий.

Ознакомиться с ходом исполнения связанных заданий и групп заданий

Ознакомиться с ходом исполнения задания или группы заданий можно с помощью раздела *Ход исполнения*.

1. Откройте карточку документа в режиме чтения.
2. Перейдите к разделу *Ход исполнения*.

   Для раскрытия групп заданий щелкните по значку ▾ слева от значка группы.

   Чтобы ознакомиться с ходом исполнения в виде дерева исполнения, нажмите кнопку **Дерево исполнения**. Для просмотра хода исполнения в виде таблицы нажмите кнопку **Таблица исполнения**.

Подписание карточек типа "Документ"

Пользователь может подписать карточки Документа электронной подписью[2] (ЭП) как в веб-версии, так и в мобильной версии сайта.

Существует два варианта работы с подписанием:

- С помощью ЭУ "Список файлов" — подписи отображаются в списке файлов.
- С помощью ЭУ "Кнопка электронной подписи" — подписи отображаются для карточки и не зависят от наличия или отсутствия файлов в карточке.

Диалог подписания в обоих случаях будет выглядеть одинаково.

Оба варианта позволяют подписать основные и дополнительные файлы, атрибуты документа в любых сочетаниях.

Подробнее о подписании см. далее на текущей странице.

Подписание выполняется с целью защиты данных карточки от изменений, после изменения данных ЭП станет недействительной.

Документ может быть подписан простой, усиленной или усовершенствованной

ЭП. Возможность подписания тем или иным видом ЭП регулируется настройками системы Docsvision, а также наличием необходимых программных средств на компьютере пользователя.

Ограничения подписания

Размер файлов для подписания в Web-клиенте неограничен со стороны Docsvision, существуют ограничения со стороны КриптоПро, подробнее см. в документации КриПтоПро.

Работа с ЭП на iOS и Android через интерфейс Web-клиента не поддерживается без специального мобильного приложения.

Подписание документа

Чтобы подписать документ:

1. Откройте карточку документа в режиме чтения.
2. Найдите кнопку (по умолчанию располагается в области заголовка карточки или списке файлов).

   В зависимости от настроек в Конструкторе состояний подписание или просмотр журнала подписей могут быть недоступны. Также в зависимости от настроек кнопка может не отображаться.

3. Нажмите на кнопку и в раскрывающемся меню выберите *Подписать*.

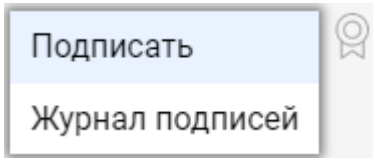


Рисунок 161. Меню кнопки подписания

На экране появится окно *Электронная подпись*.

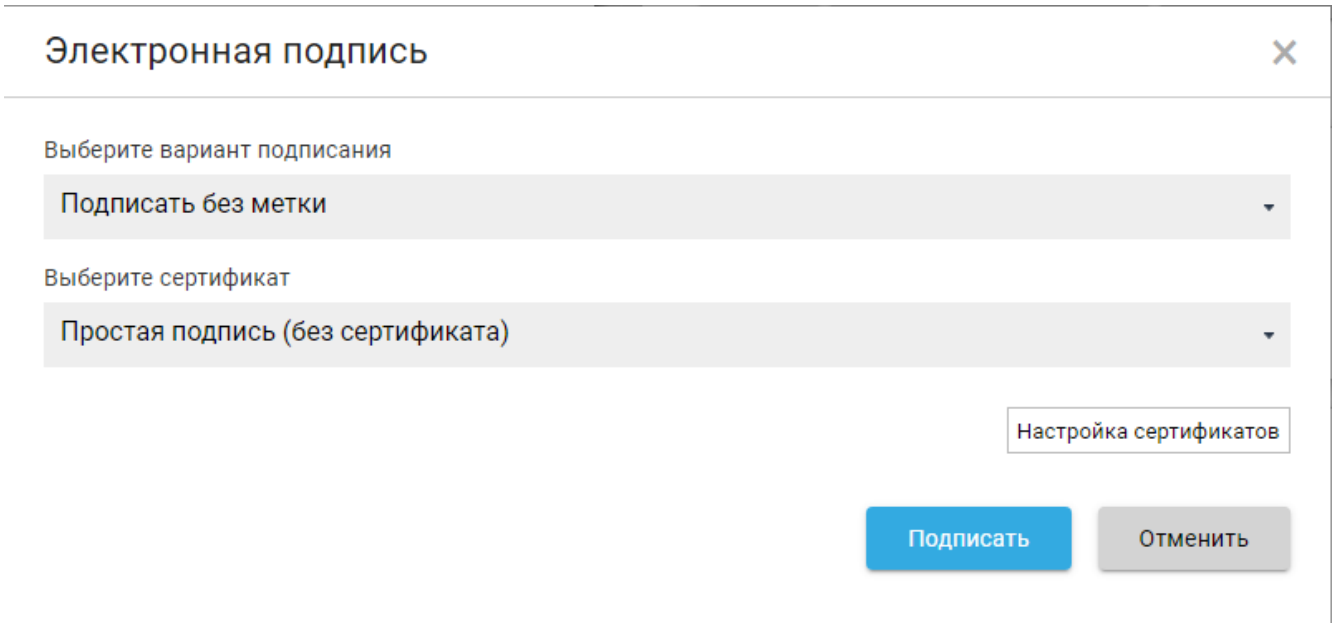


Рисунок 162. Параметры электронной подписи

4. Выберите метку для подписи.

> Метки конкретизируют решение сотрудника, подписавшего карточку документа.
>
> Если метки отсутствуют (в *Справочнике меток подписей*), выбрать метку нельзя.

5. Выберите вариант подписания из списка:
   - ***Простая подпись (без сертификата)*** — документ будет подписан учётной записью сотрудника.

     Функция может быть недоступна, если простая подпись запрещена настройками карточки.

   - Выберите любой сертификат из списка, чтобы подписать документ усиленной локальной или облачной подписью.

     Чтобы иметь возможность подписать документ локальной усиленной подписью, необходимо установить компоненты КриптоПро. Чтобы подписать документ облачной подписью, необходима регистрация в сервисе ЭП КриптоПро.

Варианты подписания

Выбор варианта подписания зависит от настроек *Справочника сотрудников* и наличия действительных сертификатов для пользователя:

Когда компоненты для работы с усиленной ЭП не установлены:

- В окне будет отображаться подсказка: `Не установлен компонент для использования усиленной подписи` и в списке будет доступен только вариант ***Простая подпись***.

Когда компоненты для работы с усиленной ЭП установлены:

- Если у пользователя есть только один действующий сертификат, в списке вариантов подписания будут присутствовать этот сертификат и ***Простая подпись***.
- Если локальный сертификат в персональном хранилище сотрудника не совпадает с указанным в справочнике сотрудников, в дополнение к сертификату и ***Простой подписи*** будет отображаться предупреждающее сообщение: `Сертификат в персональном хранилище не совпадает с указанным в Справочнике сотрудников`.
- Если у пользователя есть несколько сертификатов, в списке будут присутствовать варианты:
  - ***Простая подпись***.
  - Сертификаты, установленные в персональном хранилище сотрудника.

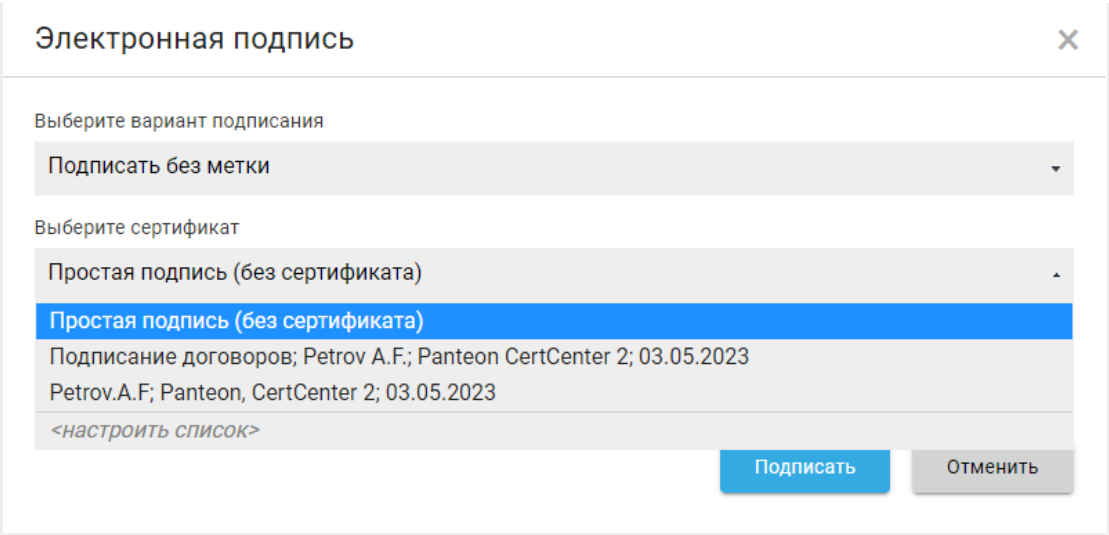


Рисунок 163. Выбор сертификата для подписания из списка

- Сертификаты, установленные в облачном хранилище.

Пункт ***<настроить список>*** открывает окно *Настройка сертификатов*, подробнее см. Подписание облачной подписью.

Если у пользователя отсутствуют сертификаты, в списке будет отображаться сертификат по умолчанию из справочника сотрудников и ***Простая подпись*** (если она разрешена настройками карточки).

6. При необходимости выполните **Настройки способов подписи**, подробнее см. Подписание облачной подписью.
7. Нажмите на кнопку **Подписать**.

Если ЭП не может быть сформирована, появится сообщение `Электронная подпись не может быть сформирована. Не указаны атрибуты для включения в подпись`.

Сообщение выводится в случае, если:

- В карточке отсутствуют основные и дополнительные файлы. Или для вида карточек дополнительные файлы не включаются в подпись.
- В настройках вида карточки не указаны атрибуты, включаемые в подпись.

Может быть запрошено подтверждение подписи. Если для пользователя

настроено несколько способов подтверждения подписи, запрошен будет только один, любой из заданных.

Если для сертификата требуется ввести пин-код, введите его во всплывающем окне.

Если выполняется подписание с использованием облачного сертификата может потребоваться авторизация в сервисе электронной подписи КриптоПро.

Если в сервисе электронной подписи настроена двухфакторная аутентификация, будет предложено ввести код подтверждения из СМС, электронной почты или подтвердить вход в приложении.
В приложении при подтверждении отображаются файлы для подписи, если таковые есть.

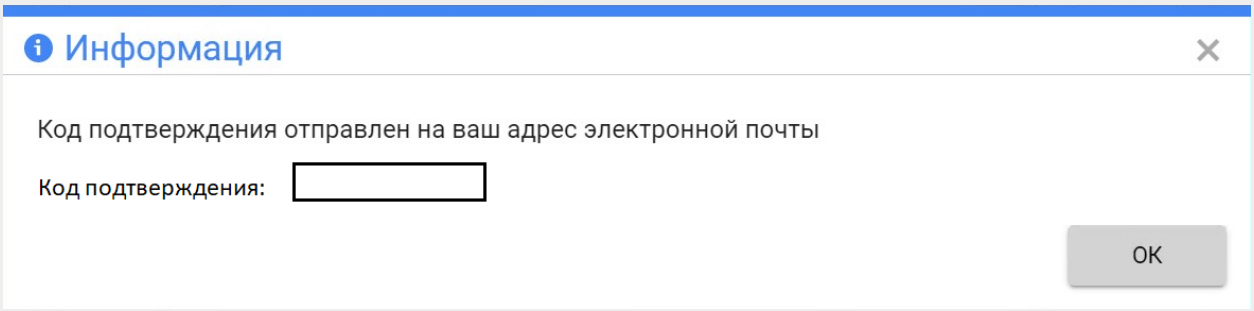


Рисунок 164. Двухфакторная аутентификация

Если подписание завершилось успешно, появится сообщение Электронная подпись сформирована успешно. Цвет значка сменится на серый.

Ознакомиться с информацией о подписях документа можно в журнале подписей.

Подписание с помощью мобильного устройства

Docsvision Web-клиент поддерживает возможность подписания облачной подписью на мобильных устройствах. использование локальной подписи на мобильных устойствах не поддерживается.

Чтобы подписать документ с мобильного устройства:

1. Убедитесь, что для сотрудников настроено использование облачных сертификатов, см. подробнее "Подписание облачной подписью".
2. Установите приложение КриптоПро myDSS 2.0 или DSS Client для работы с

подписью на мобильных устройствах. Приложения обладают схожей функциональностью, рекомендуется использовать DSS Client как официальное приложение КриптоПро.

3. Убедитесь в наличии выпущенного облачного сертификата с местом хранения на устройстве, а не в облаке.
4. Установите двухфакторное подтверждение для сертификата в мобильном приложении.
5. Откройте с мобильного устройства карточку документа на подписание и выберите нужный сертификат.
   - Сертификат будет доступен по умолчанию, если он указан в справочнике сотрудников.
   - В остальных случаях сертификат будет доступен в форме настройки списка сертификатов.
6. Подпишите документ выбранным сертификатом.

Подписание с помощью МЧД

При подписании документа можно использовать МЧД. В зависимости от настроек, выполненных администратором, МЧД может быть желательна, обязательна или не требоваться.

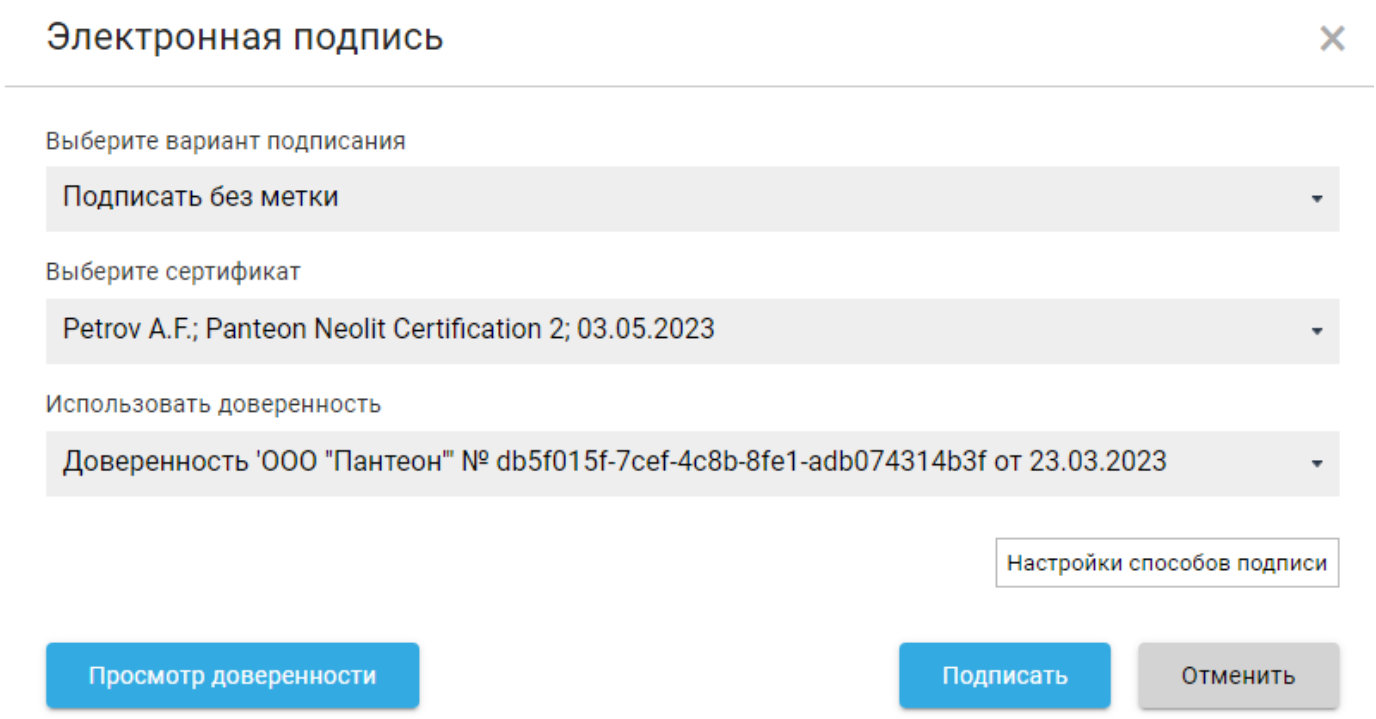


Рисунок 165. Подписание с использованием доверенности

МЧД или машиночитаемая доверенность содержит данные:

- О доверителе — организации, выдавшей доверенность.
- О подписанте — физическом лице, уполномоченным на подписание.
- О представителе — физическом или юридическом лице, которое уполномочено совершать действие.
- О полномочиях — правах, которые предоставлены подписанту.
- О дате совершения доверенности.
- О сроке действия доверенности.

При формировании ЭП в стандартных диалогах подписания Web-клиента может быть использована МЧД.

Подпишите документ стандартным образом:

1. Нажмите на кнопку и в раскрывающемся меню выберите *Подписать*.

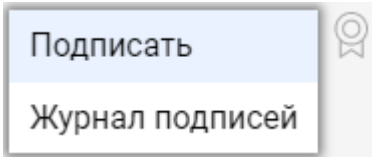


Рисунок 166. Меню кнопки подписания

2. В окне *Электронная подпись* выберите сертификат ЭП и доверенность из раскрывающегося списка.

> Список доступных доверенностей не отображается, если не выбран сертификат.

Доступные для пользователя доверенности выбираются по специальному алгоритму, см. подробнее руководство по настройке системы.

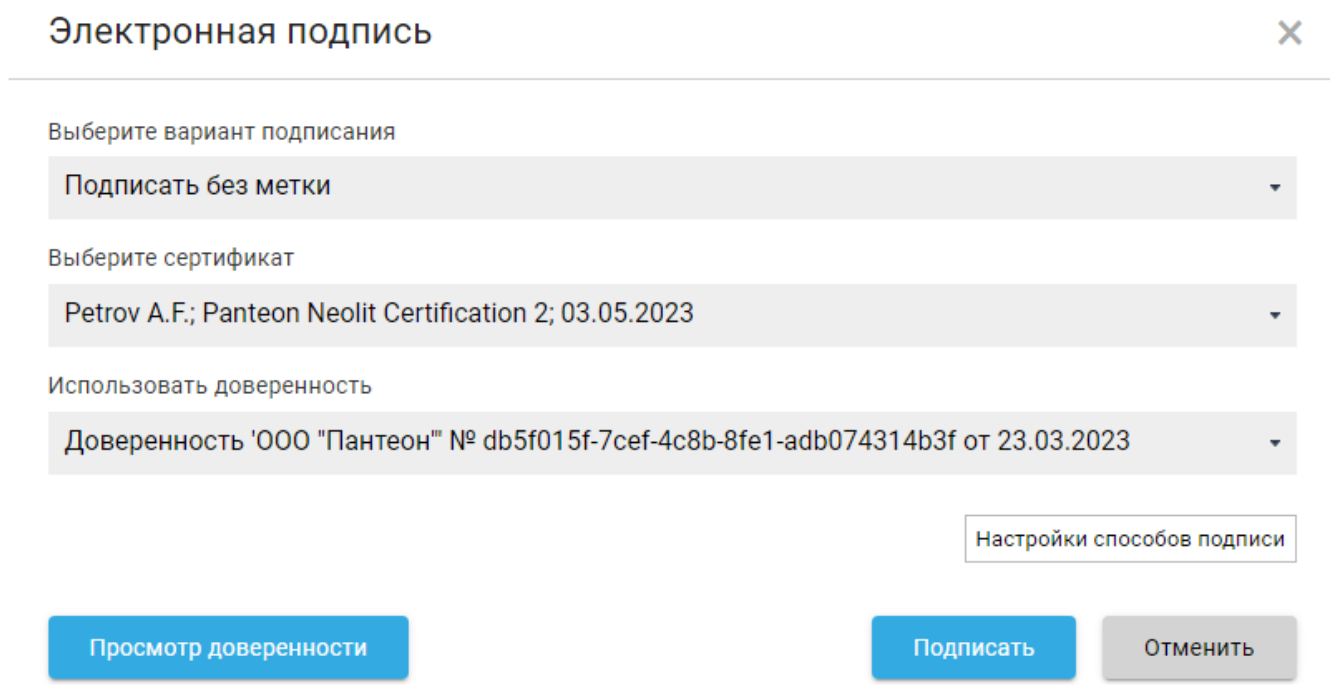


Рисунок 167. Использовать доверенность при подписании

3. Выбранную доверенность можно просмотреть, нажав на кнопку **Просмотр доверенности**.

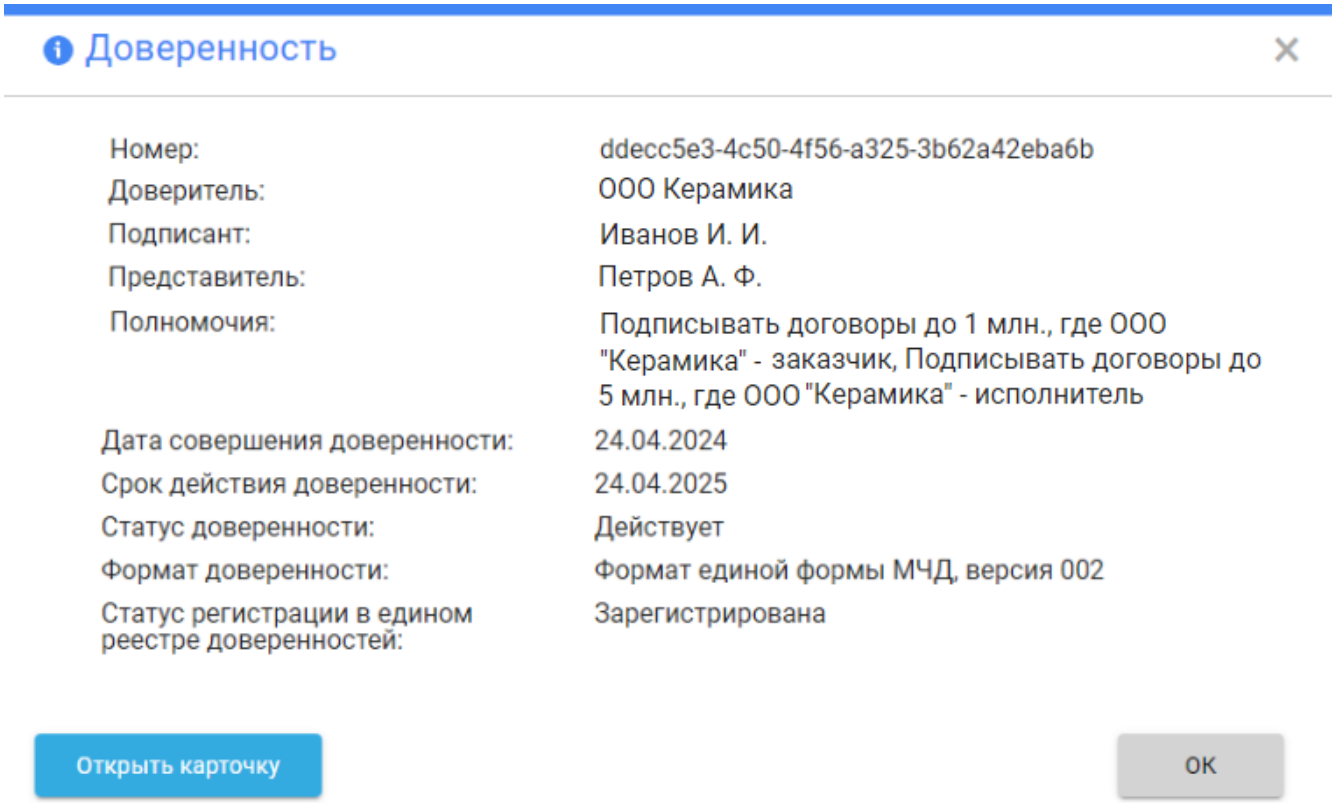


Рисунок 168. Просмотр информации об МЧД

Кнопка будет заблокирована, если доверенность не выбрана. При

использовании простой подписи кнопка и список доступных МЧД не отображаются.

4. Если подходящие МЧД не были найдены, будет выдано соответствующее сообщение.

> При подписании с использованием МЧД, информация о доверенности сохраняется в Журнал ЭП. Информация о доверенности также сохраняется при подписании из задания согласования.
>
> В момент подписания с использованием МЧД проверяется её действительность и факт отозванности.
>
> В документе, подписанном с использованием МЧД, так же как при подписании с использованием ЭП будет отображаться штамп подписи.

Для работы с кодами полномочий предусмотрен "Справочник кодов полномочий".

Журнал электронных подписей

1. Откройте карточку документа в режиме чтения.
2. Найдите кнопку  (по умолчанию располагается в области заголовка карточки или списке файлов).
3. В раскрывающемся списке кнопки выберите *Журнал подписей*.

> Будет открыто окно со списком подписей и результатами их проверки.

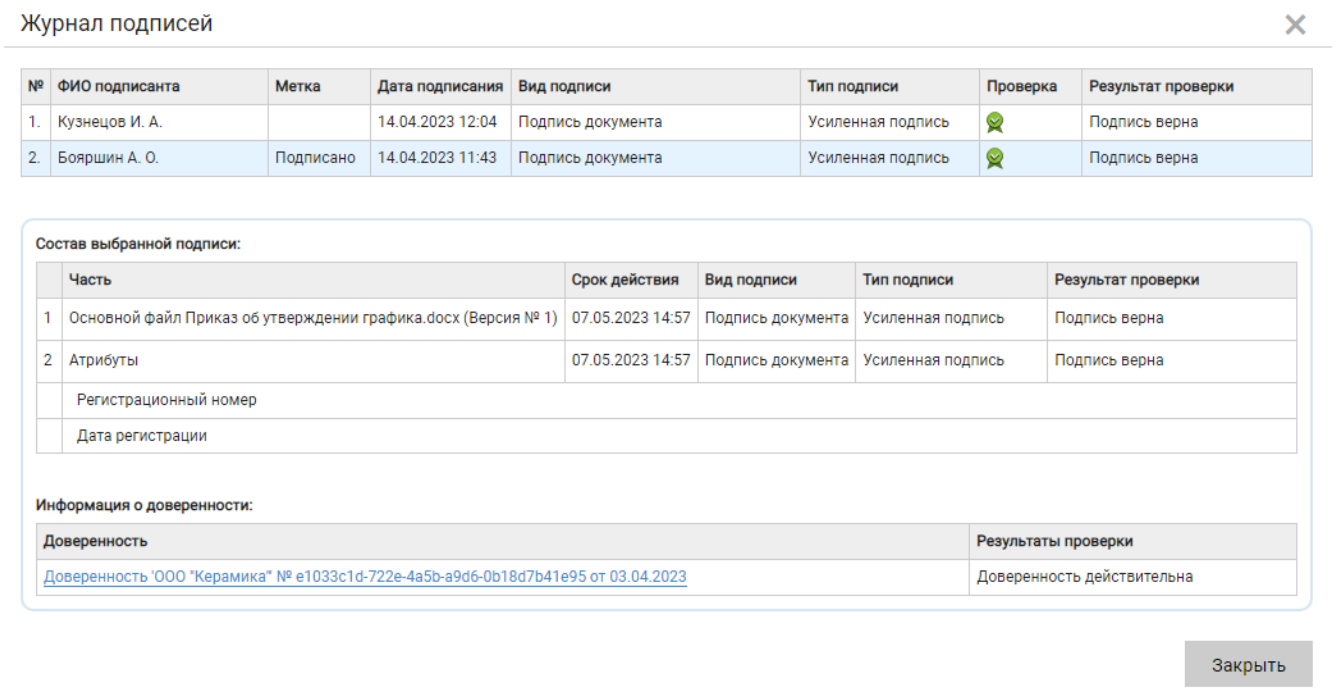


Рисунок 169. Список наложенных подписей и информация о доверенности

Окно "Журнал подписей" содержит два раздела:

Список подписей

Перечень электронных подписей в данном документе.

Список включает следующую информацию:

- ФИО подписанта.

  Если подписант присутствует в справочнике, данные берутся из *Справочника сотрудников*. Если подписант отсутствует в справочнике, данные берутся из подписи.

- Метку подписи.
- Дату подписания.
- Вид подписи. Подписание документа или импортированная подпись.
- Тип подписи. Простая или усиленная.
- Значок и результат проверки подписи.

Подпись может быть верна и не верна, что отражается внешним видом значка валидации:

- — подпись неверна, если хотя бы одна часть подписи неверна.
- — подпись верна и содержит подписи всех основных файлов.
- — подпись верна, но содержит подписи не всех основных файлов.

- — подпись верна, но срок действия сертификата, которым выполнено подписание, истек.

  Данный вариант неприменим к подписи, в которой установлена метка времени (TSP/OCSP) — проверка осуществляется по метке времени.

Дополнительная информация об импортированной подписи

Если документ был подписан с использованием импортированной из файла подписи, отображается *Дополнительная информация об импортированной подписи*.
В поле *Вид подписи* у такой подписи будет указано значение *Импортированная подпись*.

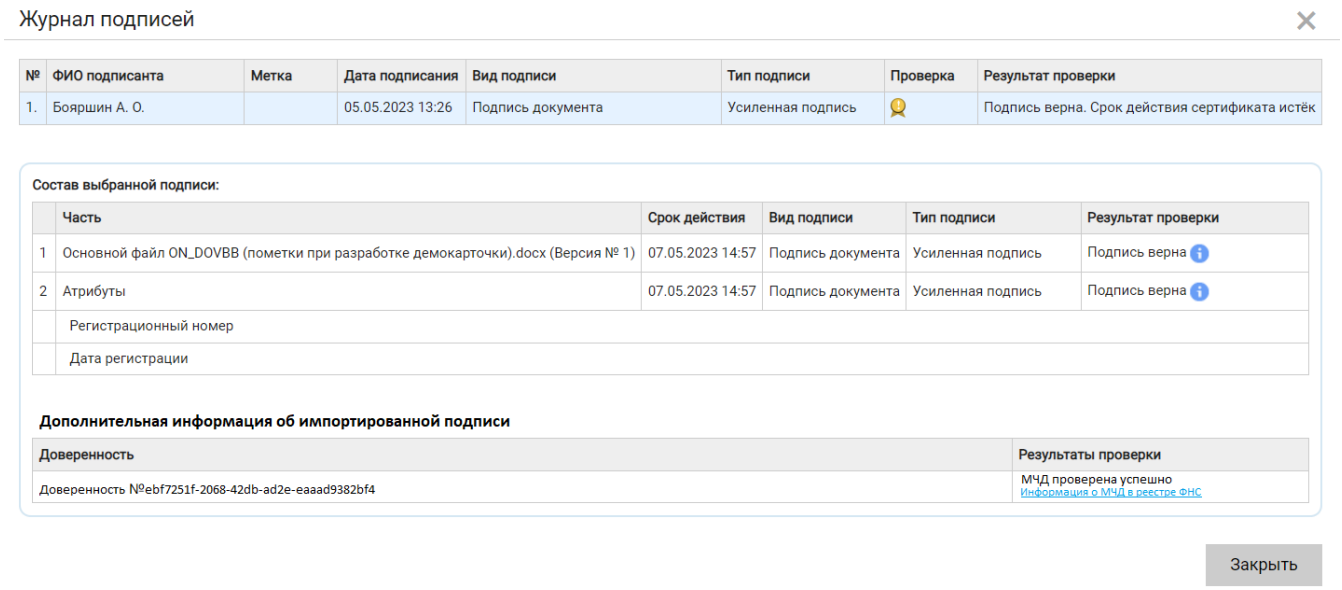


Рисунок 170. Журнал электронных подписей с информацией об импортированной подписи

Если ЭП была загружена в карточку, содержащую больше одного подписанного основного файла, результат проверки загруженной подписи будет смешанным: Не все основные файлы подписаны. При этом в разделе *Состав подписи* положительный результат проверки будет только у файла, для которого была загружена отсоединённая подпись. Такой же результат проверки может возникать, если все файлы подписаны, но имеют разные подписи.

В колонке *Доверенность* таблицы *Дополнительная информация об импортированной подписи* отображается номер сторонней доверенности из журнала подписей. В колонке *Результат проверки* отображается статус МЧД и ссылка на проверку МЧД в реестре ФНС (в случае, если МЧД передана в распределенный реестр ФНС).

Состав выбранной подписи

Информация о частях подписи, выбранной в списке подписей. Нажмите на строку подписи в списке, чтобы просмотреть её состав.

Подпись может включать следующие части:

- Подписанные основные файлы с указанием версии.
- Подписанные дополнительные файлы.
- Подписанные атрибуты документа, если их подписание настроено.

Для каждой части отображаются:

- Название подписанного элемента.
- Вид подписи.
- Тип подписи.
- Результат проверки данной части подписи.

Информация о доверенности

В таблице отображается информация о доверенности. Таблица имеет колонки: *Доверенность* и *Результат проверки*. *Информация о доверенности* отображается, только если при подписании использовалась МЧД. Если выбирается другая подпись, то информация о доверенности обновляется. Если при этом к другой подписи МЧД не была приложена, то *Информация о доверенности* будет скрыта.

Если в СКД (см. документацию разработчика, раздел "PowerOfAttorney — класс") доверенность хранится в XML-формате, при нажатии на ссылку в колонке *Доверенность* открывается всплывающее окно со следующими полями:

- Номер доверенности.
- Доверитель.
- Подписант.
- Представитель.
- Полномочия: поля *Код полномочий* и *Название полномочий* из Справочника полномочий или свободное текстовое описание.
- Дата совершения доверенности.
- Срок действия доверенности.

- Статус доверенности: подготовка, действует, отозвана (с указанием даты отзыва).
- Формат доверенности.
- Кнопка **Открыть карточку** — отображается, если в СКД есть ссылка на ПКД.

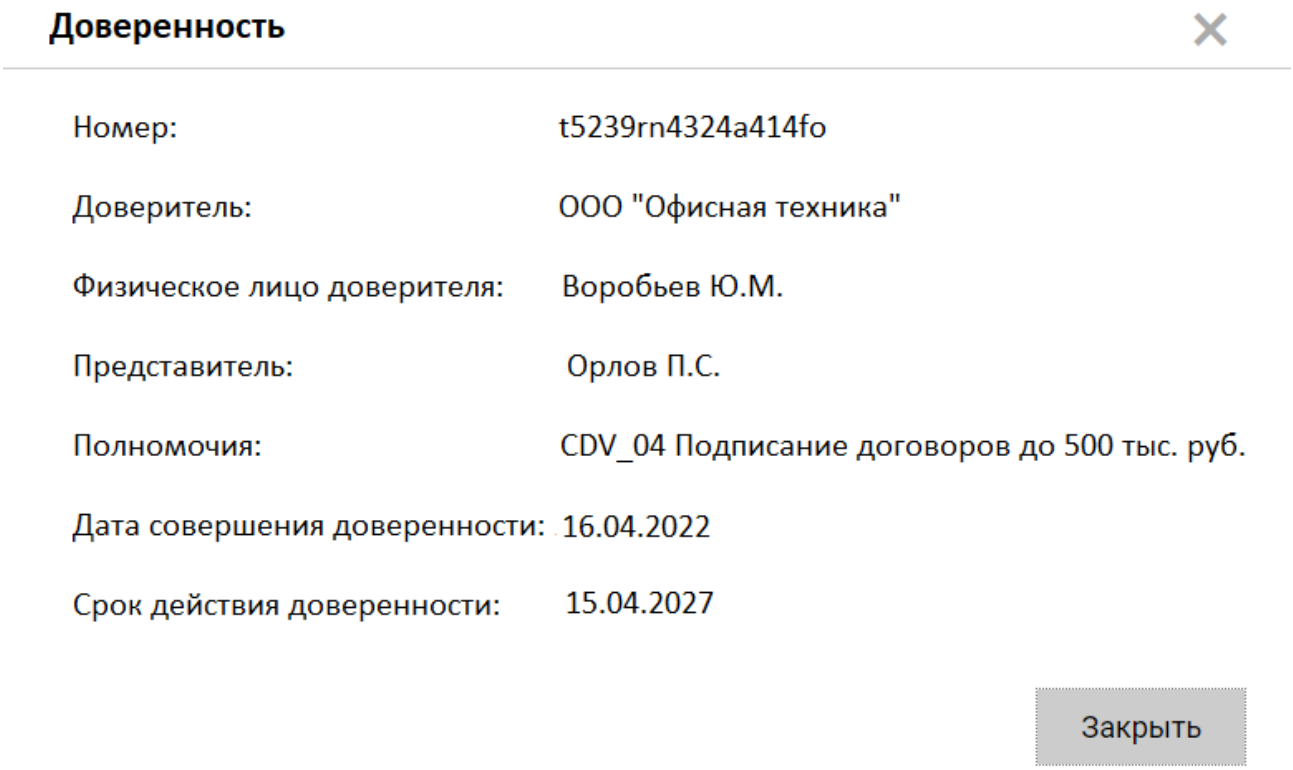


Рисунок 171. Просмотр доверенности

Результаты проверки подписи

Для подписи могут отображаться следующие виды результатов проверки:

Подпись верна

Все текущие версии основных файлов подписаны и подписи верны.

Подпись не верна

Все текущие версии основных файлов подписаны, но подписи не верны.

Не все основные файлы подписаны

Подписаны не все текущие версии основных файлов. Присутствуют как подписанные, так и не подписанные файлы. Такая ситуация возможна, когда после наложения подписи были добавлены новые основные файлы.

Если была наложена квалифицированная усиленная ЭП, в списке подписей

будет присутствовать информация о типе подписи.

Если в результате проверки выяснится, что усиленная подпись не верна, в журнале будет указана причина:

- `Срок действия сертификата истек`.
- `Не удалось проверить сертификат`.
- `Файлы были изменены`.

Все возможные варианты результата проверки подписи представлены в таблице.

Таблица 11. Результаты проверки подписи

Результаты проверки подписи	**Суммарный статус подписи**	**Статус валидных частей подписи**	**Статус НЕ валидных частей подписи**
Сертификат валиден либо используется простая подпись:			
Все файлы, атрибуты подписаны и подписи валидны	Подпись верна	Подпись верна	-
Файлы, атрибуты подписаны, но результат проверки отрицательный (данные изменены)	Подпись не верна	Подпись верна	Подпись не верна
Подписи валидны, но присутствуют части подписи не для всех файлов документа	Подписаны не все файлы	Подпись верна	Не подписан
Сертификат не валиден или не проверен:			
Не удалось проверить сертификат	Подпись не проверена (Не удалось проверить сертификат)	-	-

Результаты проверки подписи	Суммарный статус подписи	Статус валидных частей подписи	Статус НЕ валидных частей подписи
Сертификат отозван	Подпись не верна (Сертификат отозван)	-	-
Срок действия сертификата истек	Подпись не верна (Срок действия сертификата истек)	-	-

Результат проверки доверенности

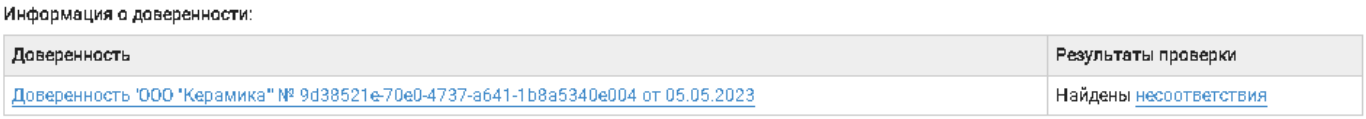


Рисунок 172. Информация о доверенности

Если в колонке *Результат проверки* указано *Найдены несоответствия*, при нажатии на колонку появится сообщение о найденных несоответствиях.

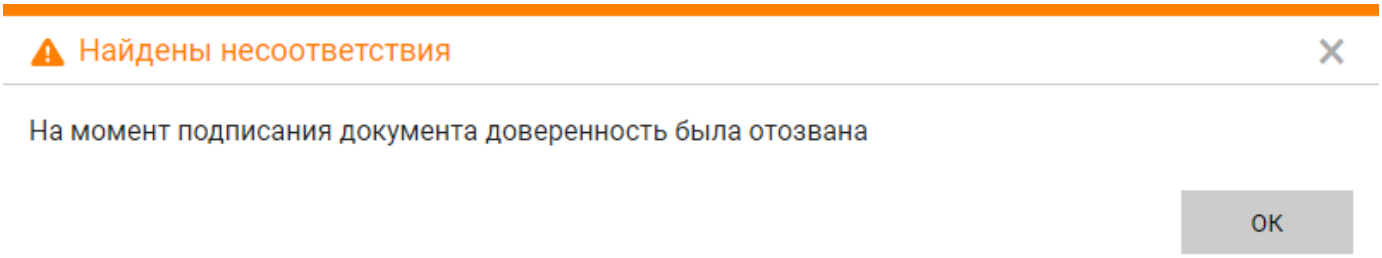


Рисунок 173. Найденные несоответствия

Если проверка документа и МЧД происходит значительно позже подписания, могут проявиться следующие сценарии:

- Проверка наличия заявления об отмене доверенности может дать положительный результат. Однако, это не означает, что доверенность была отменена до подписания документа.
- Проверка действия сертификата руководителя организации может дать отрицательный результат. Однако это не значит, что сертификат не действовал на момент подписания МЧД.

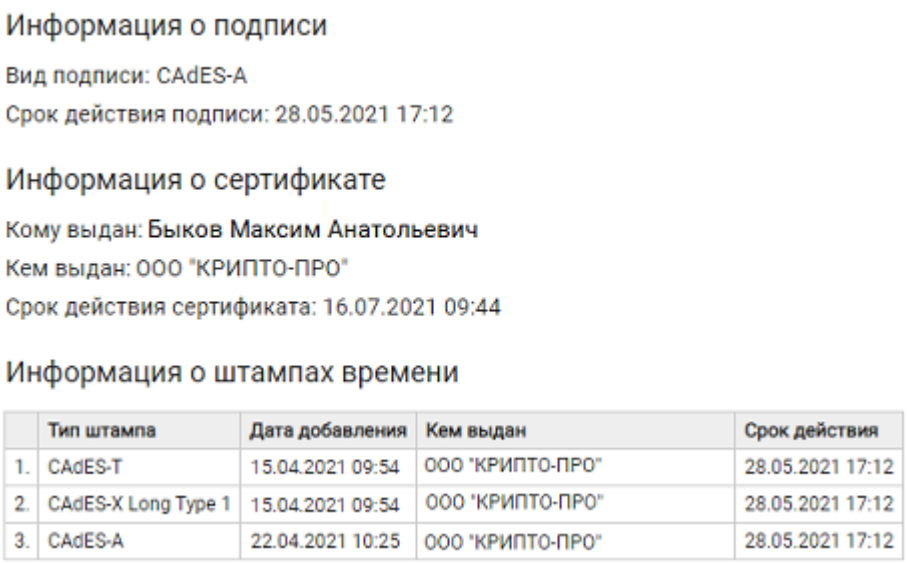
Для однозначной автоматической проверки полномочий участникам электронного документооборота необходимо использовать усовершенствованную УКЭП с меткой доверенного времени как при подписании

самого документа, так и при подписании МЧД.

Срок действия и расширенная информация о подписи

Для квалифицированной электронной подписи в журнале подписей отображаются две дополнительные колонки.

В колонке *Штамп времени* отображается точная дата подписания документа. В колонке *Срок действия* отображается дата устаревания штампа электронной подписи.



Информация о подписи

Вид подписи: CAdES-A

Срок действия подписи: 28.05.2021 17:12

Информация о сертификате

Кому выдан: Быков Максим Анатольевич

Кем выдан: ООО "КРИПТО-ПРО"

Срок действия сертификата: 16.07.2021 09:44

Информация о штампах времени

	Тип штампа	Дата добавления	Кем выдан	Срок действия
1.	CAdES-T	15.04.2021 09:54	ООО "КРИПТО-ПРО"	28.05.2021 17:12
2.	CAdES-X Long Type 1	15.04.2021 09:54	ООО "КРИПТО-ПРО"	28.05.2021 17:12
3.	CAdES-A	22.04.2021 10:25	ООО "КРИПТО-ПРО"	28.05.2021 17:12

Рисунок 174. Расширенная информация о наложенном штампе квалифицированной подписи

При нажатии на иконку рядом с результатом проверки подписи выводится расширенная информация о сертификате и штампе времени подписи. При этом самый близкий к текущей дате штамп времени выводится сверху.

Информация о сертификате:

- Тип подписи.
- Кому выдан (ФИО или организация).
- Кем выдан.
- Срок действия сертификата.

Информация о штампе времени:

- Тип штампа.
- Дата добавления.
- Кем выдан.

- Срок действия.



- Только для сервера под управлением Linux:
  - Криптосервис необходим для проверки всех подписей и формирования CAdES-BES и CAdES-X Long Type 1.
- Если сервер Docsvision работает под управлением Windows, проверка и формирование подписи работает без криптосервиса и лицензии на него, но в таком случае отображение штампа времени будет недоступно.

Распечатать документ со штампом электронной подписи

Web-клиент позволяет печатать файлы, приложенные к карточке со штампом электронной подписи. Данная возможность доступна только для указанных ниже форматов файлов с действительной ЭП (подпись верна). Файлы должны быть подписаны с помощью сертификата, возможность печати штампа для простой подписи не предусмотрена.

Поддерживаемые форматы файлов для печати со штампом подписи: `.djvu`, `.doc`, `.docm`, `.docx`, `.docxf`, `.dot`, `.dotm`, `.dotx`, `.epub`, `.fb2`, `.fodt`, `.htm`, `.html`, `.mht`, `.mhtml`, `.odt`, `.ott`, `.oxps`, `.pdf`, `.rtf`, `.stw`, `.sxw`, `.txt`, `.wps`, `.wpt`, `.xps`, `.et`, `.ett`, `.fods`, `.ods`, `.ots`, `.sxc`, `.xls`, `.xlsb`, `.xlsm`, `.xlsx`, `.xlt`, `.xltm`, `.xltx`, `.dps`, `.dpt`, `.fodp`, `.odp`, `.otp`, `.pot`, `.potm`, `.potx`, `.pps`, `.ppsm`, `.ppsx`, `.ppt`, `.pptm`, `.pptx`, `.sxi`

Чтобы распечатать документ со штампом ЭП:

1. Откройте документ, содержащий подписанный файл.
2. В меню основного файла нажмите ⋮ **> Показать с ЭП**.

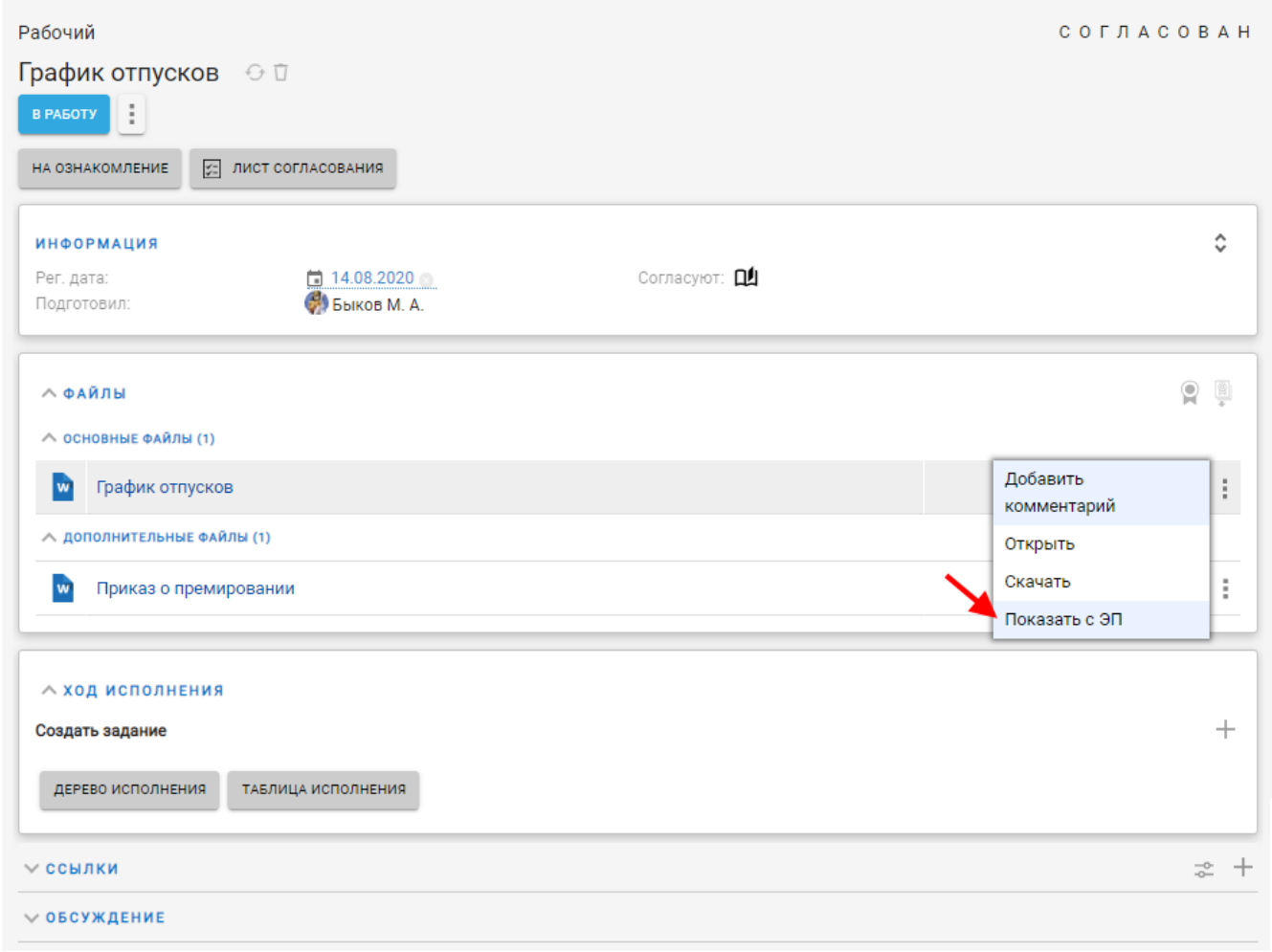
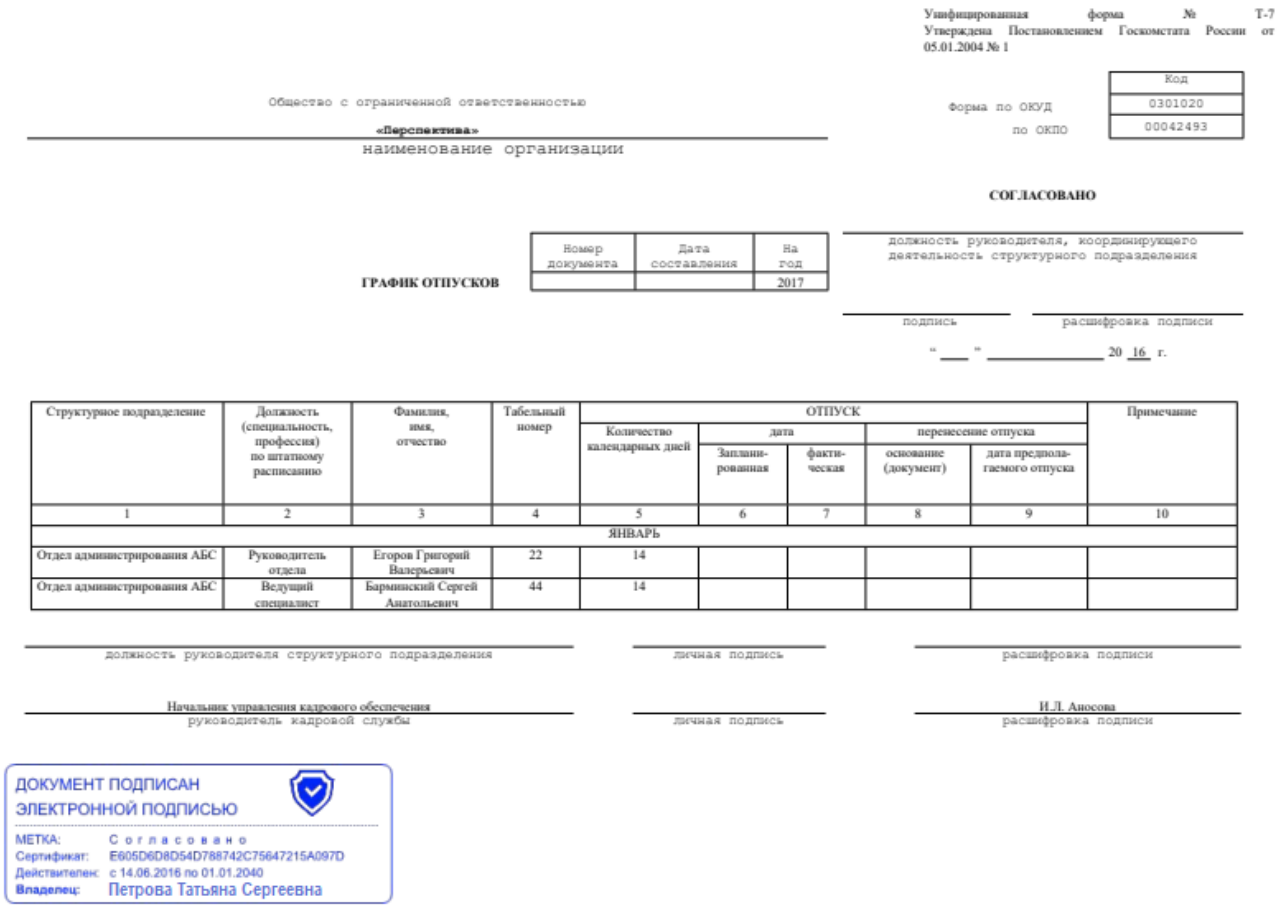


Рисунок 175. Меню печати штампа электронной подписи

Будет сформирован файл PDF с содержимым печатаемого файла и штампом электронной подписи. Если файл подписан несколькими усиленными или квалифицированными подписями, штампы всех подписей будут представлены в печатной версии.



Унифицированная форма № Т-7
Утверждена Постановлением Госкомстата России от 05.01.2004 № 1

Общество с ограниченной ответственностью
«Перспектива»
наименование организации

Форма по ОКУД 0301020
по ОКПО 00042493

СОГЛАСОВАНО

должность руководителя, координирующего деятельность структурного подразделения

подпись расшифровка подписи

"___" _____ 20 16 г.

ГРАФИК ОТПУСКОВ

Номер документа	Дата составления	На год
		2017

Структурное подразделение	Должность (специальность, профессия) по штатному расписанию	Фамилия, имя, отчество	Табельный номер	ОТПУСК					Примечание
				Количество календарных дней	дата		перенесение отпуска		
					Запланированная	фактическая	основание (документ)	дата предполагаемого отпуска	
1	2	3	4	5	6	7	8	9	10
ЯНВАРЬ									
Отдел администрирования АБС	Руководитель отдела	Егоров Григорий Валерьевич	22	14					
Отдел администрирования АБС	Ведущий специалист	Барминский Сергей Анатольевич	44	14					

должность руководителя структурного подразделения — личная подпись — расшифровка подписи

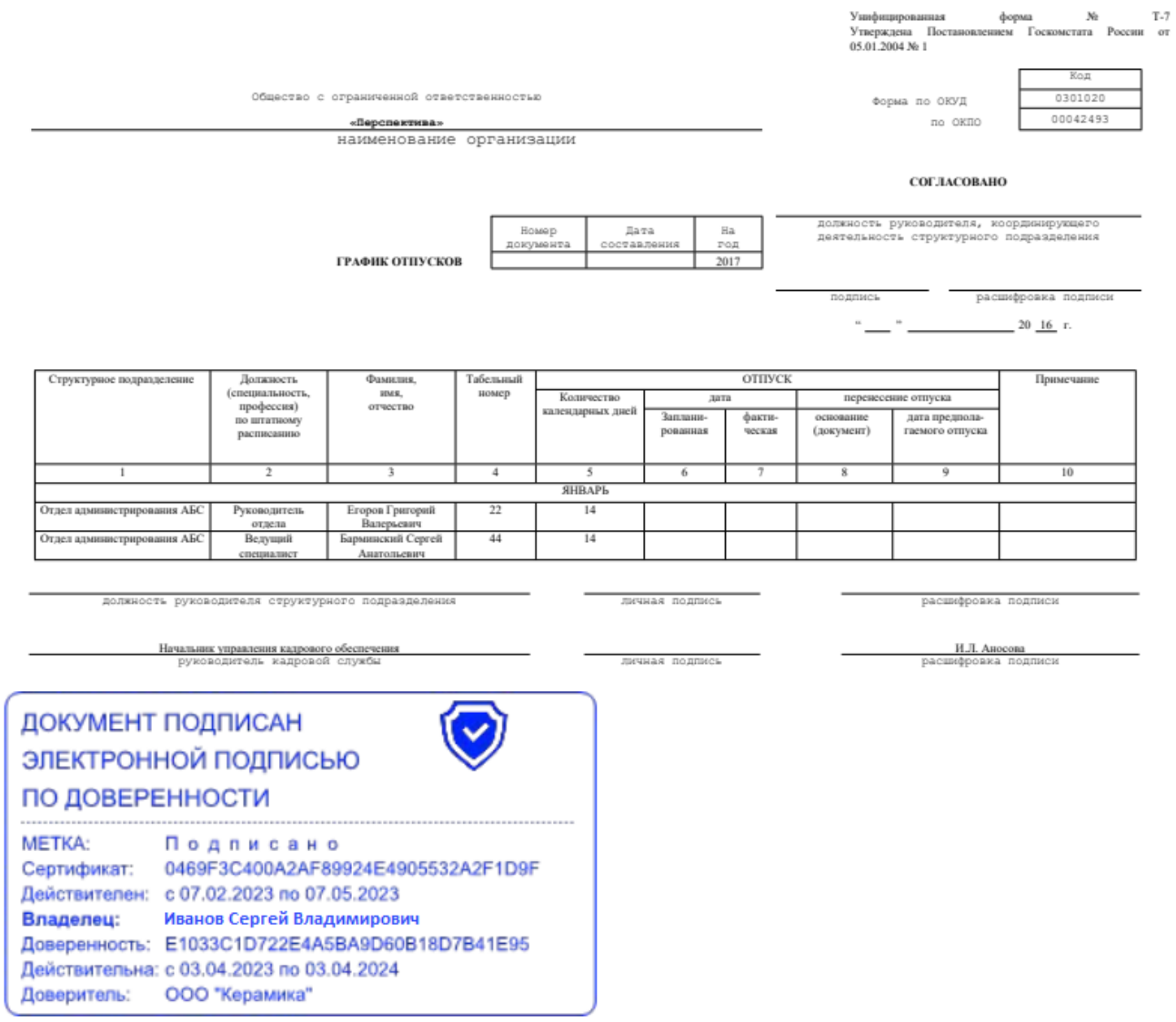
Начальник управления кадрового обеспечения (руководитель кадровой службы) — личная подпись — И.Л. Аносова (расшифровка подписи)

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
МЕТКА: Согласовано
Сертификат: E605D6D8D54D788742C75647215A097D
Действителен: с 14.06.2016 по 01.01.2040
Владелец: Петрова Татьяна Сергеевна

Рисунок 176. Пример файла со штампом электронной подписи

Штамп ЭП включает в себя метку подписи, номер, срок действия и владельца сертификата ЭП, а также логотип, который может быть настроен.

Штампы можно перемещать в пределах страницы `.pdf` файла, а также изменять их размеры. Данная возможность должна поддерживаться используемым средством просмотра `.pdf` файлов.



Унифицированная форма № Т-7
Утверждена Постановлением Госкомстата России от 05.01.2004 № 1

Общество с ограниченной ответственностью
«Перспектива»
наименование организации

Форма по ОКУД 0301020
по ОКПО 00042493

СОГЛАСОВАНО

должность руководителя, координирующего деятельность структурного подразделения

подпись расшифровка подписи

" " 20 16 г.

ГРАФИК ОТПУСКОВ

Номер документа	Дата составления	На год
		2017

Структурное подразделение	Должность (специальность, профессия) по штатному расписанию	Фамилия, имя, отчество	Табельный номер	ОТПУСК					Примечание
				Количество календарных дней	дата		перенесение отпуска		
					Запланированная	фактическая	основание (документ)	дата предполагаемого отпуска	
1	2	3	4	5	6	7	8	9	10
ЯНВАРЬ									
Отдел администрирования АБС	Руководитель отдела	Егоров Григорий Валерьевич	22	14					
Отдел администрирования АБС	Ведущий специалист	Барминский Сергей Анатольевич	44	14					

должность руководителя структурного подразделения — личная подпись — расшифровка подписи

Начальник управления кадрового обеспечения (руководитель кадровой службы) — личная подпись — И.Л. Аносова (расшифровка подписи)

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ ПО ДОВЕРЕННОСТИ

МЕТКА: Подписано
Сертификат: 0469F3C400A2AF89924E4905532A2F1D9F
Действителен: с 07.02.2023 по 07.05.2023
Владелец: Иванов Сергей Владимирович
Доверенность: E1033C1D722E4A5BA9D60B18D7B41E95
Действительна: с 03.04.2023 по 03.04.2024
Доверитель: ООО "Керамика"

Рисунок 177. Пример файла, подписанного по доверенности

Если при подписании использовалась МЧД, это будет указано на штампе ЭП. На визуализации штампа отображается следующая информация:

- *Доверенность*: указывается регистрационный номер доверенности.
- *Действительна*: указывается дата совершения доверенности и срок окончания действия.
- *Доверитель*: указывается название доверителя.



Убедитесь, что в Windows-клиенте указан *Адрес сервера Web-клиента*, см. подробнее в разделе "Системные настройки модуля Windows-клиент". Чтобы распечатать документ с ЭП требуется установить и настроить Р7-Офис по инструкции.

Экспортировать файлы с подписями

Подписанные основные файлы можно сохранить на устройство пользователя вместе с электронными подписями.



Чтобы экспортировать файлы с подписями, необходимо обладать правами на скачивание файлов карточки. Экспорт доступен, если хотя бы один основной файл подписан действительной электронной подписью с сертификатом. При наличии только простых подписей кнопка экспорта файлов с подписями не отображается.

1. Откройте карточку документа в режиме чтения.
2. В разделе *Файлы* нажмите .



Кнопка недоступна, если в карточке отсутствуют подписанные основные файлы.

На устройство пользователя будет сохранён `.zip` архив, содержащий текущие версии подписанных основных файлов с файлами отсоединённых электронных подписей. Основные файлы и отсоединённые подписи будут сгруппированы по папкам.

Экспортируются только действительные подписи текущих версий основных файлов.

При экспорте файлов с ЭП может также экспортироваться МЧД. Экспорт МЧД с файлом отсоединённой подписи будет выполнен только, если активно свойство **Экспорт с МЧД** в элементе управления Экспорт подписей.

В случае МЧД, выданной в порядке передоверия, сохраняется вся цепочка МЧД с подписями.

Если МЧД не использовалась, будет экспортирована только подпись.

Загрузить отсоединённую электронную подпись файла

Web-клиент поддерживает возможность загрузки отсоединённой электронной подписи в карточку Документа для подписания основного файла.

Отсоединённая ЭП — файл формата `.p7s`, `.sig`, `.sgn` или `.sign`, содержащий

электронную подпись подписанного файла. Отсоединённая ЭП может быть загружена с использованием кодировки DER или BASE-64.



Web-клиент поддерживает загрузку отсоединённых подписей только основных файлов.

Чтобы загрузить отсоединённую подпись:

1. Откройте карточку Документ, в которую загружается отсоединённая подпись.
2. Перейдите к разделу *Файлы*.
3. Загрузите электронную подпись:

   Если карточка Документ содержит файл, для которого загружается отсоединённая подпись:

   a. В строке основного файла, для которого загружается подпись, нажмите ⋮ > **Загрузить ЭП**.

   b. Выберите файл отсоединённой подписи (.p7s, .sig, .sgn, .sign).

   c. Нажмите **Открыть**, чтобы загрузить выбранные файлы.

   Если файл с отсоединённой подписью загружается вместе с подписью:

   a. Нажмите 🗁 > **+Основной**.

   b. Выберите подписанный файл и файл, содержащий отсоединённую подпись (.p7s, .sig, .sgn, .sign).

   c. Нажмите **Открыть**, чтобы загрузить выбранные файлы.

В атрибуте *Имя файла* отсоединённой электронной подписи должно содержаться название файла, которому принадлежит данная подпись, иначе загрузка будет невозможна.

Возможность загрузки электронной подписи будет недоступна, если запрещена операция редактирования для добавления подписи.

Если карточка Документ содержит файл с отсоединённой подписью или файл был загружен вместе с отсоединённой подписью, подпись будет прикреплена к этому файлу. Подпись прикрепляется к текущей версии основного файла. Проверить, к какому файлу прикреплена подпись можно с помощью журнала электронных подписей.

Если файл, к которому должна быть присоединена подпись, отсутствует в документе, будет выдано сообщение о невозможности загрузки подписи. Другие файлы, выбранные при загрузке, будут загружены при наличии прав.

Если присоединяемая подпись недействительна, будет выдано сообщение об ошибке. Подпись загружена не будет.

Если оставшийся срок действия сертификата подписи составляет менее 5 дней, будет выдано предупреждение: `Срок действия сертификата закончится <Дата>`. Пользователь может подтвердить или отменить загрузку подписи.

Если выбран файл с **присоединённой** электронной подписью (файл-контейнер, содержащий и подписанный файл и электронную подпись), будет выдано предупреждение: `Необходимо использовать отсоединенную подпись`.

Алгоритм выбора сертификата

Сертификаты в диалоге подписания отображаются в установленном порядке или в порядке, определённом пользователем.

По умолчанию всегда выбирается первый сертификат из списка. Список либо формируется пользователем самостоятельно, либо автоматически.

Список, сформированный автоматически, состоит из простой подписи, сертификатов пользователя и сертификата из карточки сотрудника в справочнике сотрудников. Если в карточке сотрудника в справочнике сотрудников указан сертификат, которого нет в списке, он будет добавлен в начало списка.



В зависимости от настроек, выполненных администратором Docsvision, диалог подписания может не отображаться.

Использование простой подписи может быть запрещено, в таком случае она не отображается в списке. Если в списке отсутствуют и сертификаты и простая подпись, будет предложено настроить сертификаты.

Подписание облачной подписью

В дополнение к локальным сертификатам можно настроить облачные. Для настройки потребуется авторизоваться в сервисе электронной подписи КриптоПро.

Настройки сертификатов доступны:

- Из диалога подписания документа.
- Из профиля пользователя.

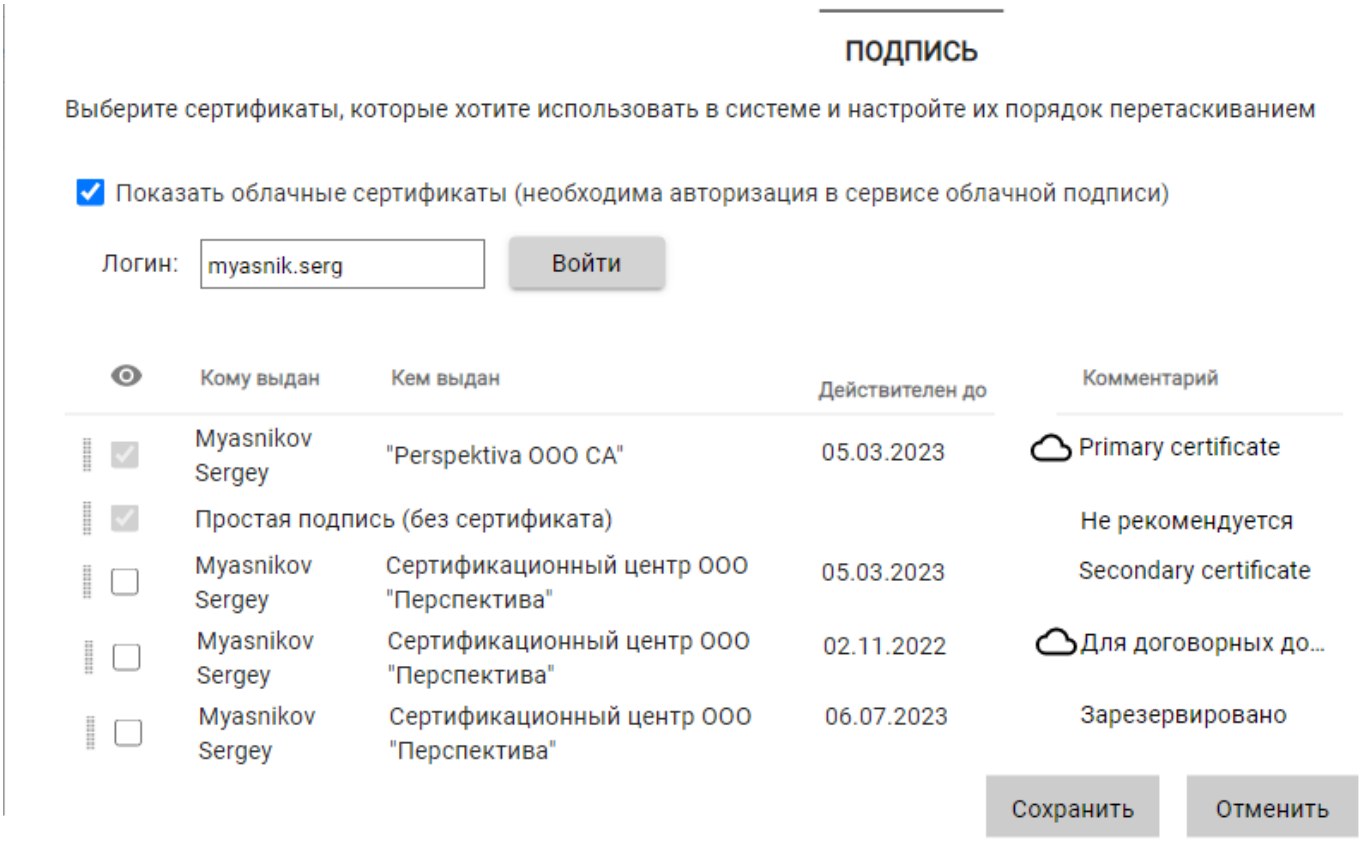


Рисунок 178. Локальные и облачные сертификаты на вкладке "Подпись"

Чтобы отобразить облачные сертификаты на вкладке:

1. Введите *Логин* в одноимённое поле и нажмите кнопку **Войти и обновить список сертификатов**.

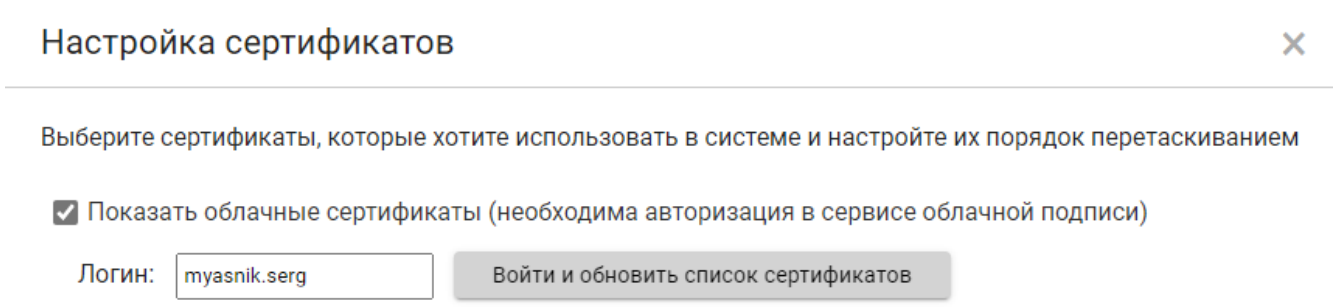


Рисунок 179. Вход в сервис электронной подписи КриптоПро

2. В новой вкладке откроется окно авторизации в сервисе электронной подписи КриптоПро.

   Если диалог авторизации был закрыт досрочно, возникнет ошибка.

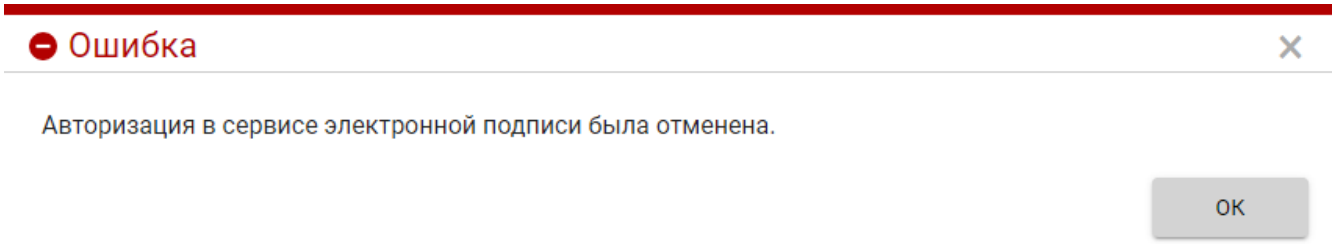


Рисунок 180. Авторизация отменена

3. Введите пароль и при необходимости поставьте флаг Запомнить пароль, затем нажмите кнопку **Войти**.

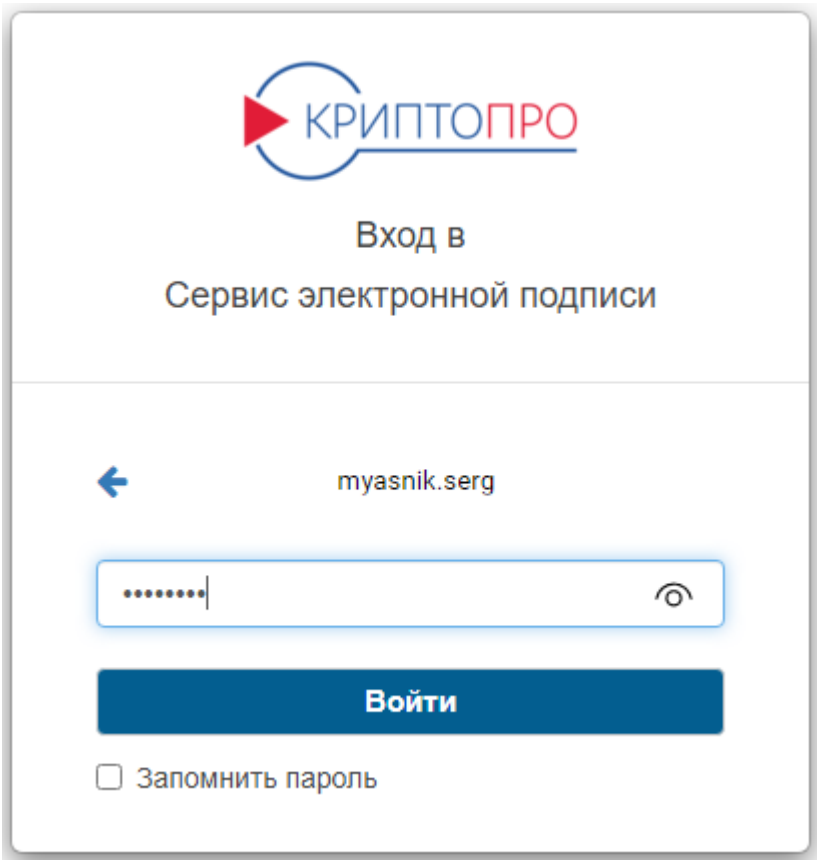


Рисунок 181. Авторизация в сервисе ЭП КриптоПро

Если в сервисе электронной подписи настроена двухфакторная аутентификация, будет предложено ввести код подтверждения из СМС, электронной почты или подтвердить вход в приложении.
В приложении при подтверждении отображаются файлы для подписи, если таковые есть.

4. Вкладка будет закрыта, и в окне *Настройки подписи* отобразятся облачные сертификаты.
5. Снимите флаг Показать облачные сертификаты, чтобы облачные сертификаты не отображались в общем списке. Выбранные для отображения сертификаты не

будут скрыты.

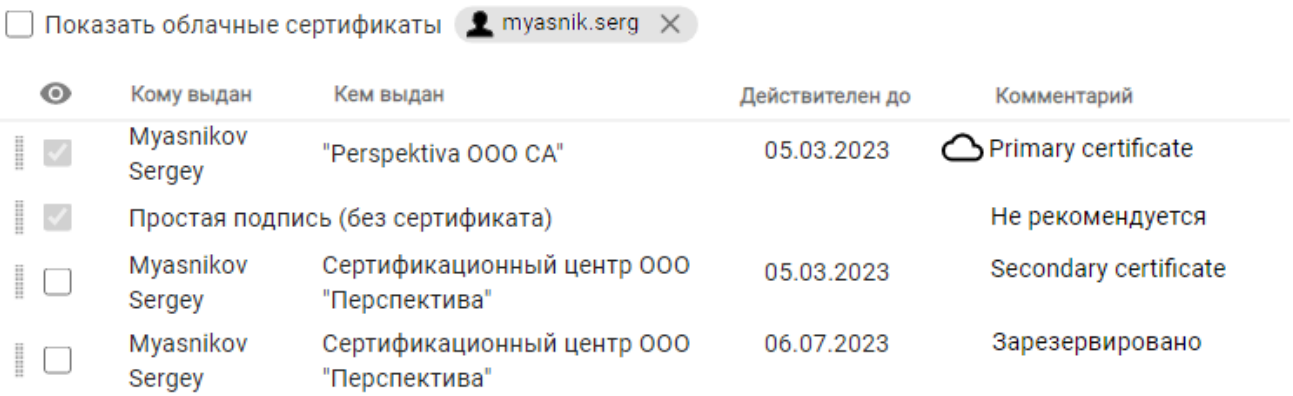


Рисунок 182. Не показывать облачные сертификаты

6. Настройте отображение сертификатов в диалоге подписания, установив флаги.
7. Настройте порядок следования сертификатов перетаскиванием.

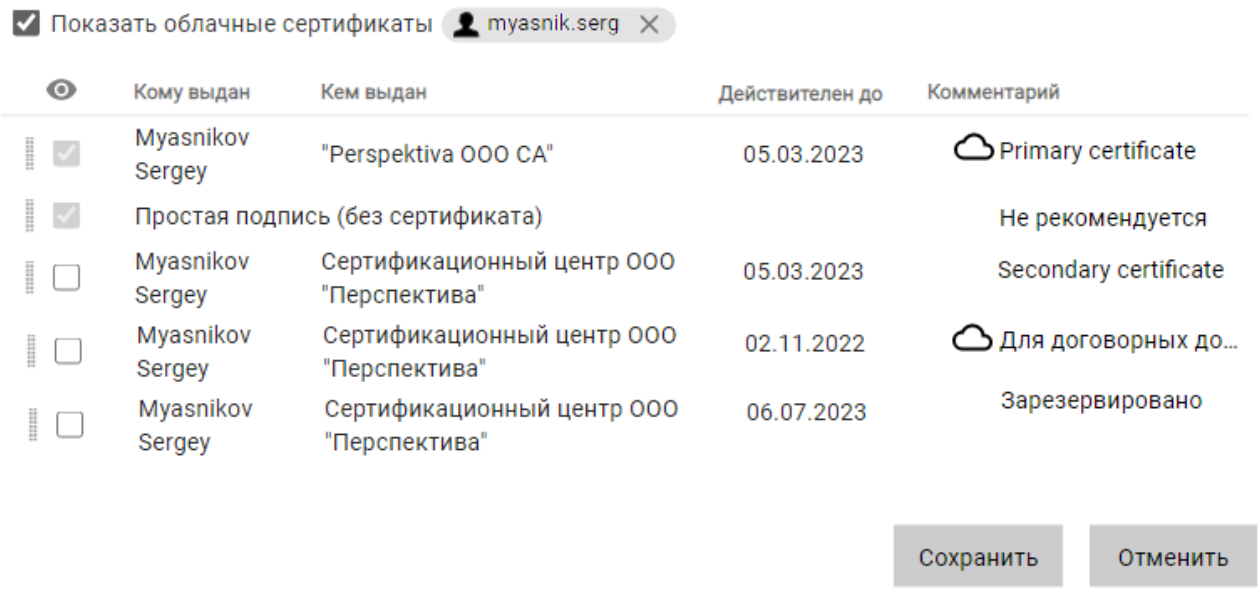


Рисунок 183. Отображение и порядок следования сертификатов

8. Нажмите крестик ✕ напротив имени пользователя, чтобы сбросить авторизацию.
9. Нажмите кнопку **Сохранить**.

10. Настройте проверку облачной подписи.

Назначить документ действующим

Действующими могут быть назначены документы видов: *Заявка*, *ОРД*, *Рабочий* и *Служебная записка*. Возможность назначения действующими документов собственных видов определяется настройками.

Назначаемый действующим документ должен быть в состоянии:

- *Зарегистрирован*, *В архиве*. Для видов документа *Заявка* и *Служебная записка*.
- *Подготовка*, *Согласован* или *Подписан*, *В архиве*. Для вида документа *ОРД*.
- *Подготовка*, *Согласован*. Для вида документа *Рабочий*.

Чтобы назначить документ действующим:

1. Откройте карточку документа в режиме чтения.
2. На панели операций нажмите кнопку **Назначить действующим**. Состояние документа изменится на `Действует`.

Отправить документ в архив

Операция перевода в архив выполняется, когда закончена активная работа с документом.

Не следует путать отправку документа в архив с архивацией карточек.

В архив можно отправить документы видов: *Входящий*, *Исходящий*, *ОРД*, *Служебная записка* и *Заявка*.

Возможность отправки в архив документов собственных видов определяется настройками.

Отправляемый в архив документ должен быть в состоянии:

- *Зарегистрирован*. Для видов документа *Входящий* и *Исходящий*.
- *Зарегистрирован* или *Действует*. Для видов документа *Заявка* и *Служебная записка*.
- *Действует*. Для вида документа *ОРД*.

Чтобы отправить документ в архив:

1. Откройте карточку документа в режиме чтения.
2. На панели операций нажмите кнопку **В архив**. Состояние документа изменится на В архиве.

Проверить уникальность Документа

При настройке Web-клиента и системы Docsvision для карточек Документ может быть включена проверка уникальности введённых данных. Web-клиент будет проверять вводимую информацию на совпадение с другими документами.

Если документ неуникален, будет выдано предупреждение. В зависимости от настроек, может также быть заблокирована возможность сохранить карточку и/или изменить её состояние.



По умолчанию уникальность атрибутов проверяется только при создании входящего документа. Для других видов документов проверка может быть включена при настройке Web-клиента.

Если проверка уникальности включена и в карточке обнаружены данные, нарушающие уникальность неуникальные атрибуты выделяются цветом.

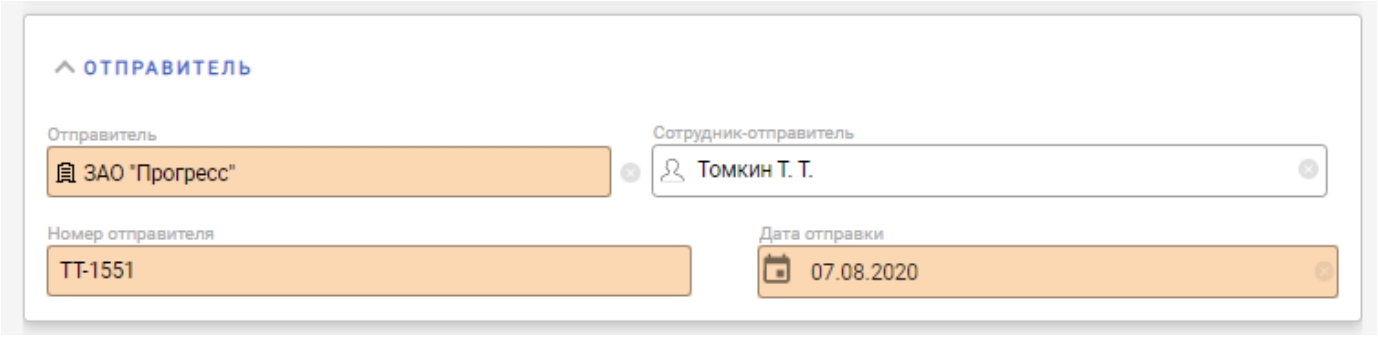


Рисунок 184. Комбинация значений отправитель, номер и дата отправления документа нарушает уникальность документа

В карточке появляется значок ⚠. Расположение значка определяется при настройке Web-клиента. При нажатии на значок будет показан список нарушений.

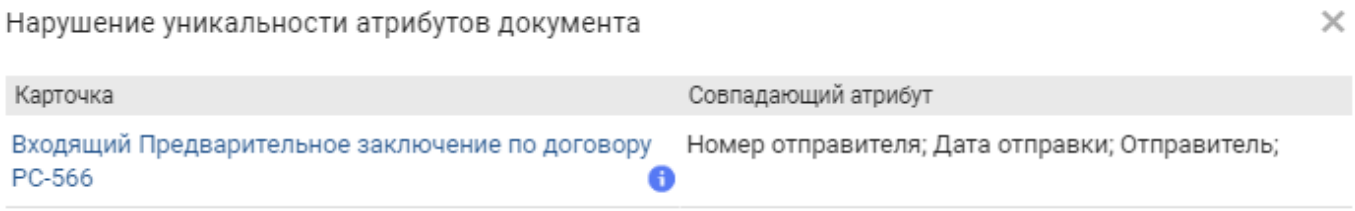


Рисунок 185. Список конфликтующих карточек и атрибутов

В списке указываются атрибуты карточки, нарушающие уникальность и другие карточки, с которыми конфликтует текущая. Если уникальность нарушают поля, не отображаемые в карточке, они также будут указаны в данном списке.

Чтобы посмотреть информацию о карточке, с которой конфликтует текущая, нажмите на значок . Чтобы перейти к карточке, нажмите на её название.

Блокировка сохранения карточек с нарушенной уникальностью

Если включена блокировка сохранения карточек с нарушенной уникальностью, при сохранении будет показано предупреждение со списком нарушений. Сохранение будет отменено.

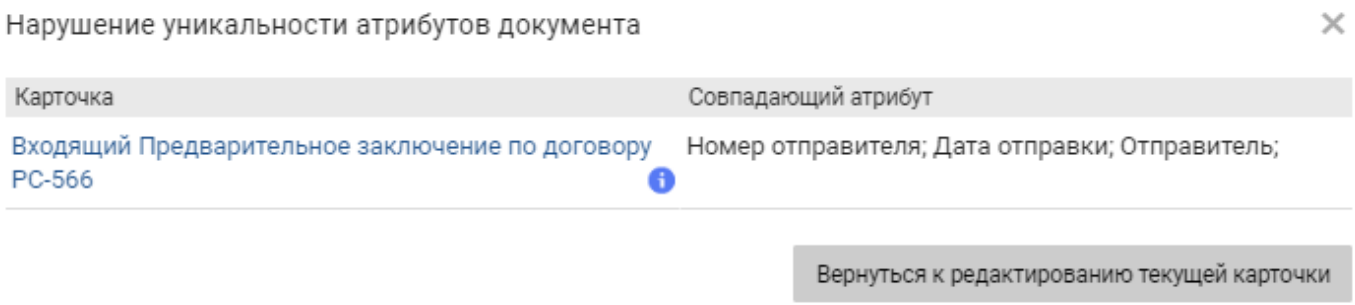


Рисунок 186. Список нарушений уникальности при сохранении карточки без блокировки сохранения

Необходимо вернуться к редактированию карточки или изменить атрибуты карточки, из-за которой нарушается уникальность текущей.

Если блокировка сохранения выключена, при сохранении неуникальной карточки также будет преложено исправить её или продолжить сохранение (команда **Создать карточку с совпадающими атрибутами**).

Проверка уникальности может выполняться некорректно для значений элемента Текст, если для данного элемента включён режим расширенного редактирования.

Списать документ в дело из карточки

Используйте только совместимые версии модулей Управление архивом и Web-клиент. Версия 6.2 и несовместима с версией 5.5.

По вопросам совместимости Управление архивом версий 6.1 и 6.2 с Web-клиентом версии 6.1 и выше проконсультируйтесь со специалистом Docsvision.

Пользователь имеет возможность списать документ в дело из карточки. Возможность становится доступной после установки компонентов модуля *Управление архивом* и выполнения необходимых настроек разметок Web-клиента в программе *Конструктор Web-разметок*. Документ можно списать только в уже созданное дело.

По умолчанию списывать документы в дела документы может только сотрудник с ролью *Ответственный*. Чтобы все сотрудники подразделения могли списывать документы в дела своего раздела, необходимо произвести дополнительные настройки. Подробно настройка описана в руководстве администратора модуля *Управление архивом*, пункт "Настройка ролевой модели справочника номенклатуры дел".

Чтобы списать документ в дело:

1. Из карточки вида Документ выберите элемент управления Дело:



Элемент управления представляет собой строку справочника и может называться иначе в зависимости от настроек разметки карточки.

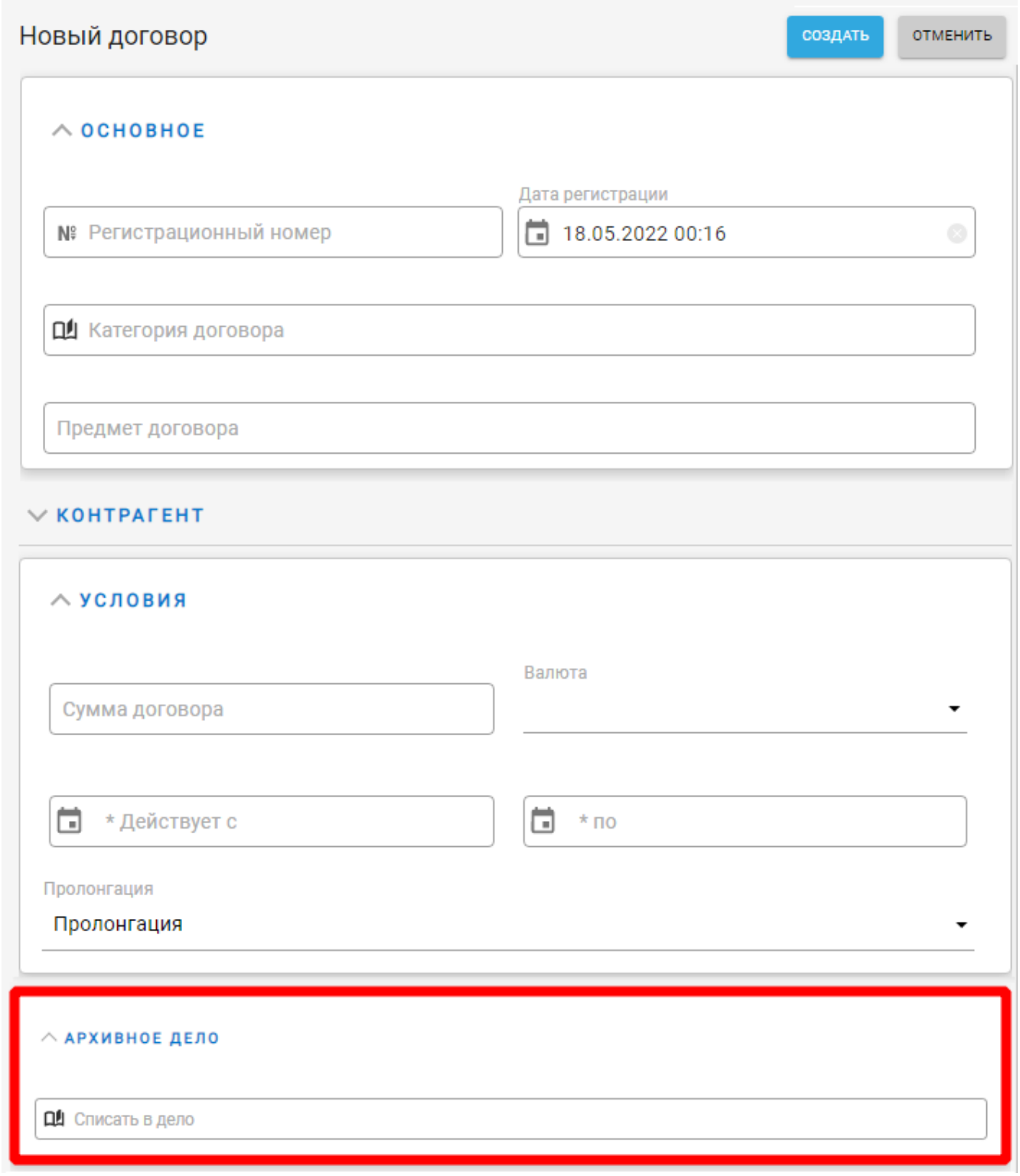


Рисунок 187. Возможный вид элемента управления для списания документа в архив

2. Выберите дело, в которое должен быть списан документ:

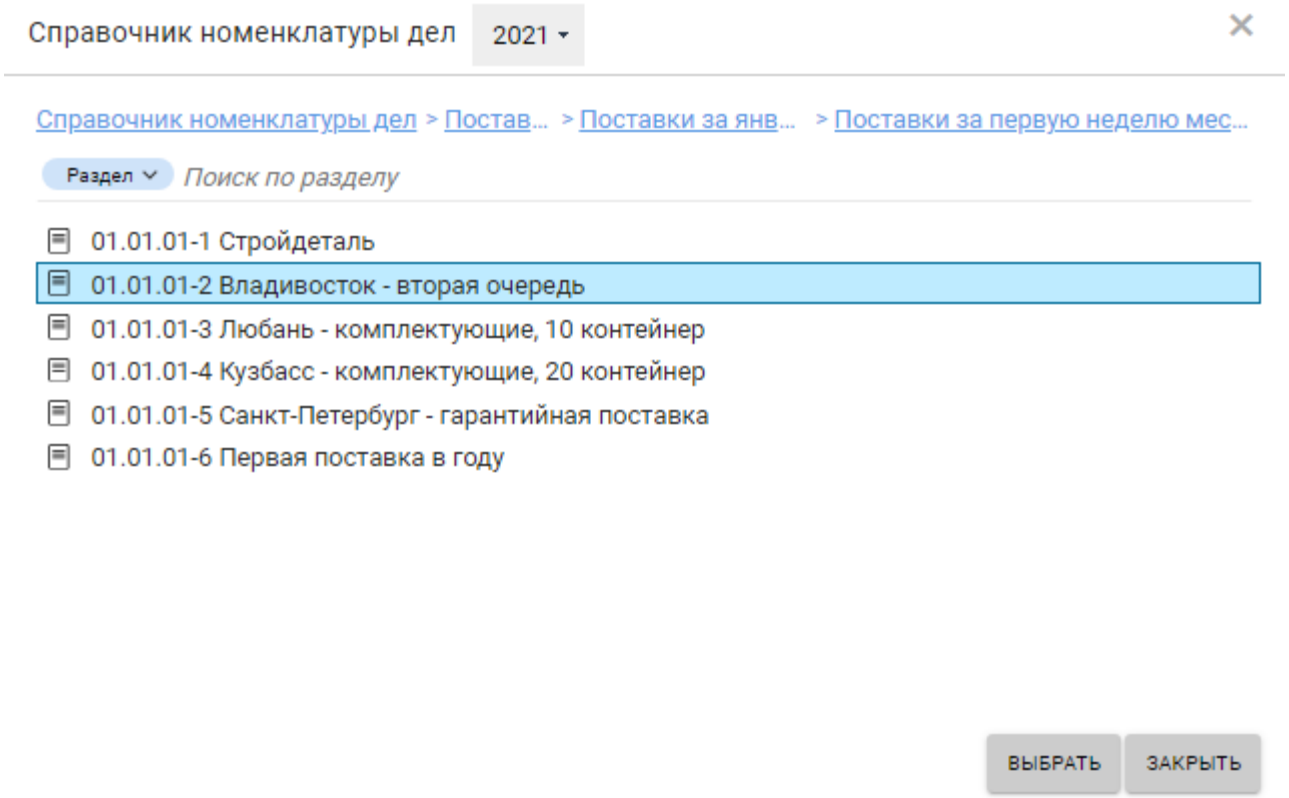


Рисунок 188. Выбор дела из справочника

Выбрать дело из справочника можно двумя способами:

- Через быстрый поиск из строки справочника.

  Поиск осуществляется по *заголовку* и *индексу* всех дел справочника, а также по *наименованию* и *индексу* разделов. Быстрый поиск по умолчанию осуществляется по наиболее близкому к текущей дате году. Чтобы выбрать дело из другого года, необходимо перейти в справочник.

- Из справочника, нажав на иконку 🕮.

  В окне справочника также может осуществляться поиск по разделам или по индексам и заголовкам дел, также доступен выбор года.

  При быстром поиске и поиске из справочника происходит проверка прав. Это значит, что пользователь увидит только те элементы, к которым у него есть доступ на чтение, изменение и списание документа в дело.

- Подробнее о правах доступа в пункте Настройки безопасности

справочника номенклатуры дел.

3. Нажмите **Выбрать**.

Документ будет списан в выбранное дело, в деле появится ссылка на карточку документа. Информация о том, что карточка списана в дело появится, только если такая настройка была выполнена в программе Конструктор Web-разметок.

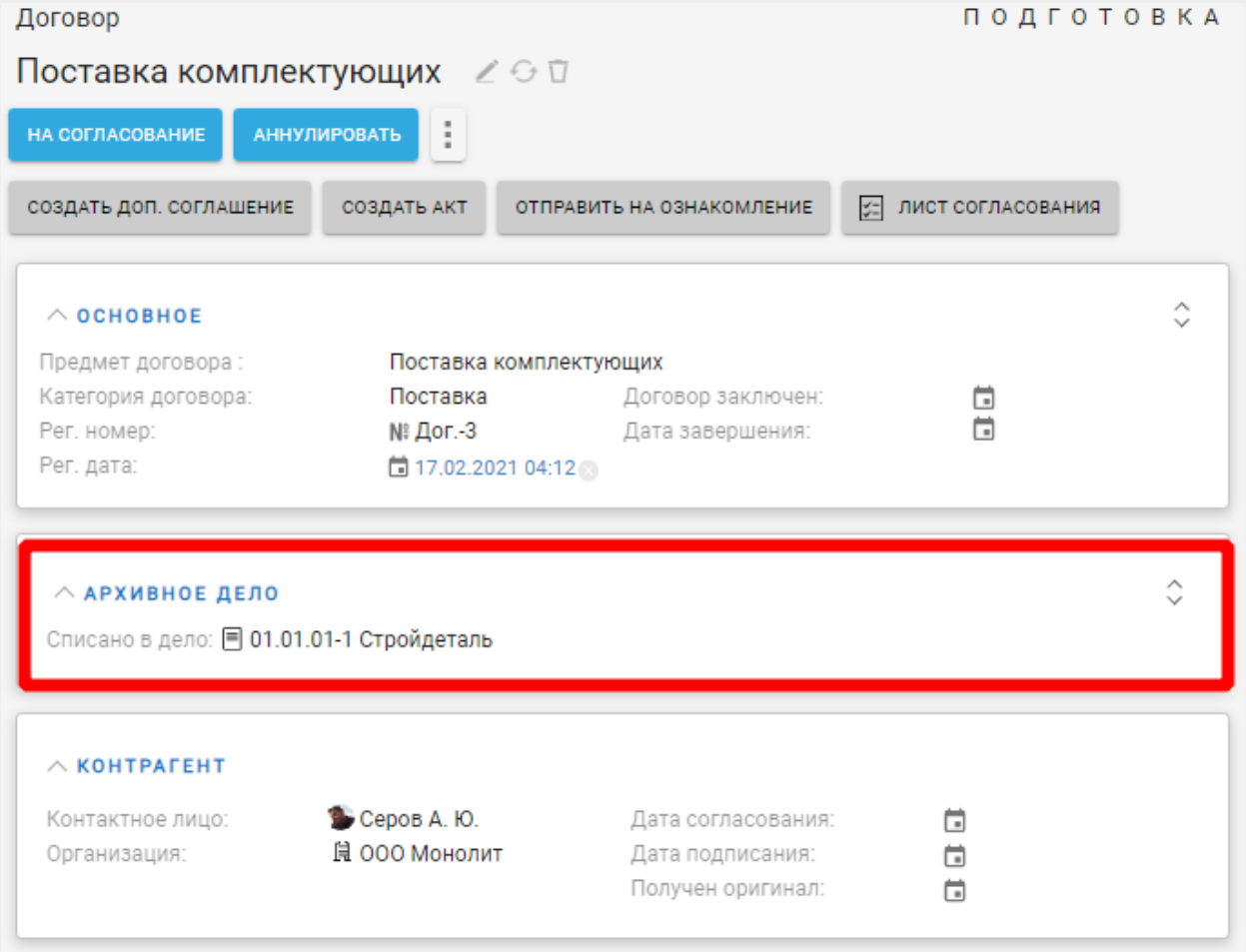


Рисунок 189. Информация о списании документа в дело

При добавлении в дело первого документа, в деле автоматически создаётся *том*. При дальнейшем добавлении документов в дело, все документы добавляются в созданный *том*.

Отправить карточку на печать

Если в карточку добавлен элемент управления "Кнопка печати", карточку можно отправить на печать.

Чтобы отправить карточку на печать:

1. Нажмите кнопку **Печать**.
2. Выберите подходящий шаблон из списка. Если доступен только один шаблон, он применится по умолчанию.

   В новой вкладке откроется карточка без интерфейса Web-клиента.

   Если вы работаете в Web-клиенте через браузер Google Chrome, сначала появится новое модальное окно. Карточка для печати откроется только после нажатия кнопки **Печать** в модальном окне. Интерфейс Web-клиента будет недоступен для работы, пока открыто модальное окно на печать и не нажата кнопка печати. Данное поведение является специфичным для браузера Google Chrome и не распространяется на другие браузеры.

3. Отправьте карточку на печать.

HTML-форматирование:

Если в карточке присутствуют поля с HTML-кодом и для печати используются стандартные шаблоны, текст в таких полях будет выводиться на печать в отформатированном виде.

По умолчанию HTML-форматирование учитывается только в БД с версией Базовые объекты 5.5.5 и выше. Для БД с Базовые объекты более ранних версий необходимо вручную отредактировать шаблоны для карточки *Документ*, привязанные на вкладке *Экспорт* в *Справочнике видов карточек*. Стандартные шаблоны хранятся в папке `C:\Program Files\Docsvision\Platform\Tools\Database\BackOffice\Xslt`.

В справочнике видов можно указать актуальные шаблоны или вручную изменить стандартные. В шаблоне необходимо обернуть HTML-текст в тег `<EscapeHTML>`.

Работа с договорными документами

Web-клиент предоставляет возможность регистрировать и согласовывать договоры и дополнительные соглашения к ним. При этом используются карточки видов:

Договор

Предназначена для хранения данных о заключаемом договоре.

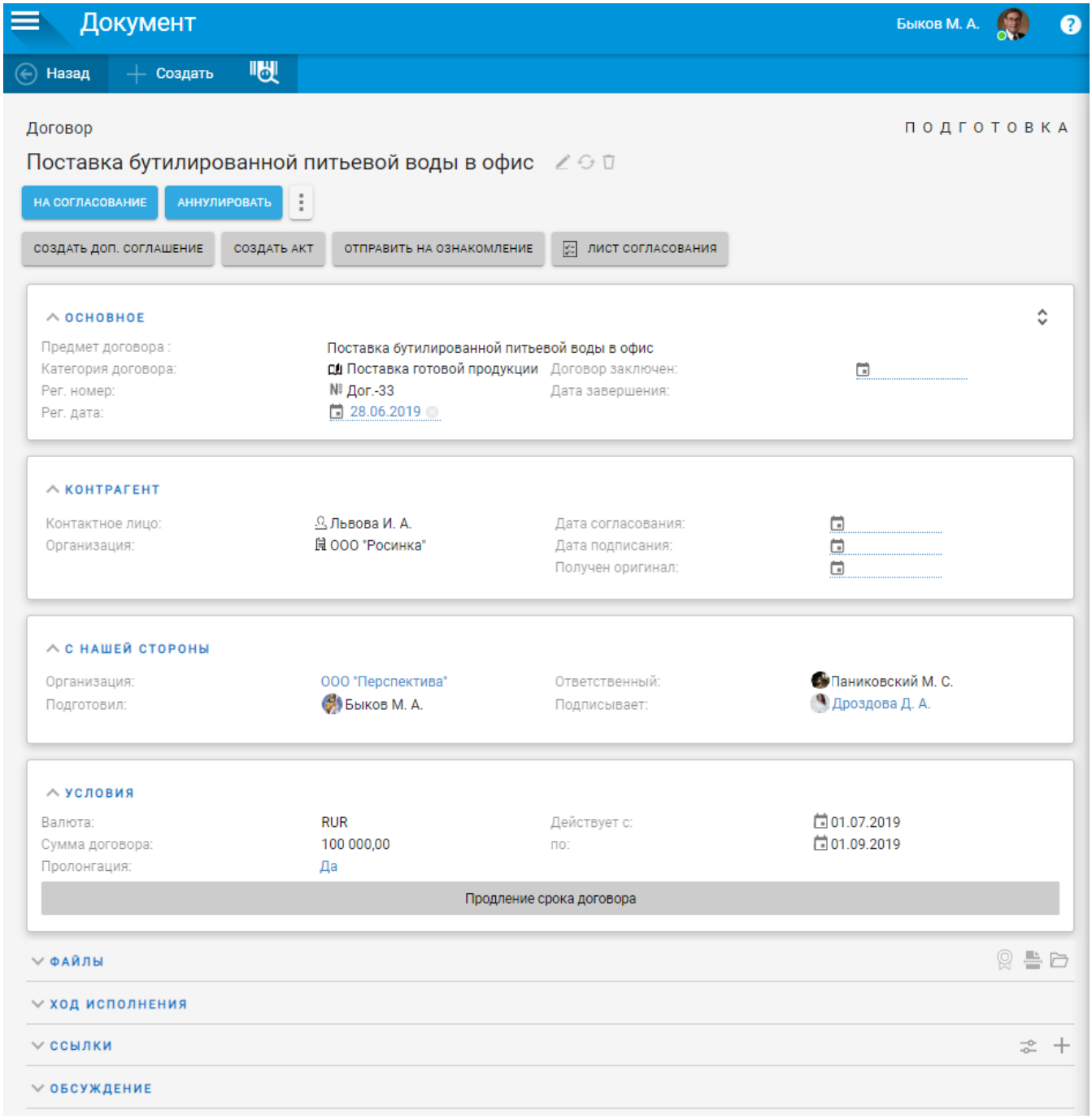


Рисунок 190. Карточка "Договор"

Дополнительное соглашение

Предназначена для хранения сведений о документе, который прилагается к заключённому ранее договору какой-либо сделки. Предполагается, что согласно данному соглашению стороны фиксируют возможные изменения, которые состоялись после заключения договора.

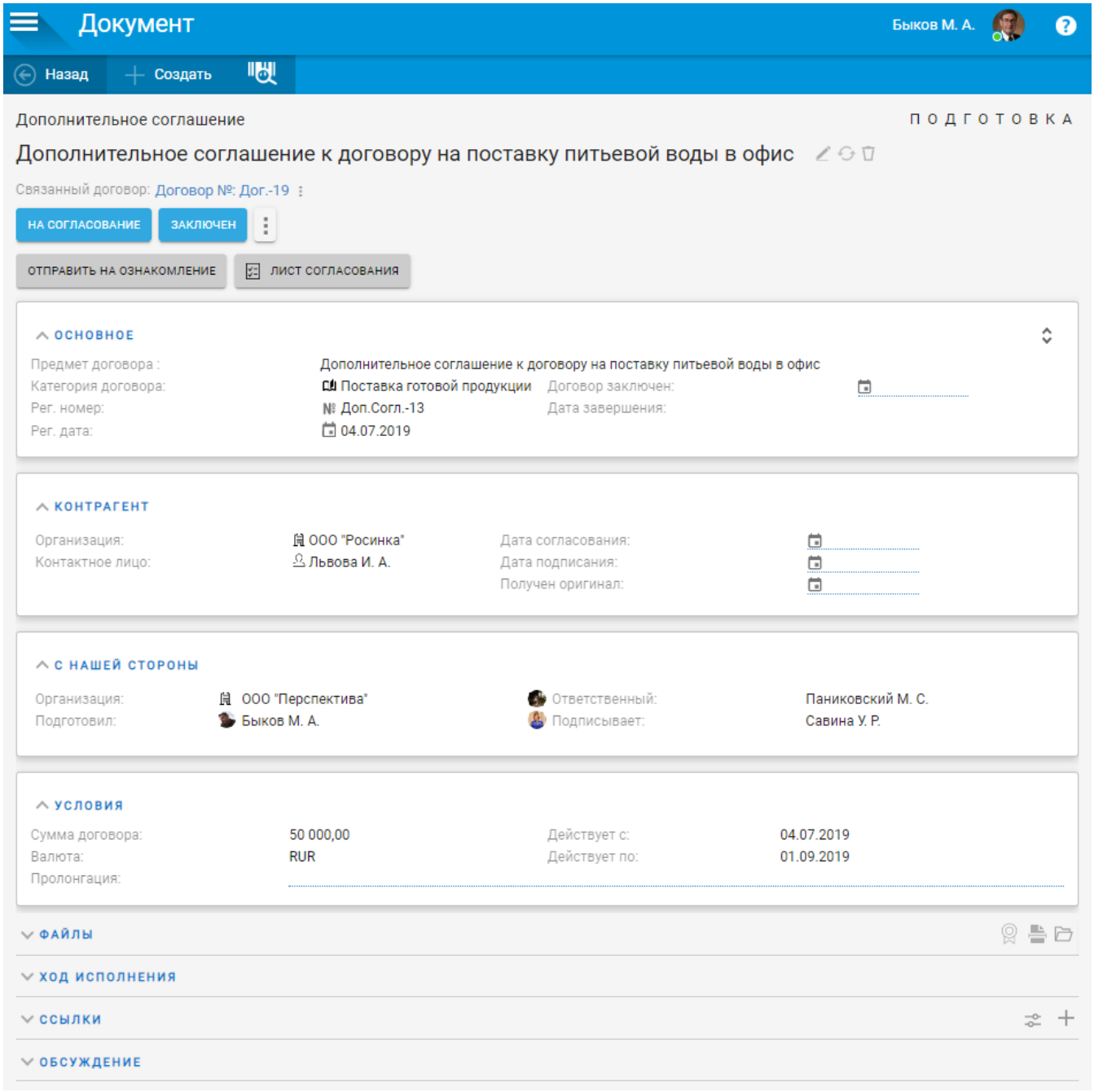


Рисунок 191. Карточка "Дополнительное соглашение"

Акт

Предназначена для хранения сведений об акте, который подписывается уполномоченными должностными лицами.

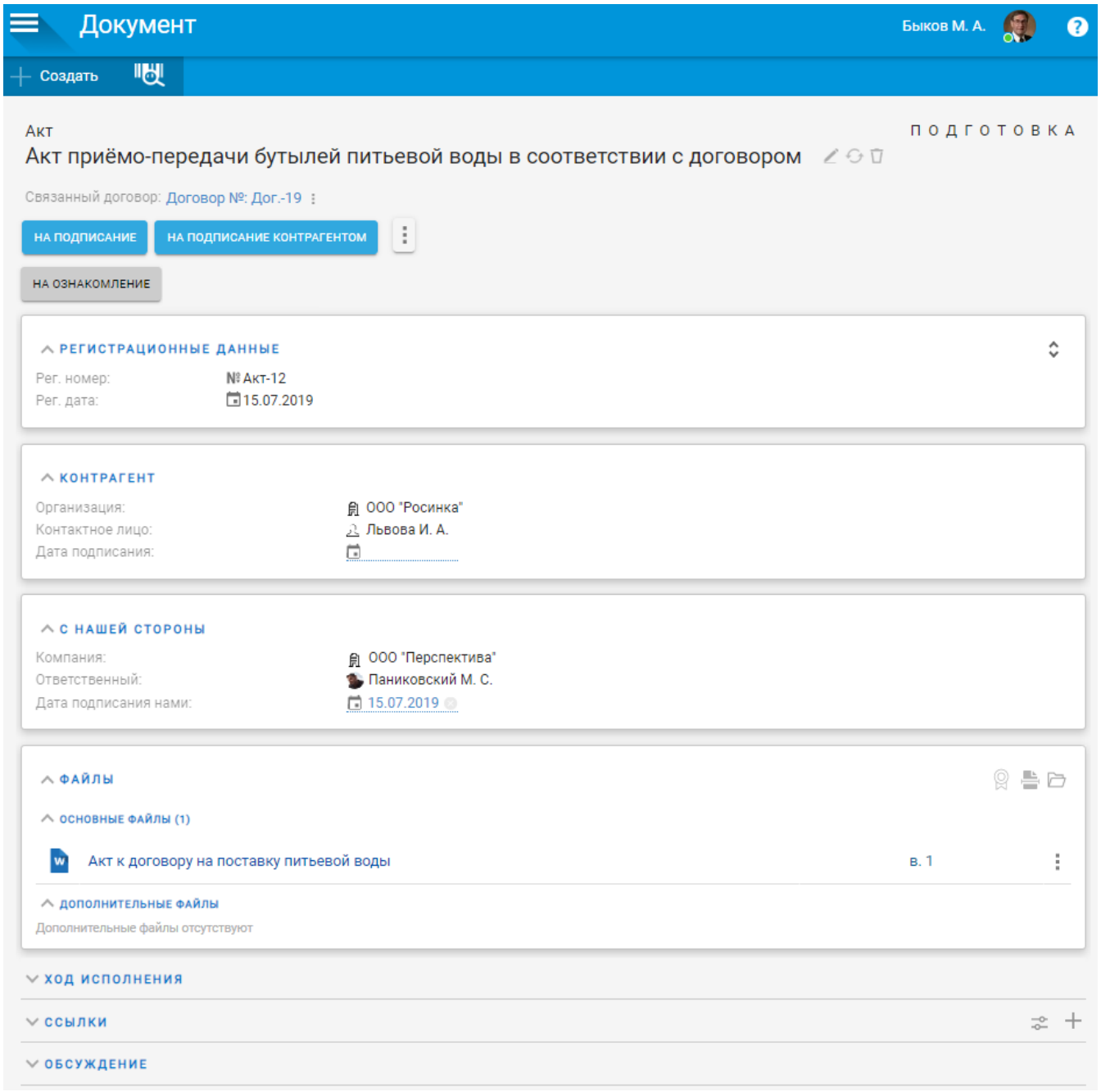


Рисунок 192. Карточка "Акт"

На представленных выше изображениях скрыта область предварительного просмотра вложенных файлов.

При согласовании договоров используются задания видов:

Консолидация Договора

Предназначена для отправки заданий на консолидацию и формирование итоговых версий документов по итогам согласования и подписания.

Согласование с контрагентом

Предназначены для совершения операций с отдельным заданием по согласованию/консолидации/подписанию договорного документа в процессе его исполнения.

Работа по договору и дополнительному соглашению

Работа с договорами и дополнительными соглашениями предполагает выполнение пользователями действий, приводящих к изменению состояния карточки. При помощи команд бокового меню можно либо перевести карточку в другое состояние относительно текущего, либо вернуться в предыдущее состояние.

Для карточек договоров и дополнительных соглашений предусмотрен следующий набор доступных состояний:

Подготовка договорного документа:

1. Подготовка

   Создание новой карточки договорного документа и заполнение первичных регистрационных данных. Сохранение карточки с присвоением регистрационного номера и заполнением даты регистрации. См. Подготовка и регистрация договора.

Согласование и подписание договорного документа:

2. На согласовании.

   В карточке выполнена команда **Создать согласование**. Создано и запущено согласование по договорному документу. Согласующим отправлены задания на согласование. Всеми согласующими принято решение по согласованию документа внутри собственной организации. Инициатором согласования выполнена консолидация полученных от согласующих версий документа, создана итоговая версия документа.

3. На согласовании контрагентом.

   В карточке согласования консолидатором, подготовившим итоговую версию документа, выполнена команда **Отправить контрагенту**. Сформировано письмо контрагенту с приложенной версией итогового договорного документа.

4. Согласован.

От контрагента получен согласованный вариант документа.

5. На подписании.

   Полученный от контрагента согласованный вариант документа передан на подписание сотруднику, указанному в поле *Подписывает* собственной организации.

6. На подписании контрагентом.

   Контрагенту отправлен оригинал договорного документа с подписью представителя собственной организации.

7. Подписан.

   Оригиналы договора подписаны со стороны собственной организации и организации контрагента. В заданиях на подписание принято решение **Подписан**.

Проставление отметок о движении договорного документа:

8. Заключён.

   В карточке договорного документа выполнена команда **Заключить**. В карточке в поле *Договор заключен* проставлена дата заключения договора. См. Проставление отметки о заключении договора/дополнительного соглашения.

9. Завершен.

   В карточке договорного документа выполнена команда **Завершить** или истек срок действия договора. В карточке в поле *Дата завершения* проставлена дата завершения действия договора. См. Проставление отметки о завершении договора/дополнительного соглашения.

10. Расторгнут.

    В карточке договорного документа выполнена команда **Расторгнуть**. В карточке в поле *Дата завершения* проставлена дата завершения действия договора. См. Расторжение договора/дополнительного соглашения.

11. Аннулирован.

    В карточке договорного документа выполнена команда **Аннулировать**. См. Аннулирование договора/дополнительного соглашения.

Перевод карточки договорного документа в архив:

12. В архиве.

    В свойствах карточки установлен атрибут **Архивирована**.

Подготовка и регистрация договора

1. Создайте карточку вида **Документ** › **Документ УД** › **Договор**.

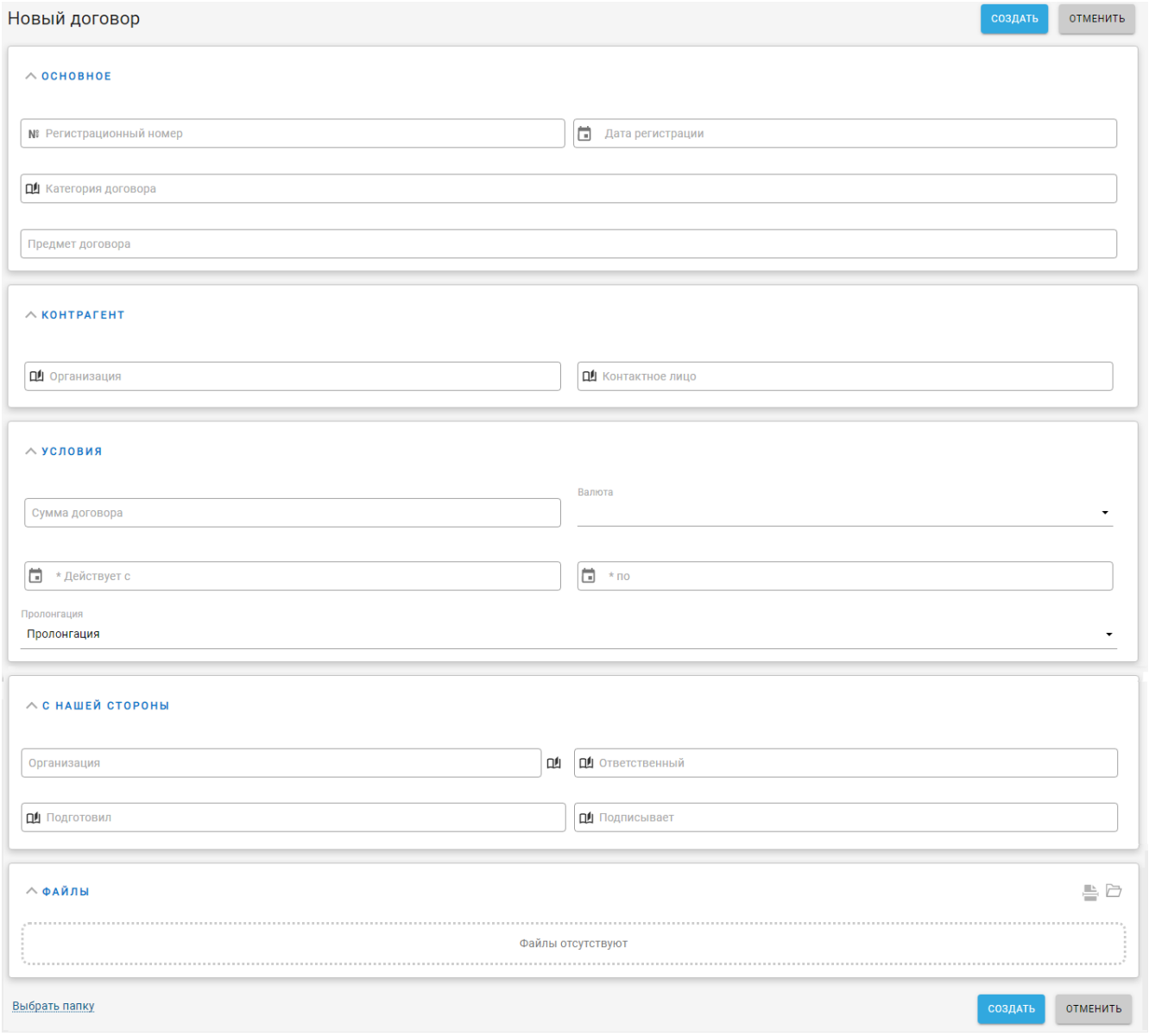


Рисунок 193. Карточка документа "Договор"

2. Заполните договор.

Основное

Регистрационный номер

Регистрационный номер договора (нажмите кнопку № для получения

номера). Если номер не присвоен, это будет сделано автоматически при сохранении карточки.

Дата регистрации

Дата регистрации договора. По умолчанию устанавливается текущая дата.

Категория договора

Категория к которой относится договор. Значение выбирается из узла "Группы договоров" Конструктора справочников.

Предмет договора

Описание предмета договора.

Контрагент

Контактное лицо

Ответственный сотрудник контрагента.

Организация

Контрагент, с которым заключён договор.

Условия

Сумма договора

Стоимость договора в цифровом выражении.

Валюта — валюта договора из списка

По умолчанию выбираются "рубли" (RUB).

Действует с, по

Период действия договора.

Пролонгация

Вариант автоматического продления договора из списка:

- ***Нет*** — периодического продления договора не предполагается. Заключённый договор может быть продлен вручную;
- ***Да*** — периодическое продление договора предполагается, но продление будет выполняться вручную;
- ***Автоматически на месяц*** — после окончания срока действия договор будет автоматически продлен на один месяц;

- ***Автоматически на квартал*** — после окончания срока действия договор будет автоматически продлен на один квартал;
- ***Автоматически на год*** — после окончания срока действия договор будет автоматически продлен на один год;
- ***Автоматически на 3 года*** — после окончания срока действия договор будет автоматически продлен на три года;
- ***Автоматически на 5 лет*** — после окончания срока действия договор будет автоматически продлен на пять лет.

По умолчанию значение не установлено.

С нашей стороны

Организация

организация или подразделение, ответственное за договор.

Ответственный

сотрудник, который отвечает за заключение и подписание договора с контрагентом. По умолчанию выбирается текущий сотрудник.

Подготовил

сотрудник, который регистрирует договор. По умолчанию выбирается текущий сотрудник.

Подписывает

сотрудник, который отвечает за подписание договора в организации. Данному сотруднику будет передан полученный от контрагента согласованный вариант договора.

Файлы

Файлы договора. По умолчанию к карточке договора прикладывается шаблон договора, настроенный администратором. Если требуется, удалите данный файл и приложите новый.

3. Нажмите кнопку **Создать**. После сохранения договор будет открыт в режиме просмотра.

Подготовка и регистрация дополнительного соглашения

Дополнительное соглашение может быть создано из карточки существующего договора или в виде отдельного документа.

1. Создайте дополнительное соглашение одним из следующих способов:
   - Создайте карточку вида **Документ › Документ УД › Дополнительное соглашение**.
   - Откройте карточку договора и нажмите кнопку **Создать дополнительное соглашение**. Договор не должен быть аннулирован, расторгнут или завершен.

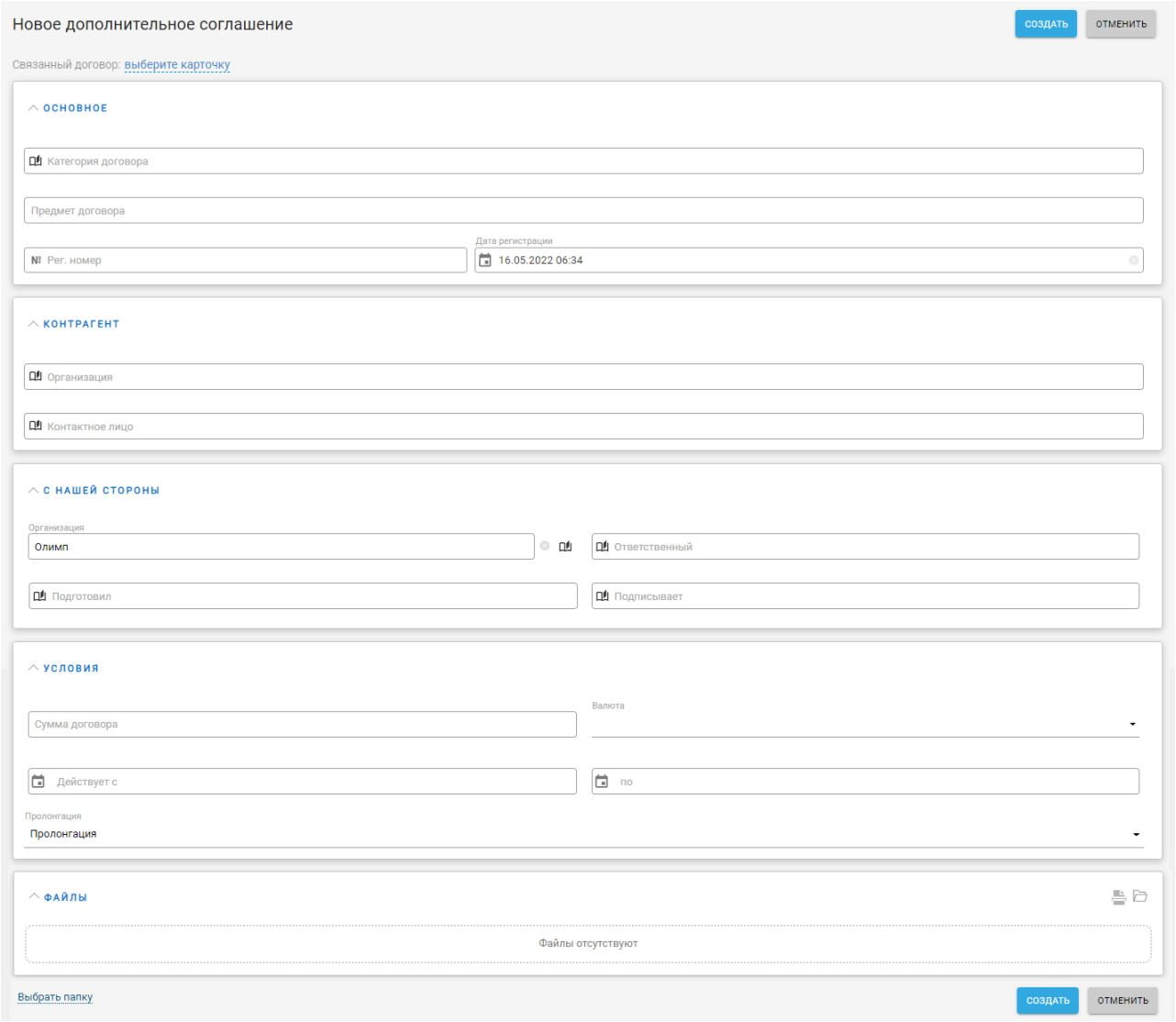


Рисунок 194. Карточка документа "Дополнительное соглашение"

2. Заполните дополнительное соглашение.
3. Нажмите кнопку **Создать**. После сохранения карточка будет открыта в режиме просмотра.

> Ссылка на дополнительное соглашение будет добавлена в договор, с

которым связано соглашение.

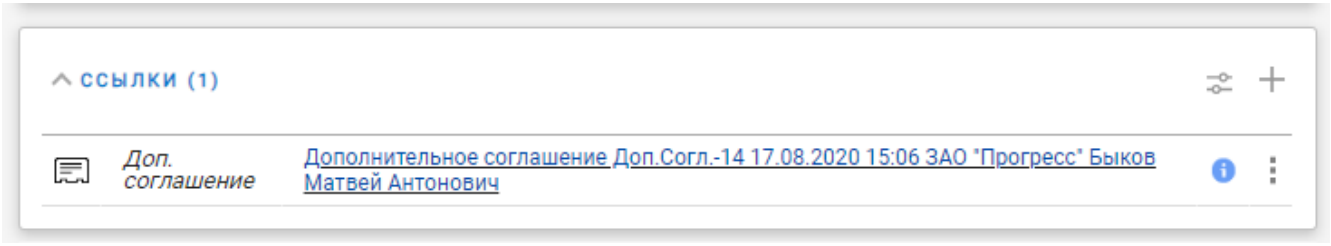


Рисунок 195. Ссылка на дополнительное соглашение в карточке договора

Перечень полей дополнительного соглашения

Связанный договор

Ссылка на карточку договора, для которого оформляется дополнительное соглашение.

После выбора договора предлагается загрузить данные договора в доп. соглашение: подтвердите или откажитесь.

Если дополнительное соглашение создаётся из договора, будет выбран "Связанный договор".

Основное

Категория договора

Категория к которой относится доп. соглашение. Значение выбирается из узла *Группы договоров* Конструктора справочников.

Предмет договора

Описание предмета доп. соглашения.

Рег. номер

Регистрационный номер доп. соглашения (нажмите кнопку № для получения номера). Если номер не присвоен, это будет сделано автоматически при сохранении карточки.

Дата регистрации

Дата регистрации доп. соглашения. По умолчанию устанавливается текущая дата.

Контрагент

Организация

Контрагент, с которым заключено доп. соглашение.

Контактное лицо

Ответственный сотрудник контрагента.

С нашей стороны

Организация

Организация/подразделение, ответственное за доп. соглашение.

Подготовил

Сотрудник, который регистрирует доп. соглашение. По умолчанию выбирается текущий сотрудник.

Ответственный

Сотрудник, который отвечает за заключение и подписание доп. соглашения с контрагентом. По умолчанию выбирается текущий сотрудник.

Подписывает

Сотрудник, который отвечает за подписание доп. соглашения в организации. Данному сотруднику будет передан полученный от контрагента согласованный вариант доп. соглашения.

Условия

Сумма договора

Стоимость договора в цифровом выражении.

Валюта

Валюта доп. соглашения из списка. По умолчанию выбираются рубли (RUB).

Действует с, по

Период действия доп. соглашения. По умолчанию устанавливается текущая дата.

Пролонгация

Вариант автоматического продления договора из списка:

- ***Нет*** — периодического продления договора не предполагается. Заключённый договор может быть продлен вручную;
- ***Да*** — периодическое продление договора предполагается, но продление будет выполняться вручную;
- ***Автоматически на месяц*** — после окончания срока действия договор будет автоматически продлен на один месяц;

- ***Автоматически на квартал*** — после окончания срока действия договор будет автоматически продлен на один квартал;
- ***Автоматически на год*** — после окончания срока действия договор будет автоматически продлен на один год;
- ***Автоматически на 3 года*** — после окончания срока действия договор будет автоматически продлен на три года;
- ***Автоматически на 5 лет*** — после окончания срока действия договор будет автоматически продлен на пять лет.

  По умолчанию значение не установлено.

Файлы

Файлы доп. соглашения. По умолчанию прикладывается шаблон доп. соглашения, настроенный администратором. Если требуется, удалите данный файл и приложите новый.

Согласование и подписание договорных документов

Web-клиент включает демонстрационный процесс согласования, по которому могут согласовываться и подписываться карточки *Договор* и *Дополнительное соглашение*.

Демонстрационный процесс согласования представляет собой выполнение описанного далее сценария работы с договором.

Согласование документа внутри организации

При запуске согласования карточка договорного документа переводится в состояние `На согласовании`.

Всем пользователям, состоящим в группе *Согласование Договоров*, рассылаются задания вида *На согласование*, к которым приложены согласуемые файлы.

Режим исполнения заданий — *Параллельно*, то есть задания рассылаются одновременно всем исполнителям и выполняются ими независимо друг от друга.

В процессе согласования документа исполнитель может ознакомиться с документами и высказать положительное или отрицательное решение посредством соответствующих кнопок бокового меню карточки задания *На согласование*.

Решение по согласованию принимается сразу для всех файлов, приложенных к

карточке задания.

Консолидация согласованного договора

Консолидация — создание итогового документа по итогам согласования внутри организации.

Когда все согласующие лица высказали своё мнение по документу, консолидатор (инициатор согласования) должен принять решение о дальнейшей судьбе договора.

1. Если принято решение о том, что договор в текущей редакции *не согласован*, консолидатор может:
   - Составить новую редакцию договора и отправить её на согласование.
   - Отменить согласование.
2. Если договор *согласован* или *согласован с замечаниями*, карточка договорного документа переводится в состояние `Согласован`. Консолидатор (после выбора итоговой на данном этапе версии документа) может:
   - Отправить договор на согласование с контрагентом.
   - Отправить договор на печать (если согласование с контрагентом было произведено ранее).
   - Отменить согласование.

Согласование и подписание документа с контрагентом

При переходе к этому этапу карточка договорного документа переводится в состояние `На согласовании Контрагентом`. Ответственному за договорной документ приходит задание вида *Согласование с контрагентом*, к которому приложена согласованная внутри организации версия договора.

Получив такое задание, сотрудник должен передать полученную версию договора на согласование в организацию контрагента, получить ответ контрагента и принять решение о дальнейшей судьбе договора.

1. Если принято решение о том, что договор в текущей редакции *не согласован*, ответственный может:

- Загрузить в карточку версию, полученную от контрагента и отправить её на согласование внутри организации.
- Отменить согласование.

2. Если договор *согласован* или *согласован с замечаниями*, ответственный может:
   - Отправить договор на печать.
   - Отправить договор на подписание внутри организации (если от контрагента вернулся подписанный оригинал).
   - Отменить согласование.



Этапы согласования договора внутри организации, консолидации и согласования с контрагентом могут циклически повторяться до выработки одобренного всеми согласующими варианта договора.

Печать бумажного экземпляра документа

Этап является техническим, предваряющим подписание бумажного оригинала договорного документа. При передаче на печать карточка договорного документа переводится в состояние "На подписании".

Печать договора осуществляет сотрудник из "Группы печати", получивший задание вида *Подписание*. После печати бумажный оригинал договорного документа передается на подписание сотрудникам, указанным в поле *Подписал* карточки договорного документа.

Подписание документа внутри организации

При передаче на подписание карточка договорного документа переводится (или остается) в состояние `На подписании`.

Всем пользователям, состоящим в группе `Подписано`, рассылаются задания вида *Подписание*, к которым приложены файлы для подписания.

Режим исполнения заданий — *Параллельно*, то есть задания рассылаются одновременно всем исполнителям и выполняются ими независимо друг от друга.

В процессе подписания исполнитель может ознакомиться с документами и высказать положительное или отрицательное решение посредством соответствующих кнопок бокового меню карточки задания *Подписание*.

Решение по подписанию принимается сразу для всех основных файлов, приложенных к карточке.



При согласовании договора или дополнительного соглашения с использованием демонстрационного процесса, заполнение полей *Подписано* и *Ответственный* в карточке *Договор* или *Дополнительное соглашение* является обязательным.

Консолидация подписанного договора

Консолидация — создание итогового документа по итогам подписания внутри организации и завершение маршрута.

Когда все подписанты высказали своё мнение по документу, консолидатор (инициатор согласования) должен принять решение о дальнейшей судьбе договора.

1. Если в подписании отказано, консолидатор может:
   - Составить новую редакцию договора и начать новый цикл согласования.
   - Отменить согласование.
2. Если договор *подписан*, консолидатору следует:
   - Завершить согласование, если ранее договор был согласован/подписан контрагентом.
   - Передать договор на согласование/подписание контрагенту, если ранее этого сделано не было.

Отправка договора на согласование

1. Откройте карточку договора в режиме чтения.

   Договор должен быть в состоянии `Подготовка`.

2. На панели операций нажмите кнопку **На согласование**. Будет открыта панель отправки на согласование.

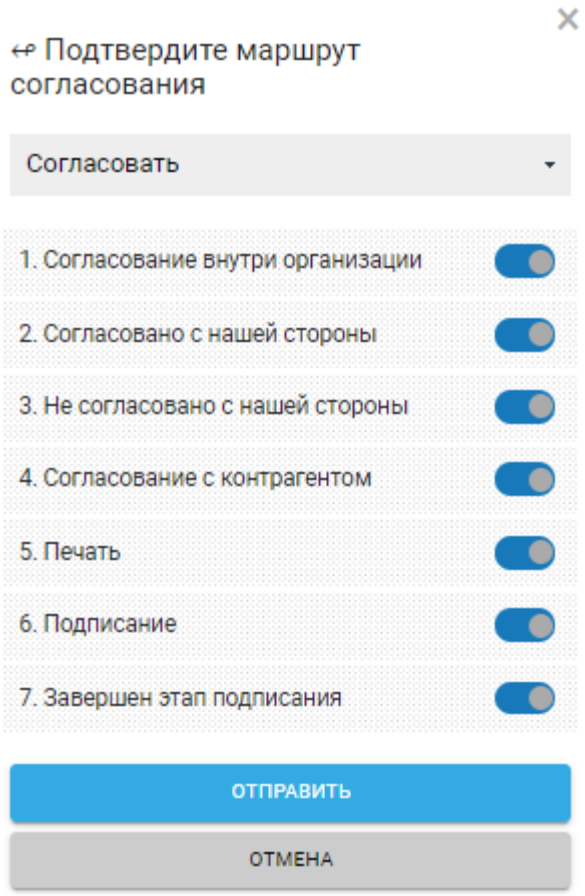


Рисунок 196. Панель отправки договора на согласование

3. Убедитесь, что в маршруте указаны участники.

Web-клиент содержит демонстрационный вариант согласования договоров:

- **Согласование внутри организации** — сотрудники из группы *Согласование договоров* в Справочнике сотрудников. Если согласующих нет, обратитесь к администратору.
- **Согласовано с нашей стороны** и **Не согласовано с нашей стороны** — сотрудник, указанный в карточке, в поле `Подготовил`.
- **Согласование с контрагентом** — сотрудник, указанный в карточке, в поле `Ответственный`.
- **Печать договоров** — сотрудники из группы *Печать договоров* в Справочнике сотрудников. Если согласующих нет, обратитесь к администратору.
- **Подписание** — сотрудник, указанный в карточке, в поле

Подписывает.

- **Завершён этап подписания** — сотрудник, указанный в карточке, в поле *Подготовил*.



Согласование может быть выполнено успешно, только если все участники заполнены правильно.

4. Нажмите кнопку **Отправить**. Состояние договора изменится на `На согласовании`. Для участников согласования будут сформированы задания на согласование.

Общая инструкция по отправке документов на согласование приведена в пункте Отправка документа на согласование.

Инструкция по управлению согласованием приведена в пункте Управление активным согласованием.

Ознакомиться с ходом согласования можно на странице *Ход согласования* карточки.

Согласование договора

1. Перейдите в папку **Мои задания** › **Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение** › **Входящие**

2. Откройте полученное задание с описанием *Согласование документов* в состоянии `Не начато`.
3. Выполните действия с согласуемыми документами:
   - Ознакомьтесь с составом согласуемых файлов.
   - Ознакомьтесь с договором: нажмите на ссылку в разделе **Ссылка на карточку** для перехода к договору.
   - Если требуется, добавьте собственную версию договора.
4. Чтобы принять решение по согласованию, нажмите кнопку:
   - **Согласовать** или **Согласовать с замечаниями** на панели операций. Когда все согласующие завершат свои задания, документ будет передан на консолидацию.
   - **Не согласовано**. Документ будет отправлен на новый цикл согласования.

- **Делегировать**, **В работу**.

Общая информация о принятии решения по согласованию приведена в пункте Задание на согласование.

Консолидация согласованного договора

Консолидация договора (принятие решения по результатам согласования) выполняется после завершения согласования договора.

В демонстрационном согласовании консолидацию выполняет сотрудник, подготовивший договор.

Согласование может быть завершено со статусом *согласовано* или *не согласовано*. В зависимости от результата будет начат один из двух демонстрационных вариантов консолидации: *Согласовано с нашей стороны* или *Не согласовано с нашей стороны*.

1. Перейдите в папку **Мои задания** › **Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение** › **Входящие**

2. Откройте полученное задание с описанием *Консолидация договора*
3. Ознакомитесь с результатными согласования и версиями согласующих.
4. Для принятия решения по согласованию на панели операций нажмите кнопку:

   Для согласованного договора:

   - **На подписание**. Передать договор на подписание в организации.
   - **На согласование с контрагентом**. Передать договор на согласование с контрагентом.
   - **Отменить**. Отменить согласование договора.

   Для несогласованного договора:

   - **Отмена** — отменить согласование договора.
   - **Повторить согласование** — запустить новый цикл согласования.

Согласование договора с контрагентом

Этап согласования договора с контрагентом выполняется, если при консолидации было принято решение **На согласование с контрагентом**.

Согласование документов с контрагентами выполняется вне системы Docsvision. Целью данного этапа является фиксация факта поступления в организацию согласованного экземпляра документа (например, документ может поступить по электронной почте или это может быть обычное письмо с документом, в котором проставлены подписи руководителей контрагента).

1. Перейдите в папку **Мои задания** › **Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение** › **Входящие**

2. Откройте полученное задание с описанием *Согласование с контрагентом* в состоянии `Не начато`.

   Задание содержит версию договора, которая должна быть согласована с контрагентом.

3. Если документ не был отправлен контрагенту письмом на предыдущем этапе, отправьте документ на согласование или подписание контрагенту способом, предусмотренным регламентом организации.
4. Получите согласованный документ от контрагента способом, предусмотренным регламентом организации.
5. Примите решение по дальнейшей работе с документом:
   - Ознакомьтесь с составом согласуемых файлов.
   - Ознакомьтесь с договором: нажмите на ссылку в разделе **Ссылка на карточку** для перехода к договору
   - Если требуется, добавьте собственную версию договора.
6. Для принятия решения по согласованию на панели операций нажмите кнопку:
   - **Согласован и подписан контрагентом**. Если контрагент прислал подписанный бумажный оригинал договора.
   - **Согласован контрагентом**. Если контрагент прислал согласованный файл договора.
   - **Повторить**. Начать новый цикл, если контрагент отказал в согласовании.
   - **Отмена**. Отменить согласование.

     Если документ был успешно согласован контрагентом, то согласно демонстрационному маршруту согласования будет выполнен переход

к этапу *Печать договора*. Соответствующее задание поступит сотрудникам из группы *Печать договоров* Справочника сотрудников.

Печать договора

В демонстрационном согласовании печать бумажного экземпляра договора осуществляет сотрудник, который включён в группу *Печать Договоров* в Справочнике сотрудников.

На печать договор поступает, если этап *Согласование с контрагентом* был завершён с результатом *Согласован контрагентом*.

1. Перейдите в папку **Мои задания › Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение › Входящие**

2. Откройте полученное задание с описанием *Подписание документов* в состоянии `Не начато`.
3. Распечатайте приложенные к карточке задания файлы.
4. Нажмите кнопку **Распечатано** на панели операций.

> После проставления отметки о печати, согласно демонстрационному маршруту согласования, будет выполнен переход к этапу *Подписание договора внутри организации*. Соответствующее задание поступит пользователю, указанному в поле `Подписывает` карточки договора.

Подписание договора

См. также "Подписание карточек типа "Документ"".

В демонстрационном согласовании договор подписывают сотрудники, указанные в поле *Подписывает* карточки Договор.

При получении задания на подписание подписант может ознакомиться с уже согласованными документами и высказать положительное или отрицательное решение.

1. Перейдите в папку **Мои задания › Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение › Входящие**

2. Откройте полученное задание с описанием *Подписание документов* в состоянии Не начато.
3. Чтобы принять решение по подписанию на панели операций нажмите кнопку:
   - **Подписано**. Договорной документ будет переведен в состояние Подписан.

     Ограничение на размер подписываемого файла в Docsvision отсутствует, однако при подписании файлов в Web-клиенте могут быть ограничения, связанные с плагином КриптоПро, см. подробнее в разделе "Подписание карточек типа "Документ"".

     Для формирования усиленной подписи с использованием сертификатов ЭП должны быть установлены компоненты КриптоПро.

   - **Отказано в подписании** — договорной документ будет переведен в состояние Подготовка.

     После проставления отметки о подписании, в соответствии с демонстрационным маршрутом согласования будет выполнен переход к этапу *Завершён этап подписания*.

     Соответствующее задание поступит пользователю, указанному в поле Подготовил карточки договора.

Консолидация подписанного договора

В демонстрационном согласовании в роли консолидатора подписанного документа выступает инициатор согласования — сотрудник, указанный в поле Подготовил согласуемой карточки договора или дополнительного соглашения.

Целью данного этапа является принятие решения о завершении согласования.

Консолидация подписанного документа отличается от консолидации согласованного документа только набором решений, которые может принять консолидатор.

1. Перейдите в папку **Мои задания** › **Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение** › **Входящие**

2. Откройте полученное задание с описанием *Консолидация договора* в состоянии Не начато.
3. Для принятия решения по подписанию на панели операций нажмите кнопку:
   - **Завершить маршрут**. Подписанный договор будет переведен в состояние Подписано, процесс согласования и подписания договора будет завершен.
   - **На согласование с контрагентом**. Договор будет передан на согласование с контрагентом.
   - **Повторить согласование**. Будет запущен новый цикл согласования договора.
   - **Отмена**. Карточка договора будет переведена в состояние Подготовка, чтобы начать новое или аннулировать согласование.

Передача договора или дополнительного соглашения на подписание контрагентом

Когда договор согласован с контрагентом и подписан внутри организации, бумажный экземпляр договорного документа отправляется для подписания контрагенту в соответствии с регламентом, принятым в организации. Чтобы сохранить эту информацию, после того, как договор отправлен, проставьте соответствующую отметку в карточке.

Чтобы передать карточку Договор или Дополнительное соглашение на подписание контрагентом:

1. Откройте карточку *Договор* или *Дополнительное соглашение* в состоянии На подписании или Подписан.
2. Нажмите кнопку **На подписание контрагентом** на панели операций.

   > Карточка *Договор* или *Дополнительное соглашение* будет переведена в состояние На подписании контрагентом.

3. Когда подписанный экземпляр договора вернётся, проставьте в карточке дату в поле *Дата подписания*.
4. Если получен оригинал договора, укажите дату получения в поле *Получен оригинал*.

Заключёние договора или дополнительного соглашения

Когда договор, подписанный контрагентом, возвращён, а в карточке указана соответствующая дата, договор считается заключённым. Факт заключения договора следует отметить в карточке.

Чтобы проставить отметку о заключении договора:

1. Откройте карточку Договора или Дополнительного соглашения в состоянии Подписан или На подписании контрагентом.

   В карточке должны быть заполнены поля Дата подписания и Дата согласования.

2. Если договор необходимо заключить на определённую дату, отличную от текущей, установите нужную дату в поле Договор заключён раздела *Общие даты*.

   Если дата не указана, при сохранении карточки будет установлена текущая дата.

3. Если в поле Получен оригинал не указана дата, укажите дату получения оригинала договора.
4. Нажмите кнопку **Заключён** на панели операций.

   Договор будет переведён в состояние Заключён.

Завершение договора или дополнительного соглашения

Если для договора не настроено продление, заключённый договор будет считаться завершённым по окончании срока его действия, указанного в карточке. При необходимости договор можно завершить вручную, проставив соответствующую отметку.

Чтобы завершить договор, выполните следующие действия:

1. Откройте карточку Договора или Дополнительного соглашения в состоянии Заключён.
2. Если договор необходимо завершить в определённую дату, отличную от текущей, установите её в поле Дата завершения.

   Если дата не указана, при сохранении карточки будет установлена текущая дата.

3. Нажмите кнопку **Завершить** на панели операций.

Договор будет переведён в состояние Завершен.

Расторжение договора или дополнительного соглашения

Расторгнуть можно уже заключённый договор или дополнительное соглашение.

Чтобы расторгнуть Договор или Дополнительное соглашение:

1. Откройте карточку Договора или Дополнительного соглашения в состоянии Заключён.
2. Нажмите кнопку **Расторгнуть** на панели операций.

Будет предложено указать дату и причину расторжения.

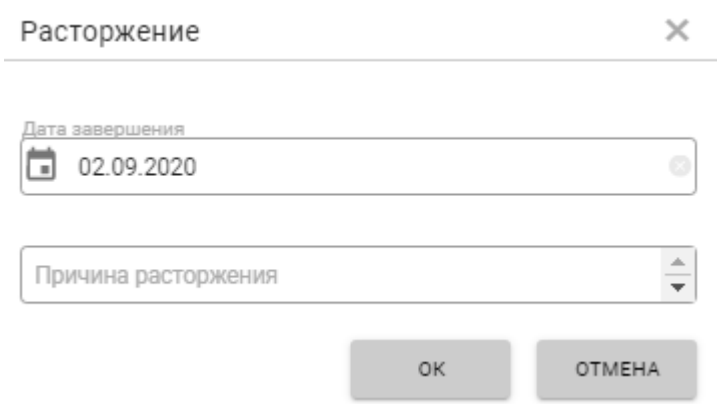


Рисунок 197. Причина расторжения договора

3. Введите комментарий о причине расторжения договора, укажите дату расторжения и нажмите кнопку **OK**.
4. Укажите дату и причину расторжения договора и нажмите кнопку **OK**.

Договор или Дополнительное соглашение будут переведены в состояние Расторгнут.

В поле Дата завершения раздела *Общие даты* карточки фиксируется дата расторжения (завершения) Договора. Причина расторжения будет добавлена в комментарии карточки.

Аннулирование договора или дополнительного соглашения

Аннулировать можно договор и дополнительное соглашение в состоянии

Подготавливается, На подписании контрагентом.

Чтобы аннулировать договорной документ:

1. Откройте карточку договорного документа.
2. Нажмите кнопку **Аннулировать** на панели операций.

> Договор будет переведен в состояние Аннулирован.

Продление срока договора

Продление срока договора может быть выполнено вручную, либо автоматически.

Договора будет продлён в следующих случаях:

- Если в результате регулярного мониторинга карточек вида *Договор* будет найдена карточка с истекшим сроком действия. Договор будет продлён автоматически в соответствии со значением поля Периодическое продление
- После обнаружения в системе связанной (вручную) с договором карточки вида *Доп. соглашение о продлении сроков* в состоянии Заключёно. Срок договора будет изменён на срок, указанный в дополнительном соглашении.

Автоматическое продление

Чтобы автоматически продлевать договор, установите одно из следующих значений в поле *Периодическое продление* карточки Договор:

- Автоматически на месяц
- Автоматически на квартал
- Автоматически на год
- Автоматически на 3 года
- Автоматически на 5 лет

Ручное продление

Чтобы вручную продлить договор, в поле *Периодическое продление* карточки Договор, установите значение: ***Да***.

Продлить срок договора вручную может сотрудник, который подготовил данный договор или является ответственным по нему. Продление допустимо для карточек в состоянии Заключён.

Чтобы вручную продлить срок договора, выполните следующие действия:

1. Откройте карточку договора.
2. Создайте дополнительное соглашение о продлении сроков к договору.
   a. Нажмите кнопку **Доп. соглашение о продлении сроков** на панели операций карточки Договор.

      Будет открыта карточка нового доп. соглашения.

   b. Заполните доп. соглашение, указав в нём новые сроки действия договора.

      Инструкция по созданию и заполнению доп. соглашения приведена в пункте Подготовка и регистрация дополнительного соглашения.

   c. Сохраните доп. соглашение.
   d. Проставьте отметку о заключении дополнительного соглашения.

      См. пункт Проставление отметки о заключении договора/дополнительного соглашения.

3. Зафиксируйте факт продления срока действия в карточке Договора.
   a. Нажмите кнопку **Продление срока договора** в разделе *Условия* карточки Договор. Будет открыто окно *Продление*.

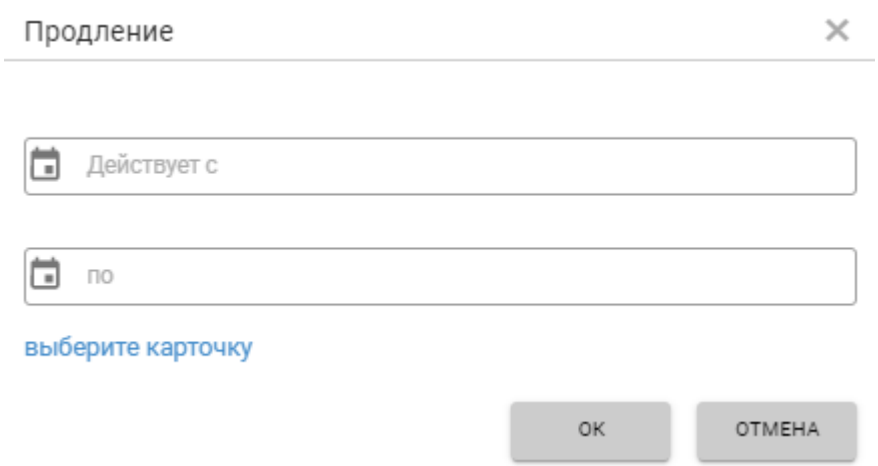


Рисунок 198. Продление договора

   b. Выберите созданное доп. соглашение о продлении сроков.

Даты действия в окне *Продление* будут установлены в значения из дополнительного соглашения.

c. Нажмите кнопку **ОК**.

Даты действия договора будут обновлены.

Работа с актами

Карточка *Акт* предназначена для хранения сведений об акте по договору. *Акт* не предполагает согласования и подписания в системе Docsvision, работа с ним заключается в проставлении отметок о движении бумажного документа, приводящих к изменению состояния карточки:

- Передаче на подписание внутри организации.
- Подписание внутри организации.
- Передаче на подписание контрагенту.
- Перевод в состояние Действителен.
- Аннулирование (при возникновении такой необходимости).

При помощи команд ленты можно либо перевести карточку в другое состояние относительно текущего, либо вернуть в предыдущее состояние.

Для карточек актов предусмотрен следующий набор доступных состояний:

Подготовка акта

Подготовка

Создание новой карточки акта и заполнение первичных регистрационных данных. Сохранение карточки с присвоением регистрационного номера и заполнением даты регистрации. См. Подготовка и регистрация акта.

Проставление отметок о движении документа

На подписании

В карточке договорного документа выполнена команда **На подписание контрагентом**. См. Проставление отметки о передаче на подписание.

Подписан

В карточке договорного документа выполнена команда **Подписан**. В карточке

в поле *Дата подписания контрагентом* проставлена дата. См. Проставление отметки о подписании.

На подписание контрагенту

В карточке договорного документа выполнена команда **На подписание контрагенту**. В карточке в поле *Дата подписания контрагентом* проставлена дата. См. Проставление отметки о передаче на подписание контрагентом.

Действителен

В карточке договорного документа выполнена команда **Назначить действительным**. См. Проставление отметки о действительности акта

Аннулирован

В карточке договорного документа выполнена команда **Аннулировать**. См. Аннулирование акта.

Подготовка и регистрация акта

Акт может быть создан из карточки существующего договора или в виде отдельного документа.

1. Создайте акт одним из следующих способов:
   - Создайте карточку вида **Документ** › **Документ УД** › **Акт**.
   - Откройте карточку договора и нажмите кнопку **Создать акт**. Договор не должен быть аннулирован, расторгнут или завершен.

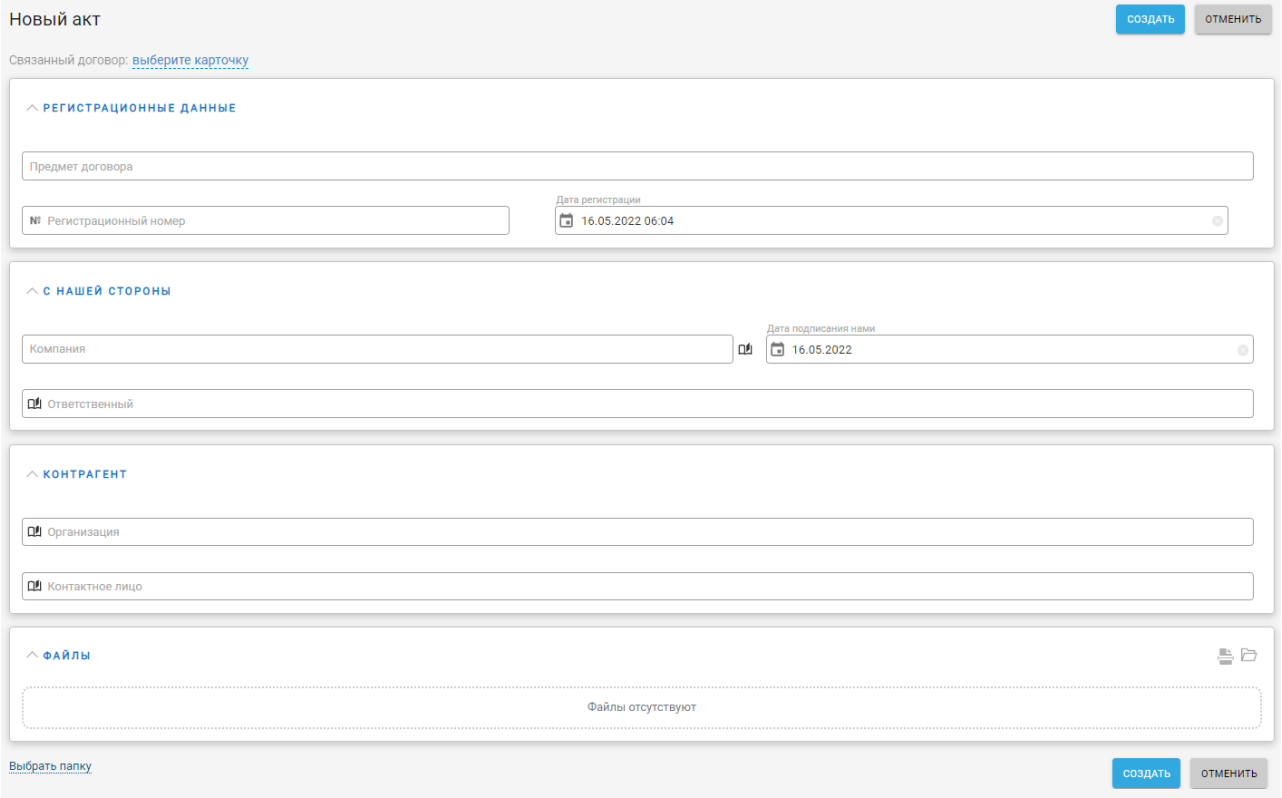


Рисунок 199. Карточка документа "Акт"

2. Нажмите кнопку **Создать**.

После сохранения карточка будет открыта в режиме просмотра.

Ссылка на акт будет добавлена в договор, с которым связан акт.

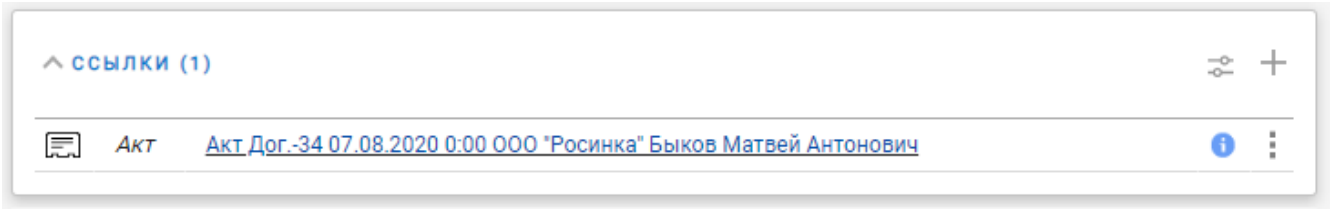


Рисунок 200. Ссылка на акт в карточке договора

Перечень полей карточки акта

Связанный договор

Ссылка на карточку договора, для которого оформляется акт.

После выбора договора будет предложено загрузить данные договора в акт: подтвердите или откажитесь.

Если акт создаётся из договора, Связанный договор будет выбран.

Регистрационные данные

Регистрационный номер

Регистрационный номер акта (нажмите кнопку №, чтобы получить номер). Если номер не присвоен, это будет сделано автоматически при сохранении карточки.

Дата регистрации

Дата регистрации акта. По умолчанию устанавливается текущая дата.

С нашей стороны

Компания

Организация или подразделение, ответственное за акт.

Дата подписания нами

Дата подписания акта, если он подписан (по умолчанию устанавливается текущая дата).

Ответственный

Сотрудник, который отвечает за подписание акта с контрагентом. По умолчанию выбирается текущий сотрудник.

Контрагент

Организация

Контрагент, с которым заключён договор.

Контактное лицо

Ответственный сотрудник контрагента.

Файлы

Файлы акта.

Отметка о передаче акта на подписание

1. Откройте акт в состоянии `Подготовка` или `На подписании контрагентом`.
2. Нажмите кнопку **На подписание** на панели операций.

> Карточка будет переведена в состояние `На подписании`.

Отметка о подписании акта

После подписания акта внутри организации необходимо проставить в карточке отметку о факте подписания.

1. Откройте акт в состоянии `На подписании`.
2. В поле *Дата подписания нами* укажите дату подписания, если не указана.
3. Нажмите кнопку **Подписан** на панели операций.

> Карточка *Акт* будет переведена в состояние `Подписан`.

Отметка о передаче акта на подписание контрагентом

Чтобы передать акт на подписание контрагентом:

1. Откройте акт в состоянии `Подготовка`, `На подписании` или `Подписан`.
2. Нажмите кнопку **На подписание контрагентом** на панели операций.

> Карточка *Акт* будет переведена в состояние `На подписании контрагентом`.
>
> После фактического подписания акта контрагентом, проставьте отметку о подписании по инструкции Проставление отметки о подписании.
>
> Перед установкой отметки укажите дату подписания контрагентом в поле *Дата подписания контрагентом* карточки.

Отметка о действительности акта

Для назначения *Акта* действительным, в карточке должны быть заполнены поля `Дата подписания нами` и `Дата подписания контрагентом`.

Чтобы проставить в карточке Акт отметку о его действительности:

1. Откройте карточку акта в состоянии `На подписании контрагентом` или `Подписан`.
2. Нажмите кнопку **Действителен** на панели операций.

> Карточка *Акт* будет переведена в состояние `Действителен`.

Возврат акта на подготовку

1. Откройте карточку акта в любом состоянии, кроме `Подготовка` или `Аннулирован`.
2. Нажмите кнопку **На подготовку** на панели операций.

   Карточка *Акт* будет переведена в состояние `На подготовке`.

Аннулирование акта

1. Откройте акт в состоянии `Подготавливается`, `На подписании`, `Подписан`.
2. Нажмите кнопку **Аннулировать** на панели операций.

   Карточка *Акт* будет переведена в состояние `Аннулирован`.

Подготовка отчётов по договорным документам

Для формирования отчётов по договорным документам пользователь должен быть включён в группу *Чтение всех договоров* в Справочнике сотрудников.

Пользователям доступны два отчёта:

- Договоры и ДС без подписанного оригинала;
- Договоры с истекающим сроком окончания.

Договоры и ДС без подписанного оригинала

Данный отчёт предназначен для поиска карточек договоров и дополнительных соглашений в состояниях `Подписан` и `Заключён`, для которых отсутствует подписанный оригинал (в карточке не заполнено поле *Получен оригинал*).

Чтобы подготовить отчёт, выполните следующие действия:

1. Откройте папку **Договоры › Отчёты › Договоры и ДС без подписанного оригинала**.

   Будет открыто окно для ввода параметров отчёта.

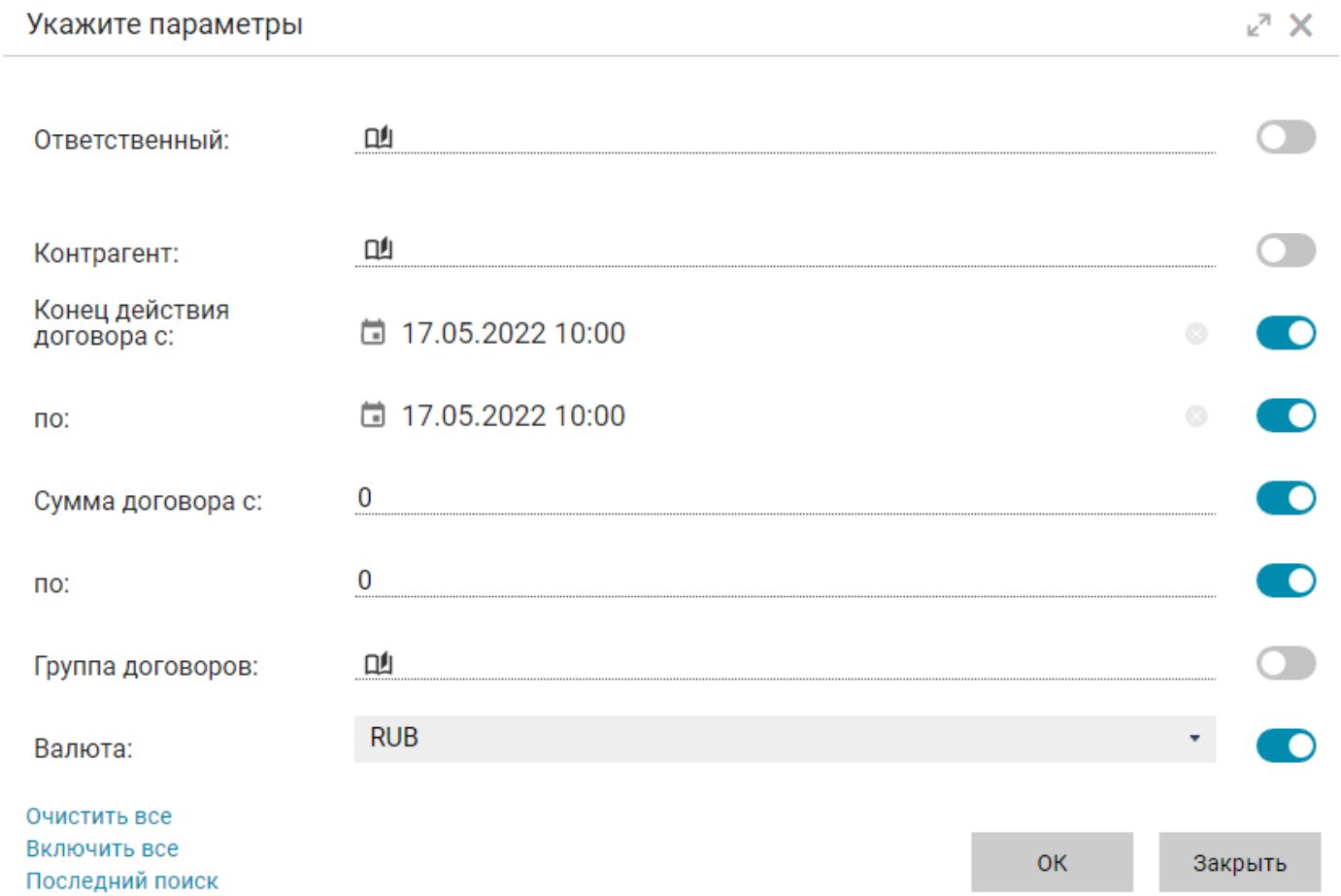


Рисунок 201. Параметры отчёта "Договоры и ДС без подписанного оригинала"

1. Укажите параметры отчёта:
   - *Ответственный*. Сотрудник, ответственный за договор.
   - *Контрагент*. Организации контрагента, с которой заключён договор.
   - *Конец действия договора с… по*. Даты начала и окончания действия договора.
   - *Сумма договора с … по*. Диапазон суммы договора.
   - *Группа договоров*. Группа, в которую включён договор.
   - *Валюта*. Валюта договора.

   Отключите переключатели в параметрах отчёта, которые не должны учитываться при поиске договоров.

2. Нажмите **ОК**. Будут показаны заключённые договора и доп. соглашения без подписанного оригинала, соответствующие установленным критериям.

Договоры с истекающим сроком окончания

Данный отчёт предназначен для поиска карточек договоров в состояниях Подписан и Заключён, у которых заканчивается срок действия.

Чтобы подготовить отчёт:

1. Откройте папку **Договоры** › **Отчёты** › **Договоры с истекающим сроком окончания**.

   Будет открыто окно для ввода параметров отчёта.

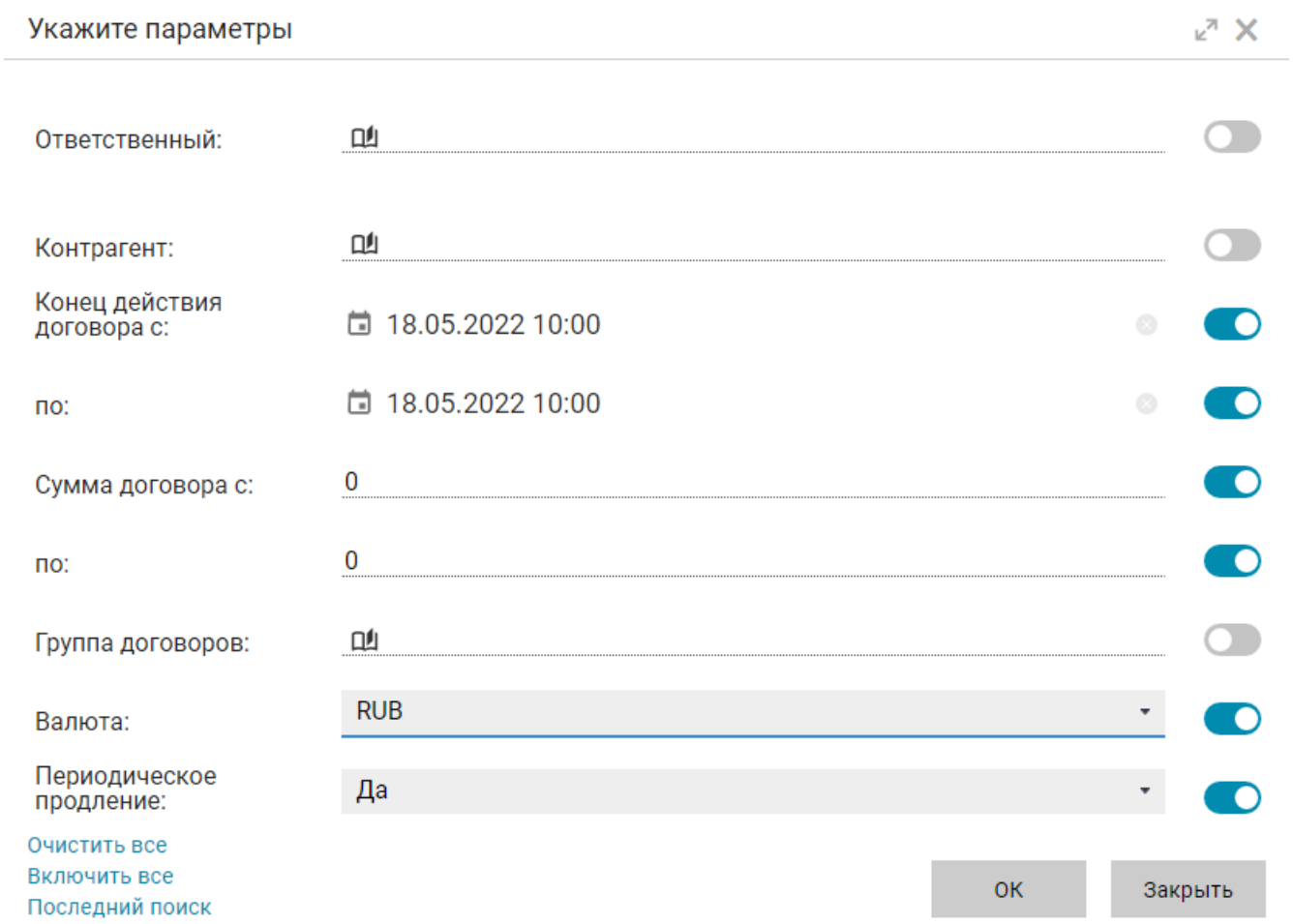


Рисунок 202. Параметры отчёта "Договоры с истекающим сроком окончания"

2. Укажите параметры отчёта:
   - *Ответственный*. Сотрудник, ответственный за договор.
   - *Контрагент*. Организации контрагента, с которой заключён договор.
   - *Конец действия договора с… по*. Даты начала и окончания действия договора.

- *Сумма договора с … по*. Диапазон суммы договора.
- *Группа договоров*. Группа, в которую включён договор.
- *Валюта*. Валюта договора.
- *Периодическое продление*. Способ продления договора.



Отключите переключатели  в параметрах отчёта, которые не должны учитываться при поиске договоров.

3. Нажмите **ОК**.

Будут показаны заключённые договора и доп. соглашения с истекающим сроком окончания.

Работа с заданиями

Карточки заданий предназначены для формирования поручений на выполнение каких-либо действий. Это могут быть поручения, связанные с исполнением определённой задачи или ознакомлением с документом. Указанные задачи выполняются с помощью карточек видов:

- *На исполнение*. Предназначено для раздачи исполнителям поручений, связанных с приложенным документом.

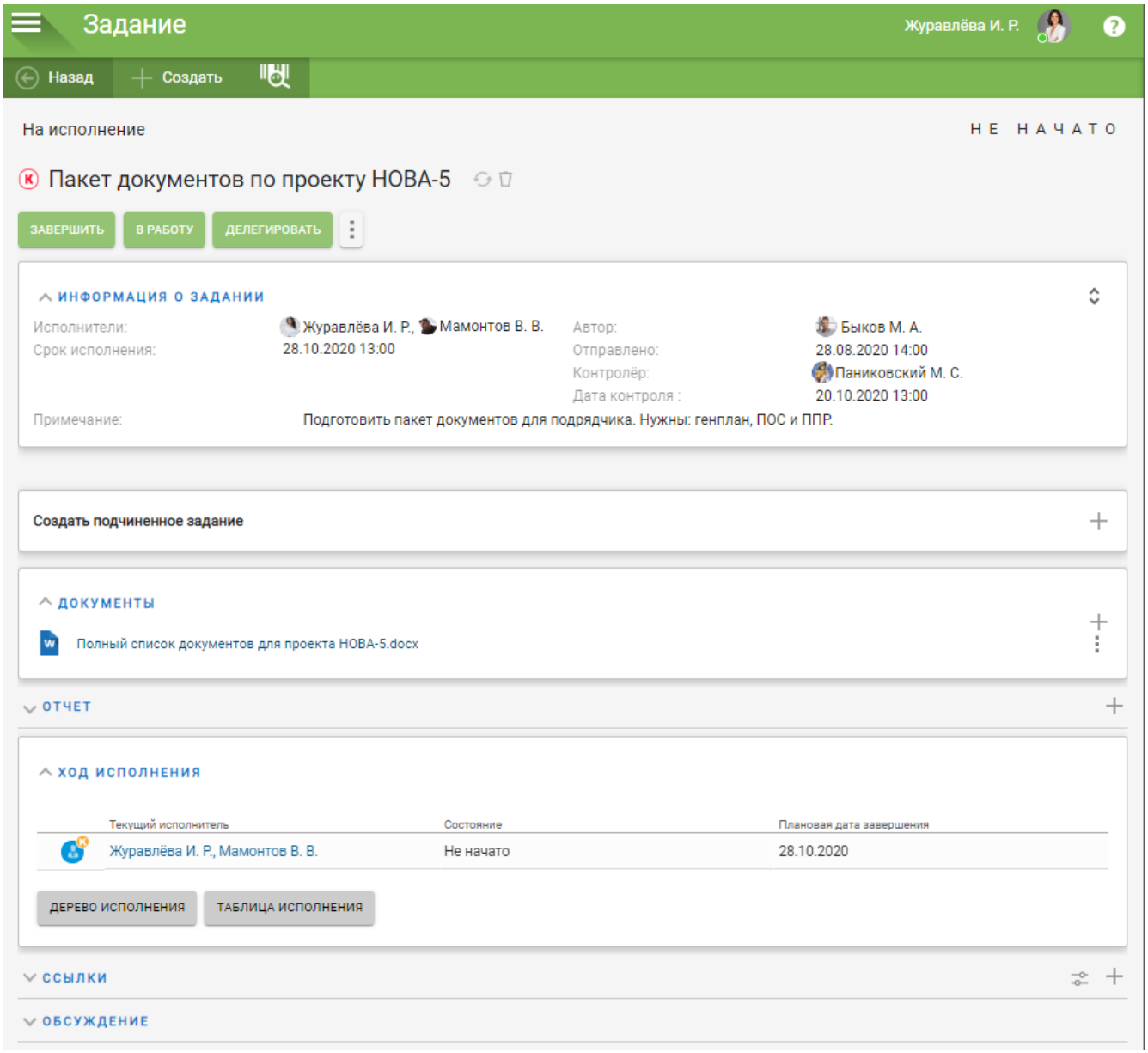


Рисунок 203. Отправленное задание "На исполнение", открытое у исполнителя

- *На ознакомление*. Предназначено для просмотра приложенного к карточке файла или карточки Документ УД.

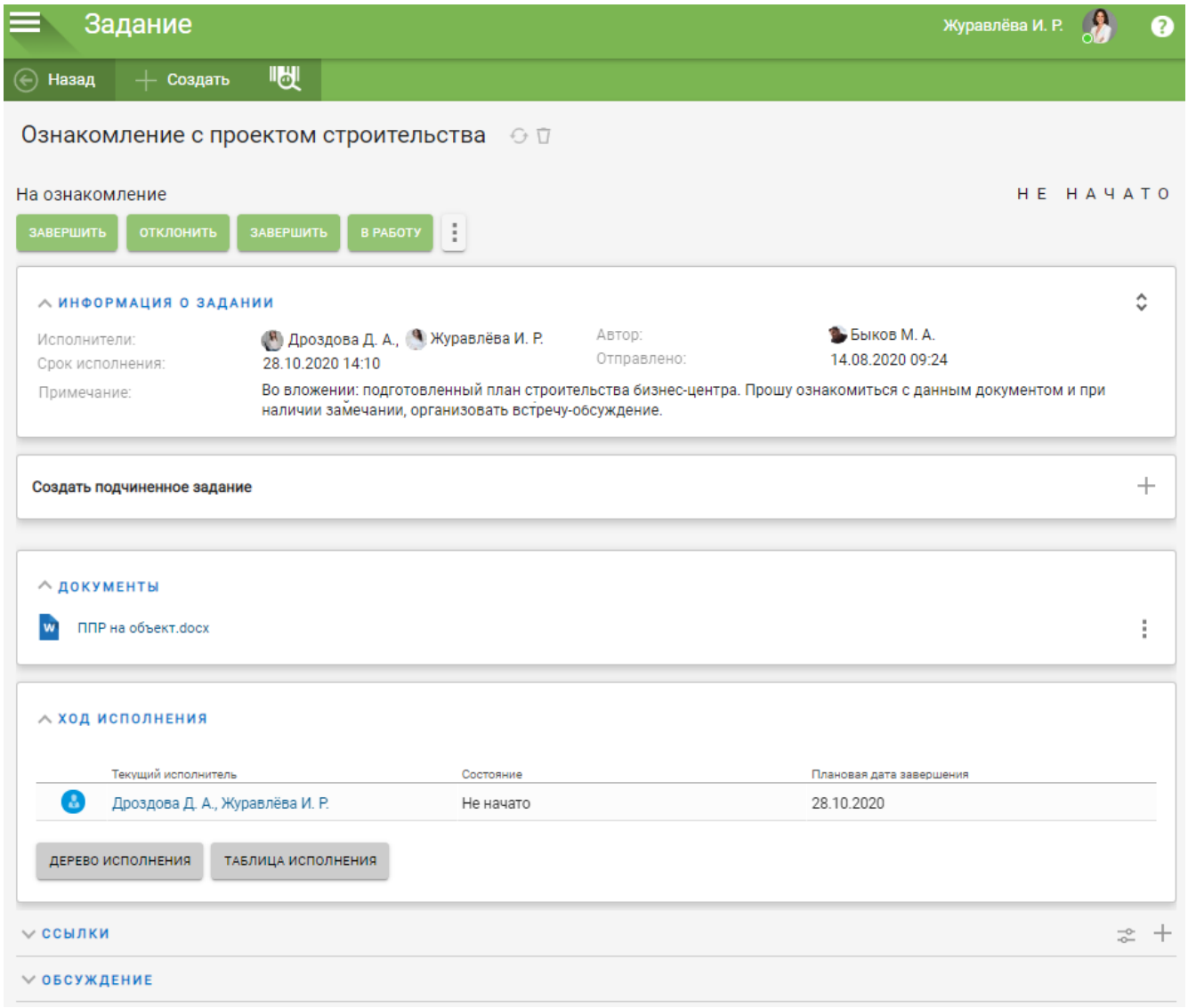


Рисунок 204. Отправленное задание "На ознакомление", открытое у исполнителя

Различные сценарии работы с заданиями рассмотрены далее.

Работа с заданиями, которые формируются при согласовании документов: (виды "На согласование", "На подписание" и "На консолидацию" рассматривается в соответствующих подразделах раздела Согласование документов.

Создание задания

1. Создайте карточку вида **Задание** › **Вид задания**. Будет открыто окно новой карточки.

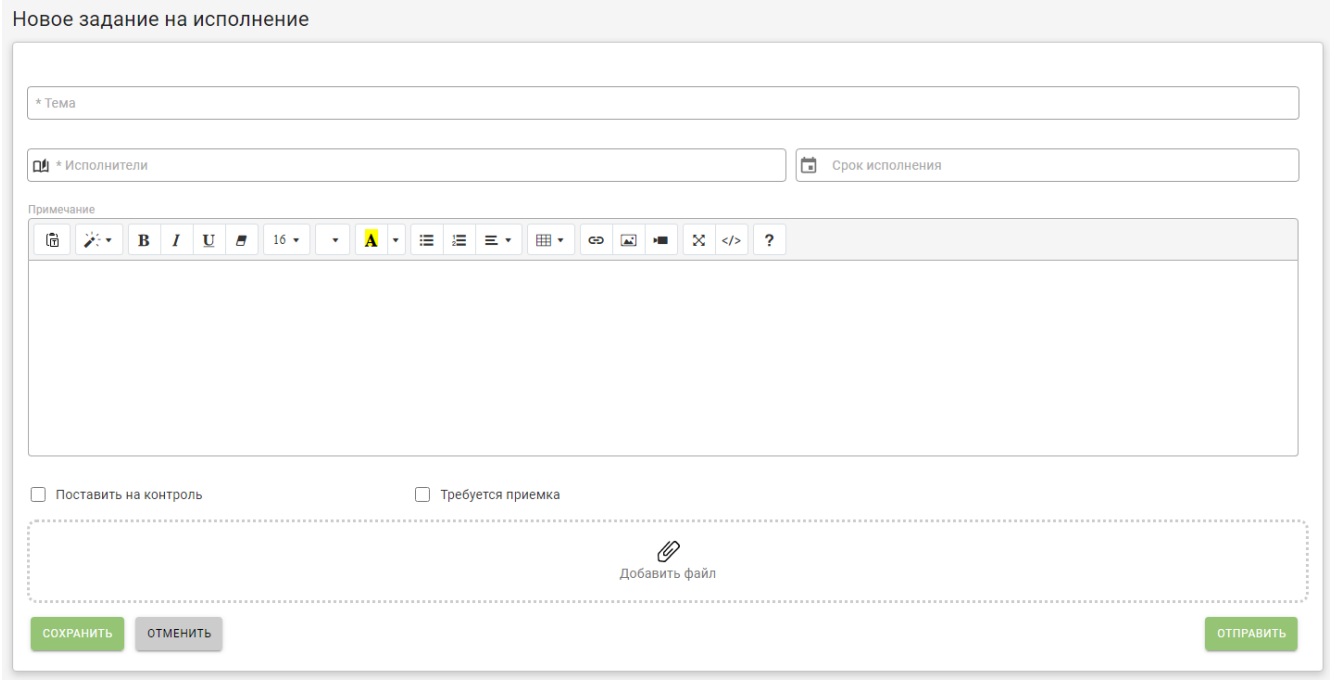


Рисунок 205. Новое задание вида "На исполнение"



Если карточка создана по шаблону, часть полей будет заполнена. Файлы из шаблона не переносятся.

2. Заполните поля задания.

   Список полей карточки зависит от вида задания. Перечень полей приведён в пунктах:

   - Перечень полей задания на исполнение,
   - Перечень полей задания на ознакомление.

3. Нажмите кнопку **Отправить** для немедленной передачи задания исполнителям или **Сохранить** для сохранения задания с возможностью отправки. Карточка будет открыта в режиме просмотра.

   Если была нажата кнопка **Отправить**, задание будет передано исполнителям.

   Ссылка на отправленное задание у Автора будет помещена в папку **Задания › Исходящие**, у Исполнителей — в папку **Задания › Входящие**.

Перечень полей задания на исполнение

Тема

Тема задания.

Обязательно для заполнения.

Исполнители

Исполнители задания.

Обязательно для заполнения.

Задание будет отправлено всем выбранным сотрудникам и сотрудникам выбранных организаций, подразделений, групп и ролей. Фактическим исполнителем станет первый сотрудник, взявший задание в работу.



Если нужно отправить каждому сотруднику собственное задание, используйте Группу заданий.

Срок исполнения

Плановый срок исполнения задания выбранными исполнителями. По умолчанию устанавливаются текущие дата и время.

Примечание

Дополнительная информация по заданию.

Поставить на контроль

Включает функции контроля исполнения задания ответственным сотрудником. При включённом контроле станут доступны параметры контроля:

Контролёр

Сотрудник, который контролирует выполнение задания.

Дата контроля

Дата, когда должен быть выполнен контроль выполнения задания.

Требуется приёмка

Включает функцию приёмки задания контролёром. После выполнения задания исполнителем, контролёр может принять результат выполнения или отказать. При включении приёмки будет автоматически включается контроль.

Добавить файл

Файлы задания.

Перечень полей задания на ознакомление

Тема

Тема задания.

Обязательно для заполнения.

Исполнители

Исполнители задания.

Обязательно для заполнения.

Задание будет отправлено всем выбранным сотрудникам и сотрудникам выбранных организаций, подразделений, групп и ролей. Фактическим исполнителем станет первый сотрудник, взявший задание в работу.



Если нужно отправить каждому сотруднику собственное задание, создайте их с помощью Группы заданий.

Срок ознакомления

Плановый срок ознакомления исполнителей с приложенным к заданию документом. По умолчанию устанавливаются текущие дата и время.

Примечание

Дополнительная информация по заданию.

Добавить файл

Файлы задания.

Изменение задания перед отправкой

Автор задания может изменить его, но только до отправки исполнителям:

1. Откройте карточку задания для редактирования.
2. Внесите требуемые изменения.
3. Нажмите кнопку **Сохранить**, чтобы сохранить изменения. Карточка будет открыта в режиме просмотра.

Отправить задание исполнителям можно из карточки, открытой для просмотра.

Отправка заданий исполнителям и мониторинг исполнения

Работа в Web-клиенте

Отправка заданий исполнителям и мониторинг исполнения

Описание процесса отправления, отслеживания исполнения и дальнейших действий с заданиями.

- Отправка задания на исполнение
- Отслеживание выполнения задания
- Отзыв задания от исполнителя
- Больше информации в навигационном меню слева…

Отправка задания на исполнение

Чтобы отправить сохранённое задание:

1. Откройте задание в состоянии "Подготовка".
2. На панели операций нажмите кнопку **Отправить**.

> Задание будет отправлено всем сотрудникам и сотрудникам выбранных организаций, подразделений, групп и ролей, указанным в поле *Исполнители*. Фактическим исполнителем станет первый сотрудник, взявший задание в работу. Состояние карточки изменится на `Не начато`.
>
> У Автора ссылка на задание будет помещена в папку **Задания › Исходящие**, у Исполнителей — в папку **Задания › Входящие**.

Отслеживание выполнения задания

Автор может отслеживать исполнение заданий в папке **Мои задания › Исходящие**, в которой отображаются отправленные задания.

Заместителю Автора нужно использовать папку **Замещение › Исходящие**.

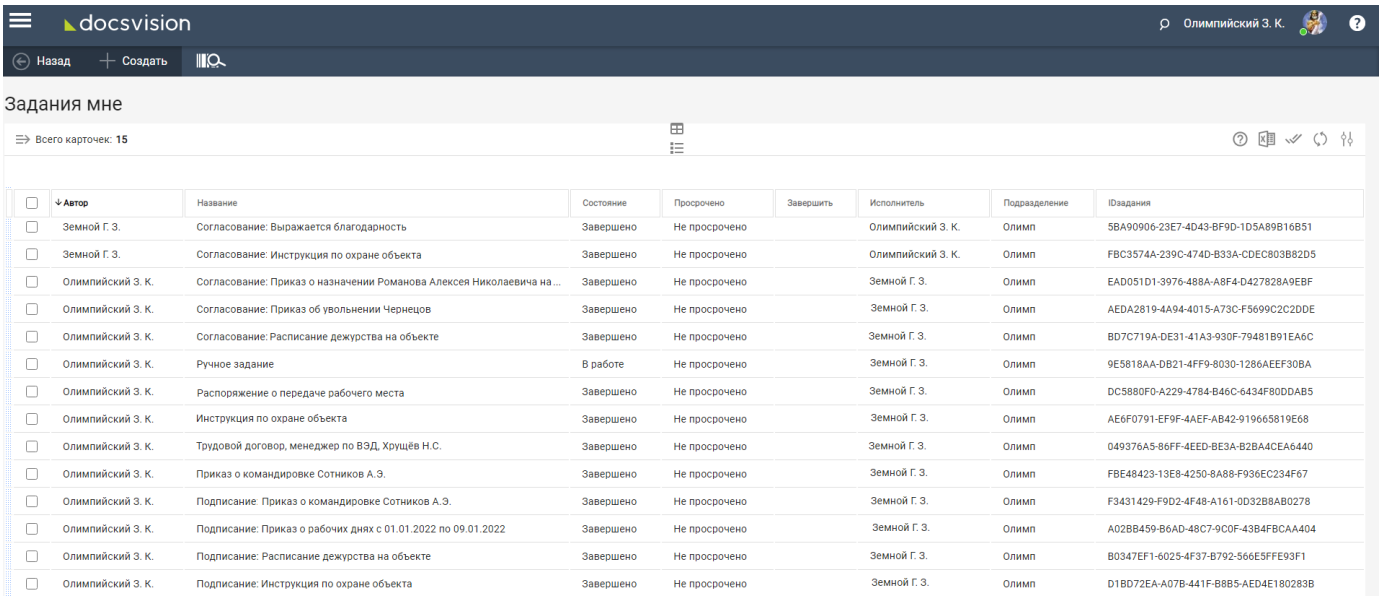


Рисунок 206. Папка **Мои задания › Исходящие**

Текущее состояние задания показано в колонке Состояние.

При изменении задания (включая смену состояния) карточка будет выделена жирным шрифтом.

завершённые задания перемещаются в папку **Мои задания › Исходящие › Завершено**.

Если задание было делегировано другому сотруднику, оно переместится в папку **Мои задания › Исходящие › Делегировано**.

Отзыв задания от исполнителя

Автор может отозвать задание, если оно не было завершено и не находится на приёмке.

1. Откройте задание.
2. На панели операций нажмите кнопку **Отозвать**.

> Состояние задания изменится на Отозвано и пропадет из папок Исполнителей. Все изменения в карточке, за исключением решения исполнителя, сохранятся.
>
> Если настройками задания предусмотрен автоматический отзыв подчиненных заданий или групп задания, то они также будут отозваны.

Завершение задания автором

Автор задания может завершить его не дожидаясь решения Исполнителя. Завершить задание можно в любом состоянии, кроме Завершено и Подготовка.

Чтобы завершить задание, выполните указанные в пункте Выполнение и завершение задания исполнителем действия.

Исполнение и делегирование заданий

Исполнитель — пользователь, который будет выполнять работы по заданию. *Исполнитель* обладает следующим набором прав на выполнение операций с заданием:

- Открытие и просмотр карточки.
- Выполнение операций со связанными документами.
- Операции со связанными заданиями и документами.
- Управление комментариями.

Исполнитель может изменять состояние карточки:

- Получить задание от автора.
- Выполнить и завершить задание самостоятельно.
- Отказаться от выполнения задания.
- Делегировать задание.

Получение задания исполнителем

Чтобы приступить к работе с входящим заданием, выполните следующие действия:

1. Перейдите в папку **Мои задания › Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение › Входящие**

2. Откройте полученную карточку *Задание*.

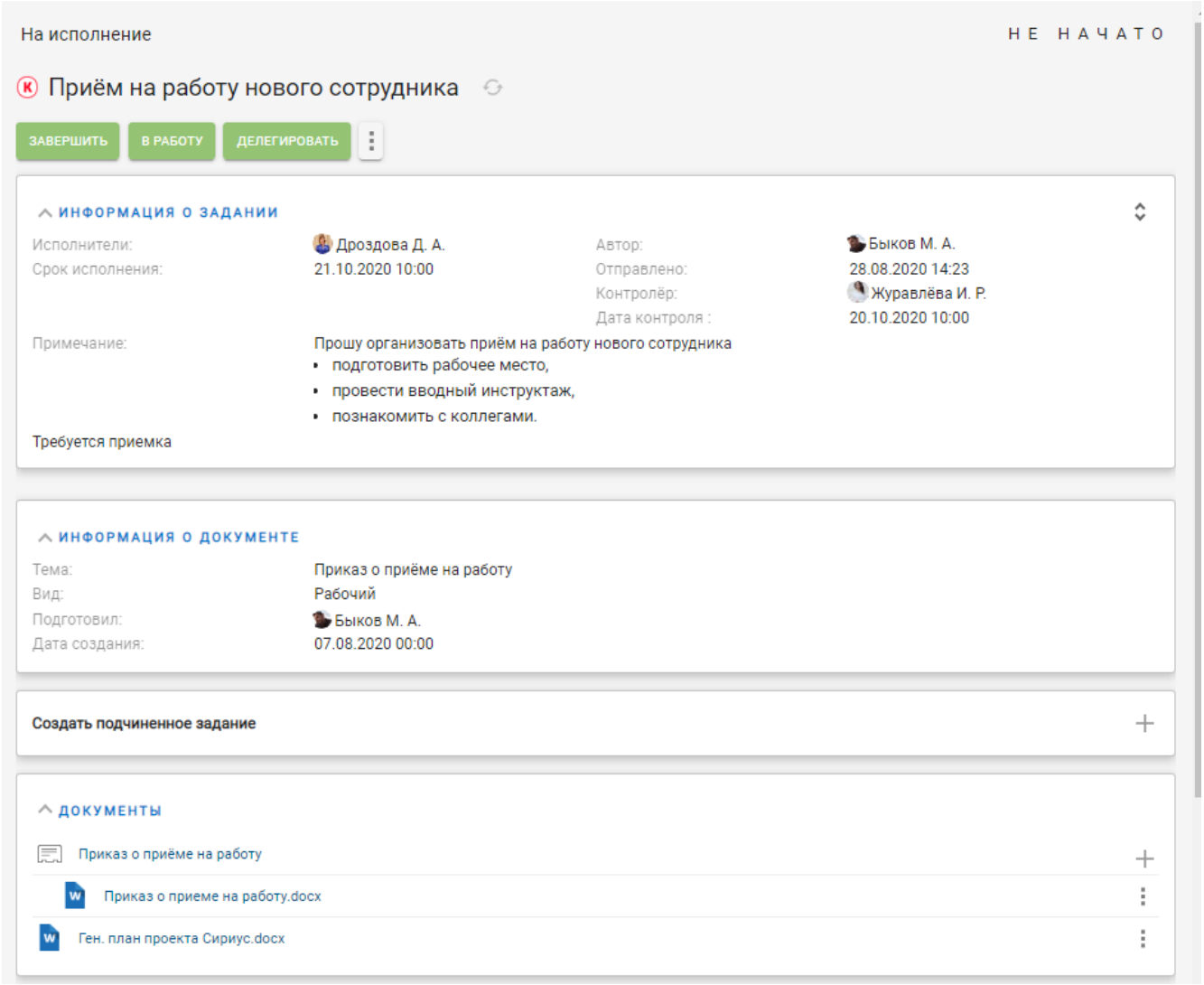


Рисунок 207. Задания в состоянии "Не начато" у исполнителя

3. Выполните необходимые действия с заданием:
   - выполните и завершите задание.
   - делегируйте задание другому сотруднику.
   - отклоните задание.

Выполнение и завершение задания исполнителем

После выполнения работ, предусмотренных заданием, завершите его:

1. Откройте карточку задания.
2. На панели операций нажмите кнопку **Завершить**.

   Чтобы завершить задание, в зависимости от настроек может потребоваться:
   - Ввести текстовый отчёт:

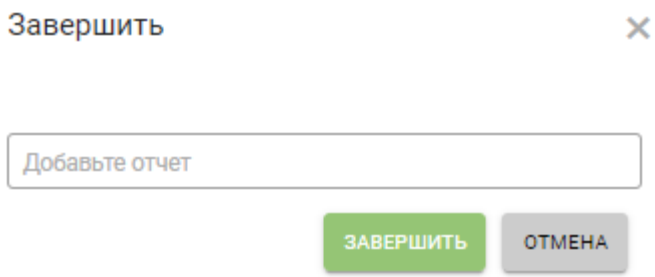


Рисунок 208. Ввод отчёта об исполнении задания

На данном этапе также могут быть указаны дополнительные параметры завершения, если это предусмотрено настройками задания.

- Приложить файл отчёта. Без файла отчёта будет показана ошибка:

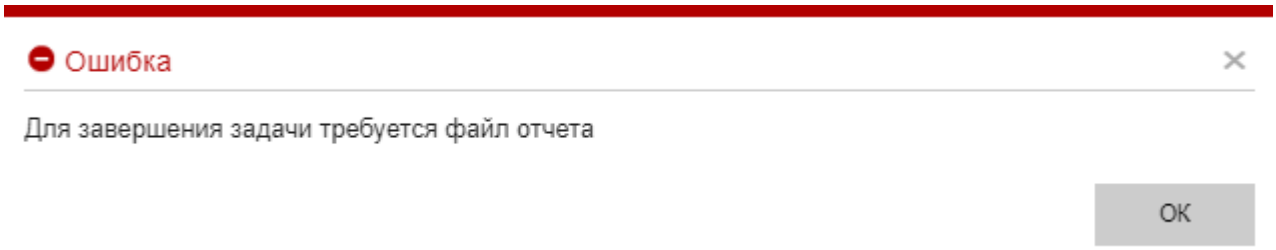


Рисунок 209. Для завершения задачи требуется файл отчёта

3. Добавьте отчёт/файл отчёта, укажите параметры завершения и нажмите кнопку **Завершить**, чтобы завершить задание.

> Состояние задания изменится на Завершено. Если задание должно пройти приёмку, то состояние будет На приёмке.
>
> Если настройками задания предусмотрено автоматическое завершение дочерних заданий и/или групп задания при завершении родительского, они также будут завершены.

Добавление отчёта по заданию

Отчёт о выполнении всех содержащихся в задании работ может быть добавлен при завершении задания или в процессе работы с ним. Отчётом может являться: текстовый комментарий, файл, ссылка на карточку.

Добавление отчёта разрешено всем *Исполнителям* (текущему исполнителю, делегатам всех уровней), *Ответственному исполнителю* (если есть такая роль) во всех состояниях, кроме Подготовка, Завершено, Отклонено, Отозвано.

При прикреплении отчёта тип ссылки всегда будет определяться настройкой,

заданной для соответствующего вида задания в *Справочнике видов карточек* системы Docsvision (вкладка *Задания*, поле Типы ссылок для отчётов).

Чтобы добавить отчёт в процессе выполнения задания:

1. Откройте карточку задания.
2. Перейдите к разделу *Отчёт*.

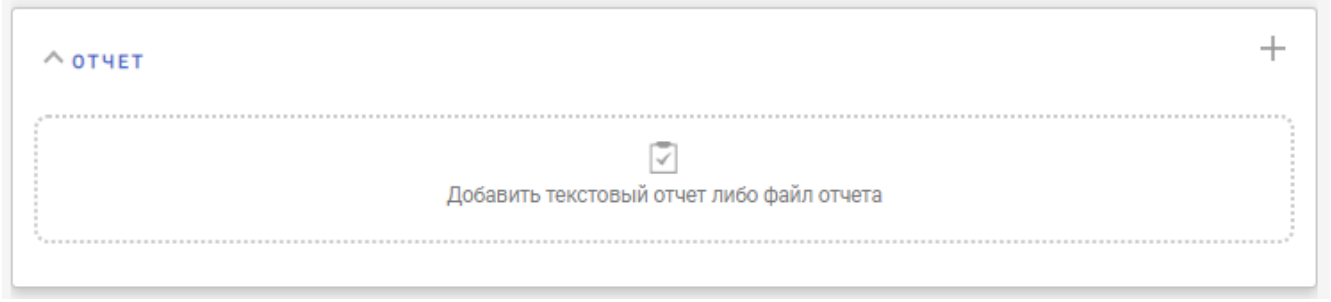


Рисунок 210. Раздел "отчёт" в карточке Задания

3. Приложите отчёт:
   - Файл отчёта.
   - Существующую или новую карточку.
   - Текстовый отчёт.

Добавление отчёта в зависимости от его вида:

Добавить отчёт в виде файла:

1. Нажмите + > .
2. Выберите добавляемый файл.

> Хранение букв ё и й в операционных системах MacOS и Windows различается. Если вы планируете работать с файлами в обеих системах, рекомендуется не использовать эти буквы в названиях файлов.

Длина имени файла в любой ОС не должна превышать 255 байт.

Отчёт в виде новой карточки:

1. Нажмите + > .
2. Выберите вид документа, заполните поля новой карточки и сохраните изменения. Ссылка на связанную карточку документа появится в разделе *Отчёт*.

Отчёт в виде существующей карточки:

1. Нажмите + > .
2. Выберите существующую карточку документа.

Текстовый отчёт:

1. Щелкните по области **Добавить текстовый отчёт либо файл отчёта**.
2. Введите комментарий в появившемся поле для ввода текста.
3. Нажмите кнопку **Сохранить**, чтобы добавить текст в отчёт.

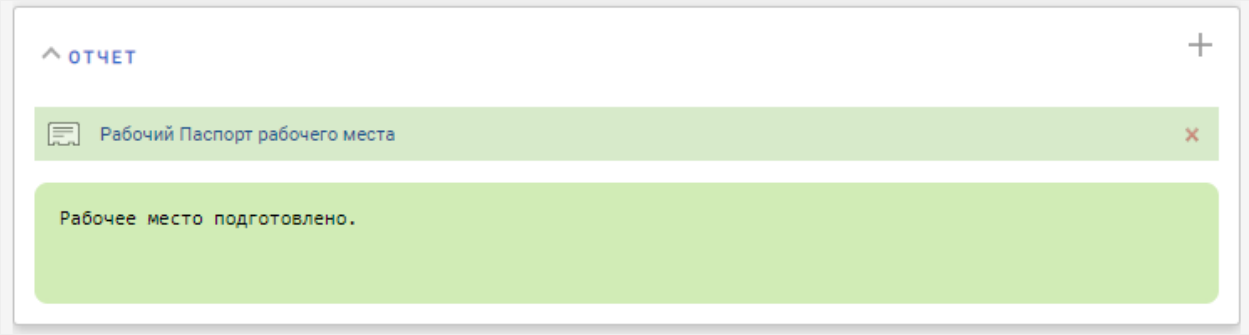


Рисунок 211. Пример заполнения отчёта

Отказ от выполнения задания

Если предусмотренные заданием работы не могут быть выполнены, отклоните его, указав причину отказа.

Чтобы отказаться от выполнения задания:

1. Откройте задание.
2. На панели операций нажмите кнопку **Отклонить**.

   Будет предложено подтвердить операцию.

3. Укажите причину отклонения и нажмите **Отклонить**.

   Состояние задания изменится на Отклонено.

Получение задания на доработку

Сотрудник, осуществляющий приёмку задания, может вернуть его исполнителю на доработку. Задание возвращается исполнителю, который его завершил.

Чтобы получить задание на доработку:

1. Перейдите в папку **Мои задания** › **Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение** › **Входящие**

2. Откройте задание, требующее доработки. Данное задание будет в состоянии `На доработке`.

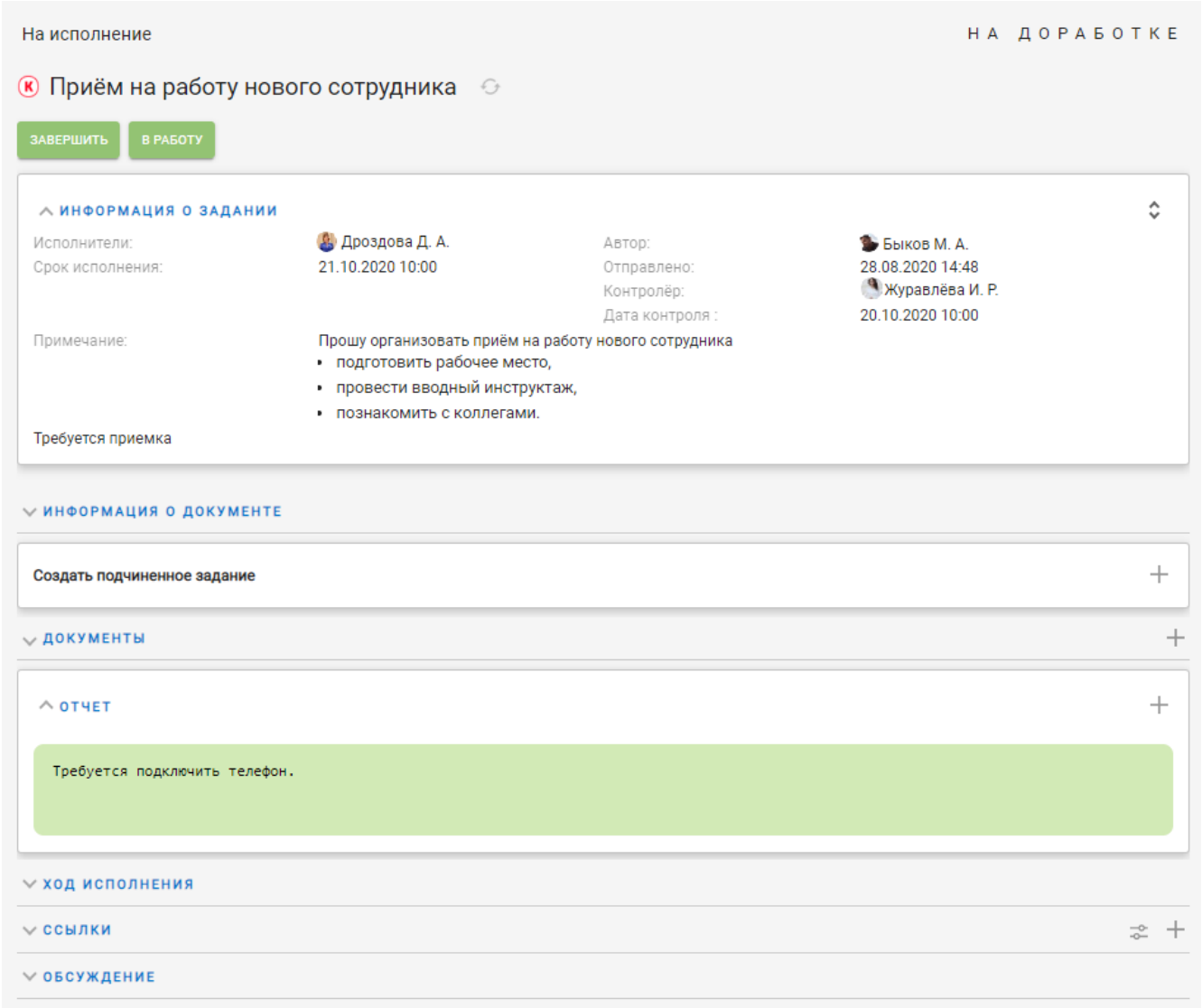


Рисунок 212. Открытое задание в состоянии "На доработке"

> Если сотрудник, осуществляющий приёмку, указал причину отказа, она будет указана в разделе *Обсуждение*.

3. Выполните дополнительные работы по заданию и завершите его.

Задание перейдет в состояние На приёмке и будет отправлено Контролёру для повторной проверки.

Делегирование задания

Делегирование — передача своего задания другому исполнителю. Делегировать можно задание в любом состоянии, кроме Возвращено с делегирования, На приёмке, Завершено.

Задание может быть делегировано несколько раз, если текущий исполнитель не установит запрет на повторное делегирование.

Механизм делегирования не следует использовать для передачи заданий по цепочке исполнения (например, от руководителя к подчиненным). Чтобы передать задание по цепочке, используйте связанные задания.

Чтобы делегировать задание другому исполнителю:

1. Откройте задание.
2. На панели операций нажмите кнопку **Делегировать**.

Будет предложено подтвердить операцию.

После подтверждения появится панель делегирования.

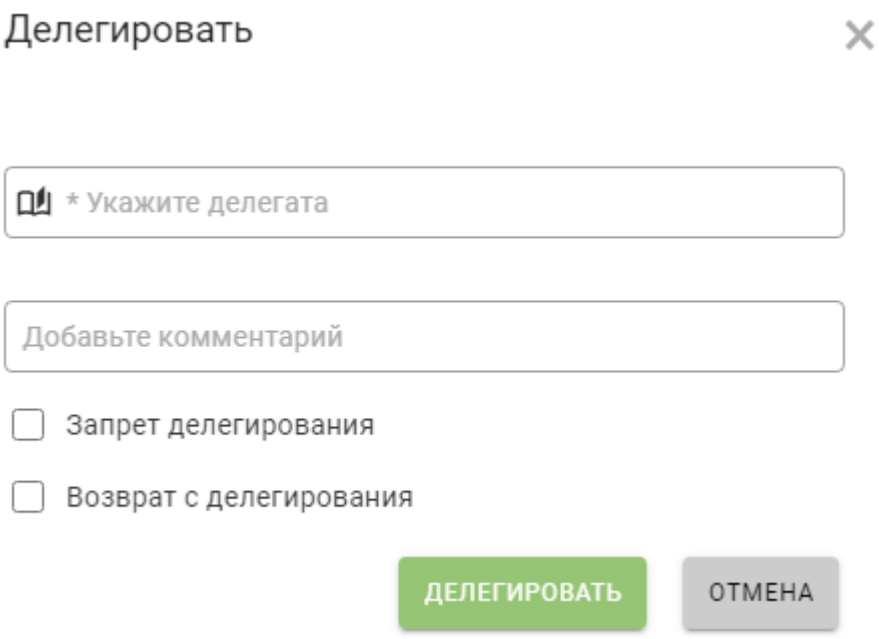


Рисунок 213. Делегирование задания

3. Укажите параметры делегирования.

Параметры делегирования

- `Укажите делегата`. Делегат — сотрудник, которому передаётся задание.
- `Добавьте комментарий`. Комментарий поможет понять причину делегирования задания. Комментарий будет добавлен в информационный блок делегирования, на странице *Главное* карточки задания, а также в комментарии.
- **`Запрет делегирования`**. Запрещает новому исполнителю делегировать задание другому сотруднику.
- **`Возврат с делегирования`**. Задание вернётся текущему исполнителю, после выполнения делегатом. Такое задание вернётся делегировавшему в состоянии `Возвращено с делегирования`.

4. Нажмите кнопку **Делегировать**

Состояние задания изменится на `Делегировано`. Задание будет перемещено в папку **Мои задания** › **Исходящие** › **Делегировано**. *Делегат* увидит данное задание в папке **Мои задания** › **Входящие**.

5. Чтобы ознакомиться с историей исполнения задания, нажмите на ссылку в списке *Исполнители*.

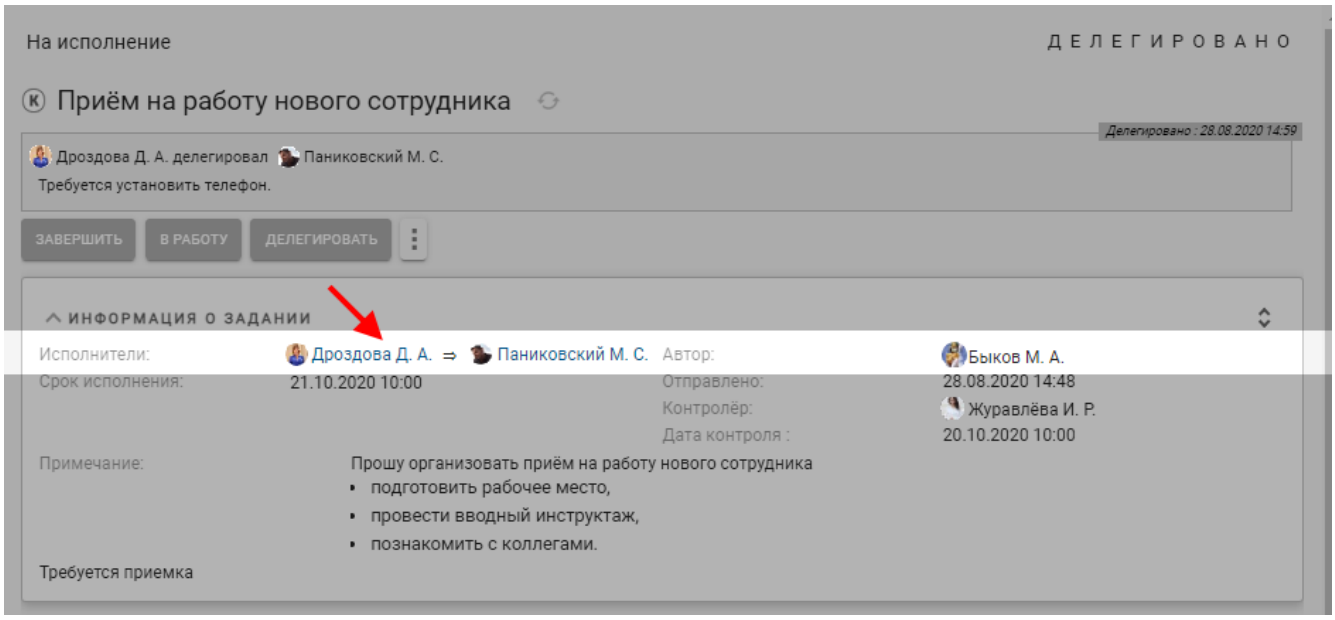


Рисунок 214. Ссылка для открытия Истории исполнения

Журнал делегирования появится на экране.

В таблице будет доступна информация о пользователе, выполнившем делегирование, назначенном делегате, дате назначения и комментарии.

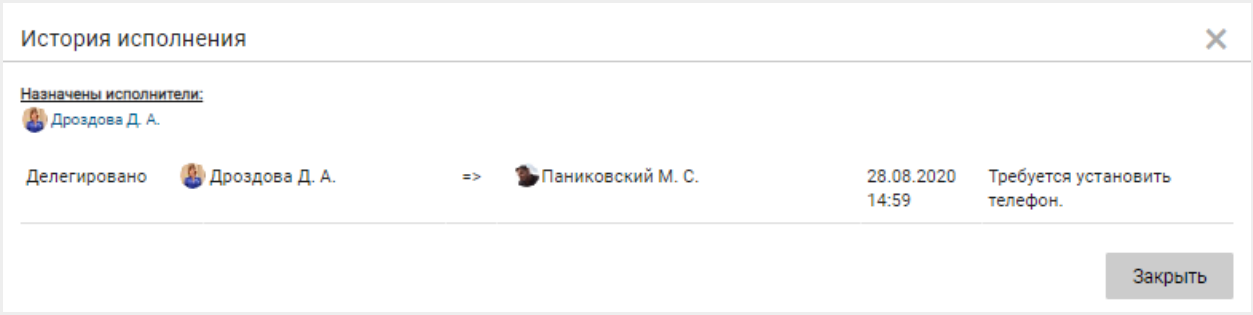


Рисунок 215. История исполнения

Отмена делегирования

Исполнитель, делегировавший задание, может отменить его.

1. Откройте карточку делегированного задания.
2. На панели операций нажмите кнопку **Отменить делегирование**.

Карточка перейдет в состояние `Не начато` или `Делегировано`.

Состояние `Делегировано` будет применено, если задание было делегировано несколько раз до отмены. В таком случае новым

исполнителем задания будет либо первоначальный, либо предыдущий делегат.

Информация об отмене делегирования, делегировавшем, его решении, дате принятия данного решения и комментарии записана в истории карточки.

У исполнителя карточка переместится в папку **Мои задания** › **Входящие**, у автора — в **Мои задания** › **Исходящие**.

Получение завершённого задания от делегата

Если при делегировании задания был установлен флаг `Возврат с делегирования`, после завершения оно будет возвращено делегировавшему.

Комментарий, добавленный делегатом при завершении задания, отображается в области *Отчёт*.

Задание будет возвращено в состоянии `Возвращено с делегирования`. Дальнейшая работа с заданием выполняется стандартным образом.

Исполнение задания делегатом

Делегат — пользователь, которому было передано задание другого исполнителя. После получения задания делегат становится новым исполнителем данного задания.

В статусе *Делегат* пользователь обладает следующим набором прав на выполнение операций с полученной карточкой:

- Открытие и просмотр карточки.
- Выполнение операций со связанными документами.
- Операции со связанными заданиями и документами.
- Управление комментариями.

Делегат может изменять состояние карточки:

- Получить делегированное задание.
- Отклонить делегированное задание.
- Выполнить и завершить задание.

- Делегировать задание.

Чтобы выполнить делегированное задание:

1. Перейдите в папку **Мои задания › Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение › Входящие**

2. Откройте задание в состоянии Делегировано.

   Информация о делегировании отображается в верхней части страницы.

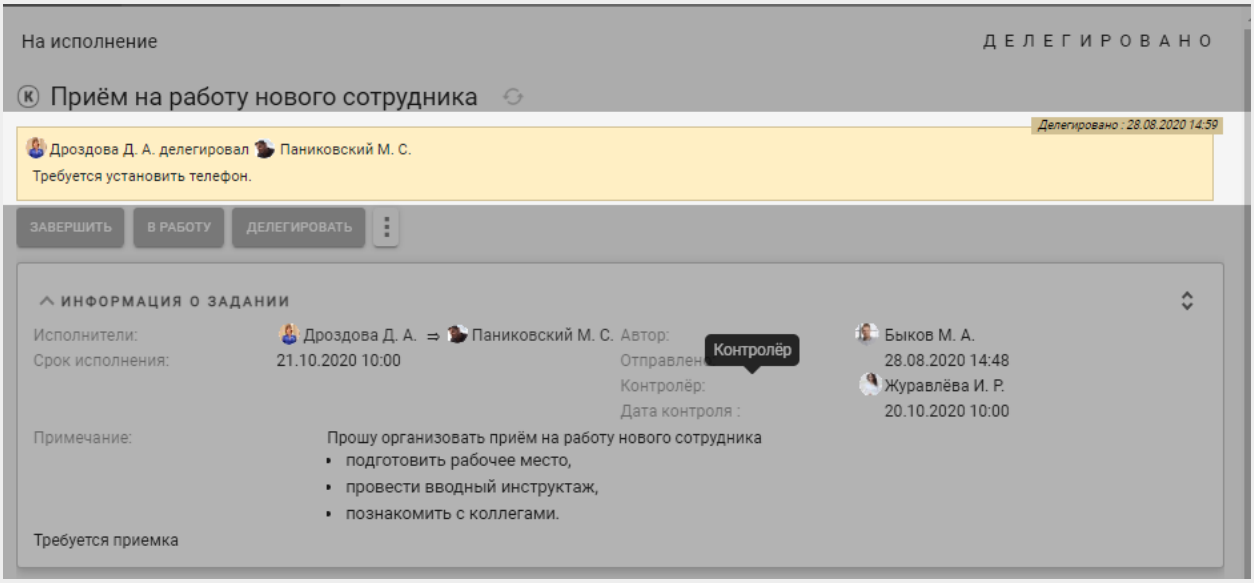


Рисунок 216. Задание делегата в состоянии "Делегировано"

3. Выполните задание стандартным образом.

Исполнение задания заместителем

Задание поступит заместителю, указанному в *Справочнике сотрудников*, если выполнены условия:

Для временного заместителя:

- В карточке замещаемого сотрудника пользователь указан как временный заместитель и являться первым *активным* пользователем в списке временных заместителей.
- Состояние замещаемого сотрудника любое, кроме Активен.
- В настройках вида карточки *Задание* должно быть настроено свойство "Автоматически делегировать при неактивности исполнителя". При этом рядом с полем Исполнитель становится доступен признак Заместителю).

Для постоянного заместителя:

- В карточке замещаемого сотрудника пользователь указан как постоянный заместитель.
- Состояние замещаемого сотрудника любое, кроме Активен.

Чтобы выполнить задание замещаемого сотрудника:

1. Перейдите в папку **Замещение** › **Входящие**, если являетесь постоянным или временным заместителем.
2. Откройте задание.

   > Исполнителем задания будет указан замещаемый сотрудник.

3. Выполните задание стандартным образом.

Комментирование задания

Обсуждение работ по заданию возможно после отправки его на исполнение.

Комментарии видят все участники данного задания. Комментировать задание могут пользователи, обладающие данным правом: *Автор*, *Исполнитель*, *Контролёр*, *Согласующий*.

Для работы с комментариями перейдите к разделу *Обсуждение* карточки Задание.

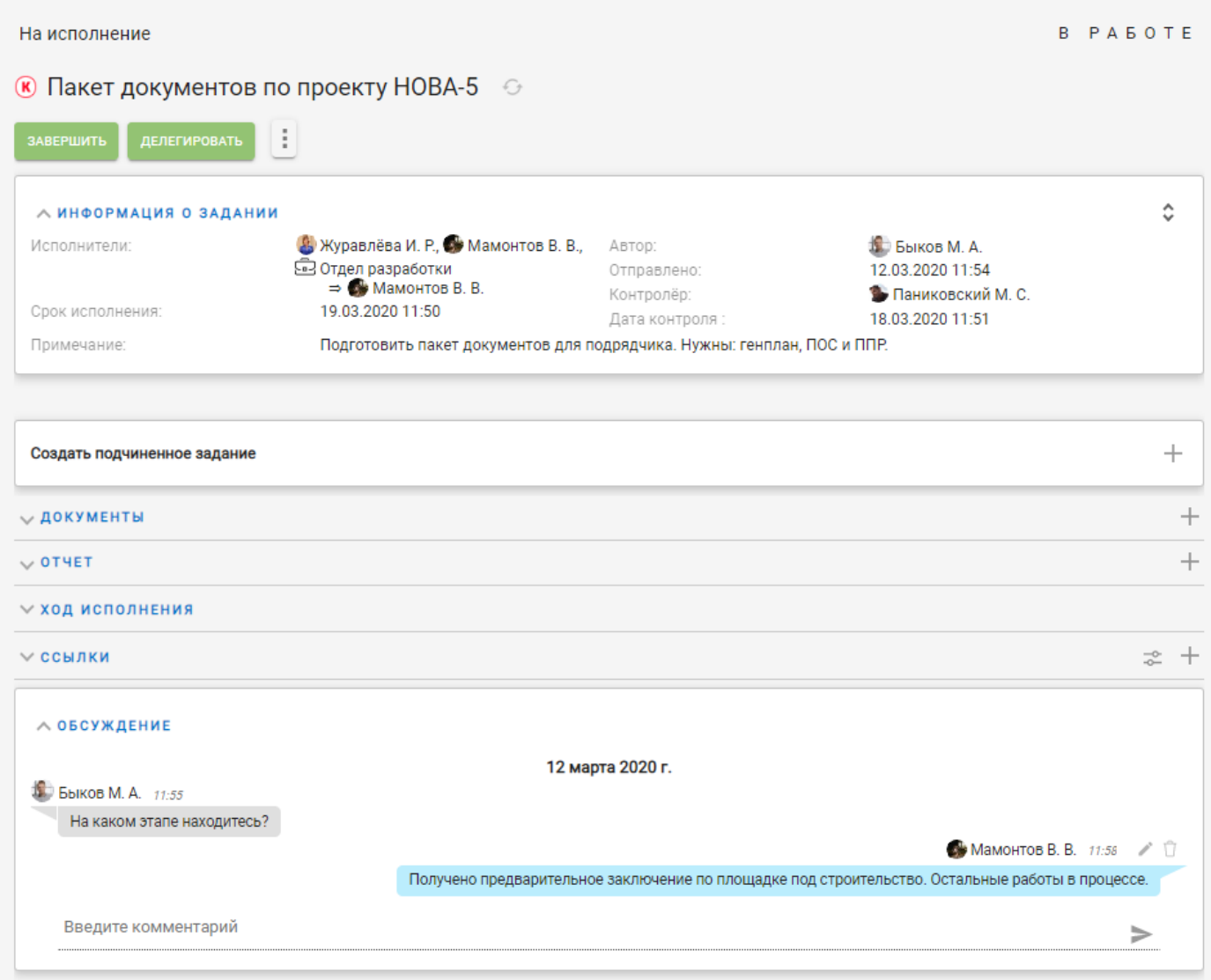


Рисунок 217. Страница для работы с комментариями карточки Задание

Общая инструкция по работе с комментариями приведена в пункте Комментарии.

Перенос срока исполнения задания

В Web-клиенте версии 6.1, сборка 1528 и выше исполнитель задания может создать запрос на перенос срока исполнения. Инициатор задания увидит запрос в карточке задания и сможет его обработать.

Вся информация о запросе на перенос для задания и работе с ним записывается в секции *Обсуждение*.

Перенос срока исполнения задания доступен для заданий решения Управление документами.

Перед использованием функции переноса срока исполнения, её

требуется, см. подробнее в разделе для администратора: "Настройка запроса на перенос срока исполнения заданий".

Запрос на перенос срока

Чтобы создать запрос на перенос срока исполнения:

1. Нажмите на ленте задания кнопку **Запрос изменения срока**. Откроется боковая панель *Запрос изменения срока*.

   Кнопка может быть недоступна из-за настроек ролевой модели или параметров разметок.

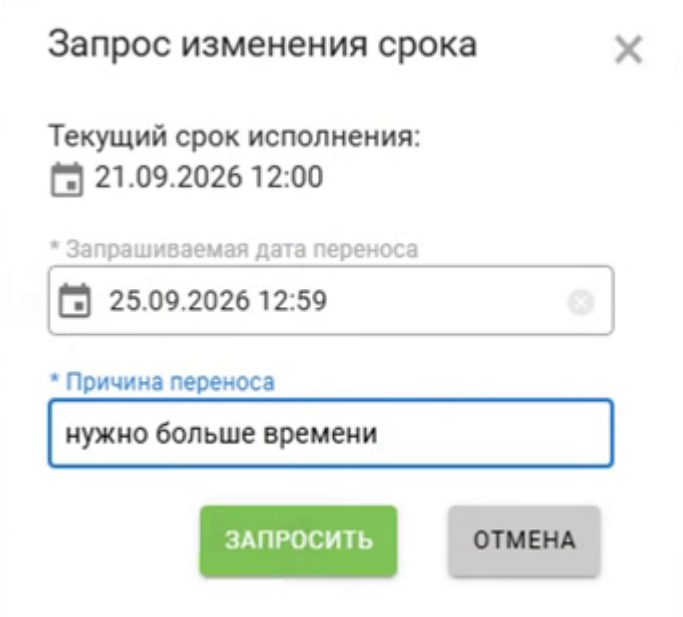


Рисунок 218. Панель запроса на изменение срока исполнения

2. Ознакомьтесь с действующим сроком исполнения задания в поле *Текущий срок исполнения*.

   Срок может отсутствовать, если он не обозначен в задании.

3. Укажите новый срок исполнения в поле *Запрашиваемая дата переноса*.

   Обязательное для заполнения поле по умолчанию отображает текущий срок исполнения. Если текущий срок не задан, будет отображаться текст `Запрашиваемая дата переноса`.

4. Укажите комментарий в поле *Причина переноса*.
5. Нажмите кнопку **Запросить**, чтобы создать запрос на перенос срока или обновить существующий запрос.

Нажатие кнопки **Отмена** закроет панель без сохранения внесённых изменений.

6. Исполнитель может отозвать собственный запрос при помощи кнопки **Отозвать** в элементе управления запроса.

Работа с запросом на перенос

Созданный исполнителем запрос на перенос срока отображается в карточке инициатора. Инициатор может подтвердить или отклонить запрос.

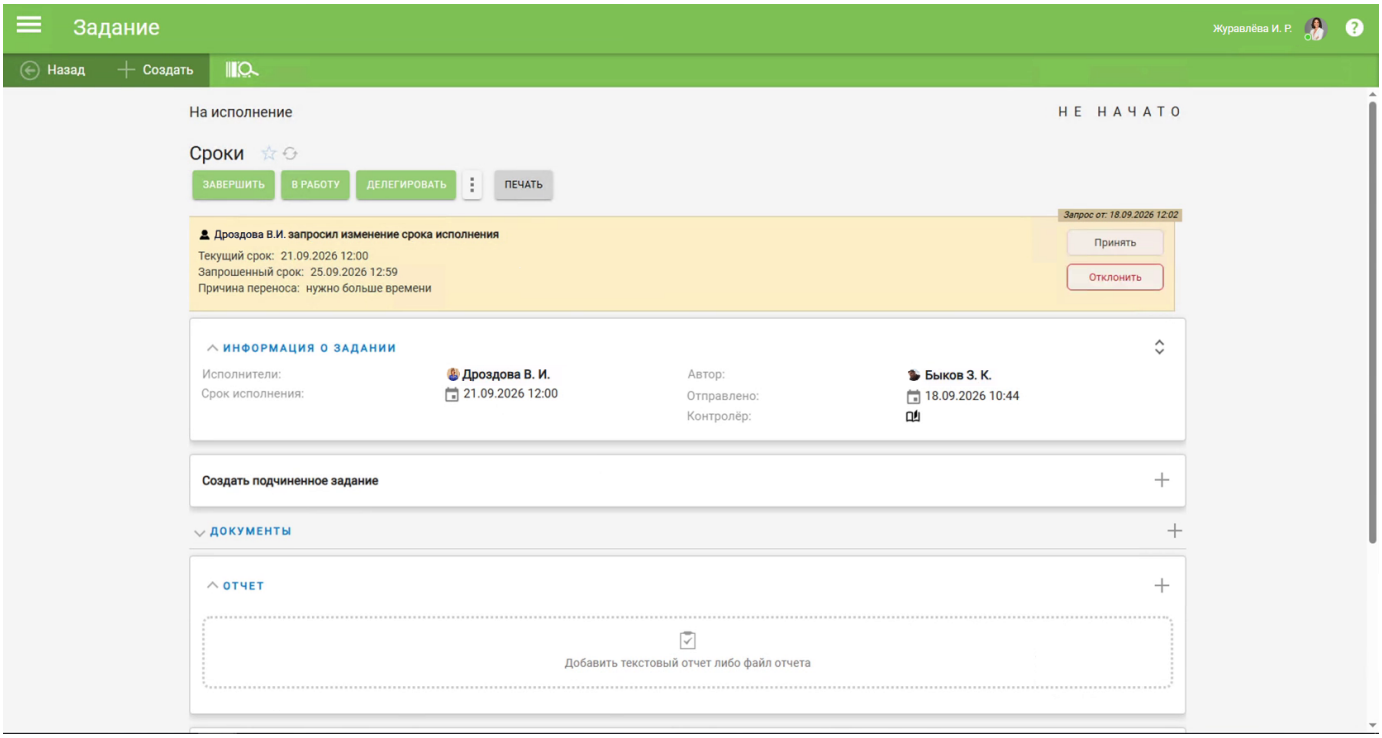


Рисунок 219. Отображение запроса на перенос срока исполнения в карточке

Запрос отображается, только когда в задании был запрошен перенос срока и он находится в статусе *Активен*. В остальных случаях запрос скрыт.

Запрос отображается отдельным элементом управления разметки под лентой карточки в виде текста на жёлтом фоне. В элементе управления представлена вся информация о запросе, включая исполнителя, запросившего перенос срока, запрошенный срок, комментарий исполнителя. Если текст комментария не помещается в одну строку, он будет сокращён. При наведении курсора на сокращённую строку будет отображён полный текст комментария. В правой части элемента управления отображаются две кнопки для принятия решения о переносе срока исполнения: **Принять** и **Отклонить**.

Нажатие на кнопку **Принять** принимает запрос и устанавливает новый срок исполнения.

Нажатие кнопки **Отклонить** отменяет запрос и оставляет срок исполнения неизменным.

После принятия или отклонения запроса панель элемента управления с информацией будет скрыта.

Если инициатор не согласен с запрашиваемым сроком, он вправе отклонить запрос на изменение срока исполнения задания и перенести дату самостоятельно при помощи существующей в Управление документами команды на ленте карточки — "Изменение срока исполнения".

Информация о переносе сроков

При создании, обновлении, принятии, отклонении и отзыве запроса переноса срока исполнения информация о запросе автоматически добавляется в блок *Обсуждение* карточки задания.

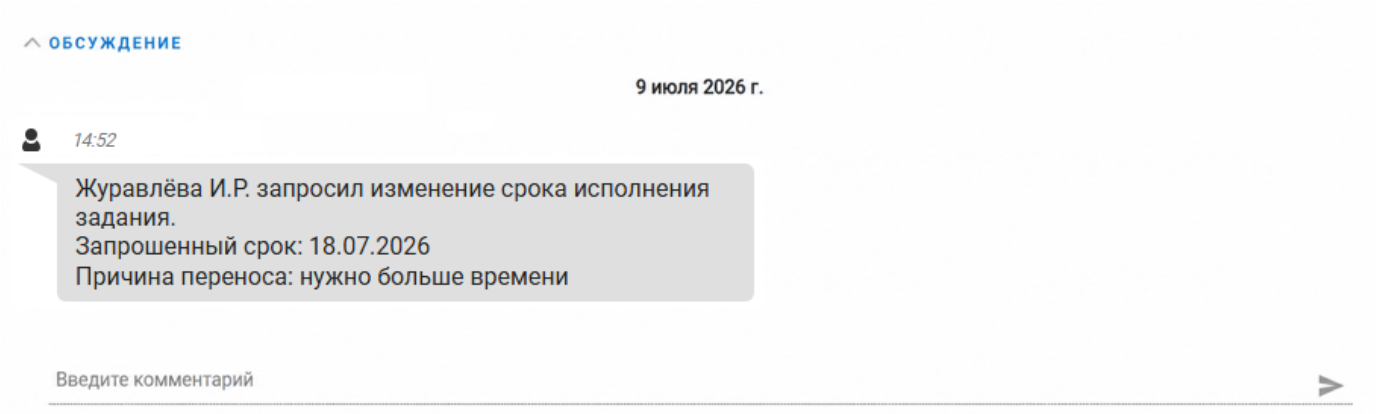


Рисунок 220. Информация о запросе на перенос сроков исполнения в блоке "Обсуждение"

Для запросов на перенос срока исполнения кнопки редактирования и удаления отображаются только пользователям, имеющим права на изменение и удаление любого комментария. Если сотруднику доступно только изменение или удаление собственного комментария, кнопки изменения и удаления не отображаются, даже если автор запроса — этот сотрудник.

Комментарий добавляется, даже если у сотрудника отсутствуют права на его добавление. При изменении срока исполнения с помощью операции "Изменение срока исполнения", отдельный комментарий в блоке *Обсуждение* не добавится.

Автоматический отзыв активного запроса на перенос срока

Активный запрос переноса срока автоматически переводится в статус Отозван в случаях, если:

- Задание перешло в состояние `Завершено`, `Отозвано` или `Отклонено`.
- Задание делегировано другому исполнителю.

  Поскольку, при делегировании изменяется текущий исполнитель задания, создатель активного запроса больше не обладает правами исполнителя и возможностью отозвать задание. Правило действует при делегировании с возвратом и без возврата.

- Срок исполнения задания изменен через существующую операцию *Изменение срока исполнения*, но активный запрос в блоке не был отклонен.

Срок исполнения не изменяется при автоматическом отзыве задания. Подтверждение сотрудника в таком случае не требуется. В запросе сохраняется дата обработки и сотрудник, действие которого привело к завершению, отзыву или отклонению задания.

Контроль исполнения заданий

Контролёр — сотрудник, который проверяет качество выполнения задания после завершения задания *Исполнителем*.

В статусе *Контролёр* пользователь обладает следующими правами на выполнение операций с карточкой:

- Открытие и просмотр карточки.
- Выполнение операций со связанными документами.
- Операции со связанными заданиями и документами.
- Управление комментариями.

Контролёр может изменить состояние карточки:

- Контролировать выполнение задания;
- Отправить задание на исполнение;
- Отозвать задание у исполнителя;
- Завершить задание.

Получение задания контролёром

1. Перейдите в папку **Мои задания** › **Входящие** › **На контроле**. Заместителю нужно перейти в папку **Замещение** › **Входящие** › **На контроле**.

2. Откройте полученную карточку *Задания*.

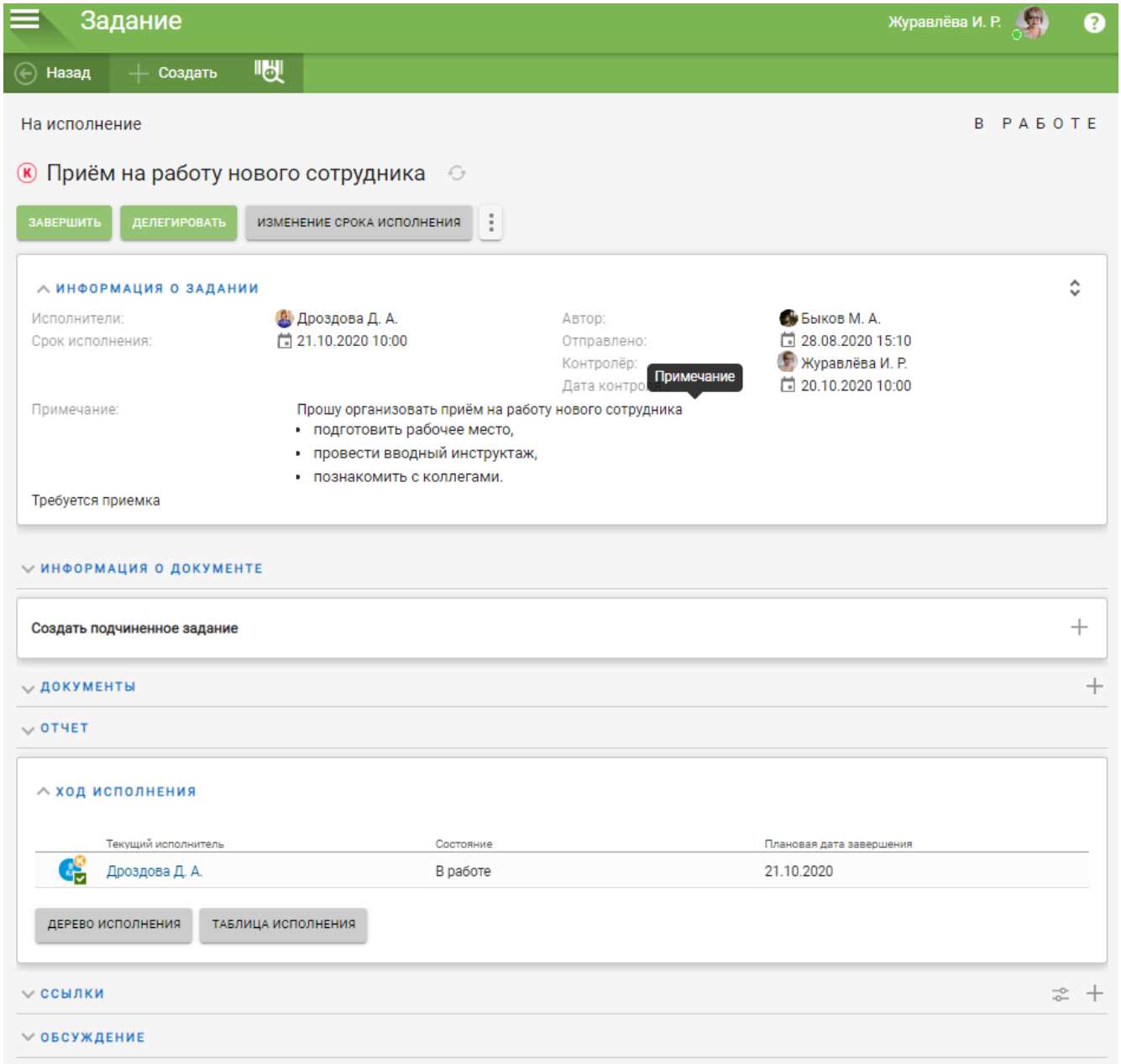


Рисунок 221. Карточка задания у контролёра

3. Проведите приёмку задания, если оно в состоянии `На приёмке`.



Контролёру также доступны операции автора задания.

Приёмка задания

Приёмка задания выполняется Контролёром, если в задании установлен признак `Требуется приёмка`.

Приёмка используется для проверки качества выполнения задания

исполнителем. По результатам приёмки контролёр может либо принять задание, либо вернуть задание исполнителю для доработки.

Чтобы выполнить приёмку задания:

1. Откройте задание в состоянии `На приёмке`.
2. Откройте полученную карточку *Задание*.

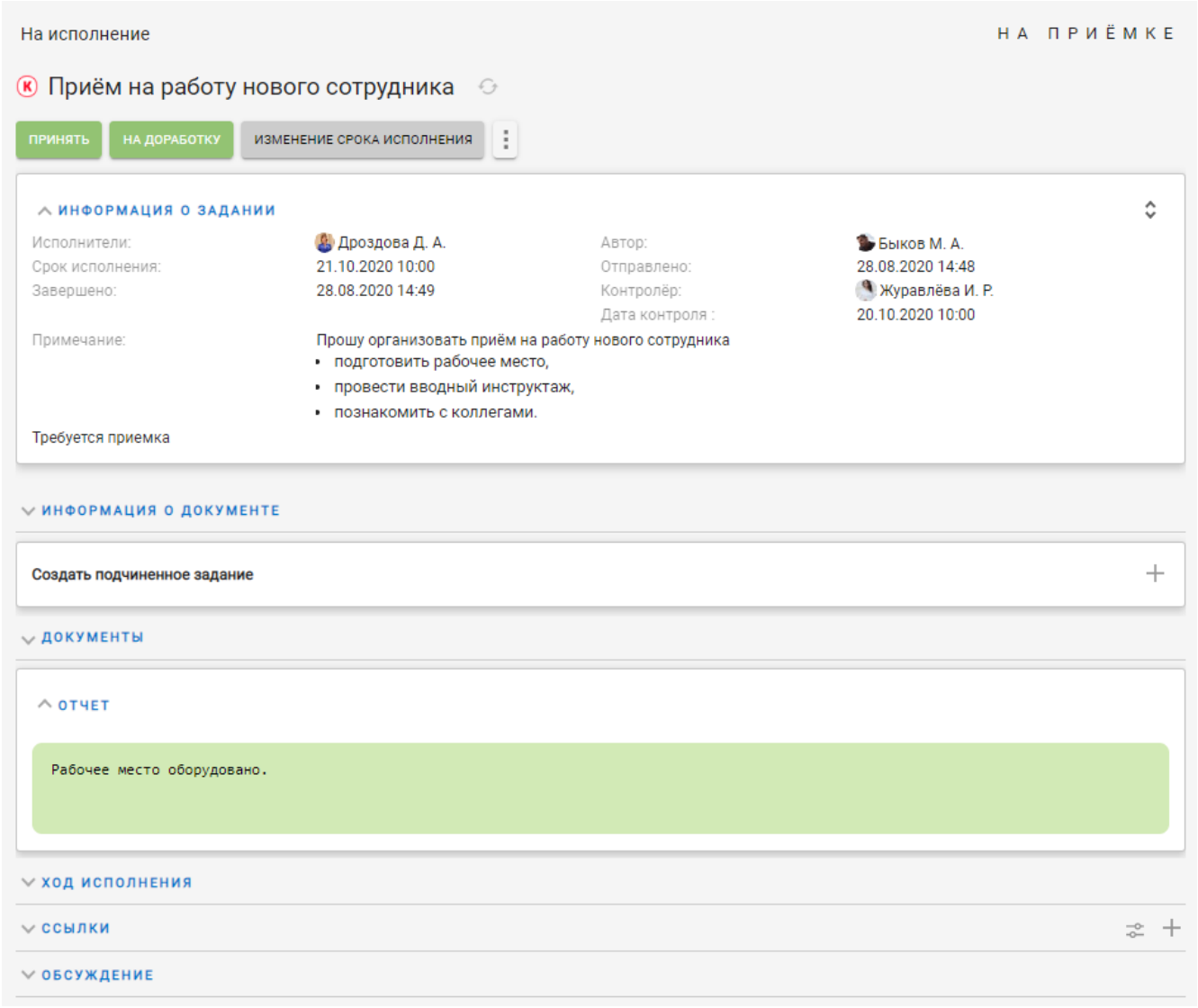


Рисунок 222. Карточка задания контролёра

3. Примите задание или отправьте его на доработку.

   - Нажмите кнопку **Принять**, чтобы завершить задание. Задание перейдет в состояние `Завершено`. В задании будет указана фактическая дата завершения.
   - Чтобы вернуть задание исполнителю для внесения изменений, укажите причину возврата и нажмите кнопку **На доработку**.

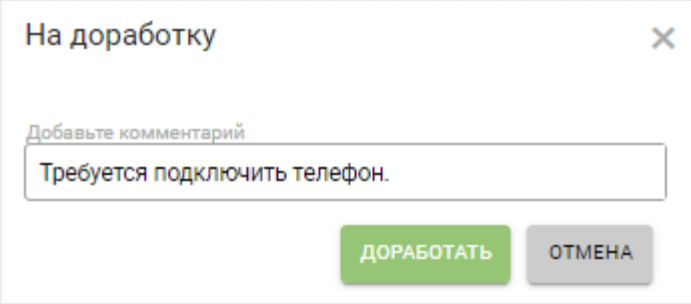


Рисунок 223. Ввод причины возврата задания на доработку

Задание перейдет в состояние `На доработке`.

Исполнитель увидит причину возврата в комментариях к заданию.

После завершения задания исполнителем, оно повторно вернется на приёмку.

Работа со связанными заданиями и документами

К заданиям могут прилагаться связанные задания или документы.

Связанные задания позволяют разделить работу над большой задачей между несколькими исполнителями.

Связанные документы — основные файлы карточки Задание.

В данном разделе описана работа с такими файлами или документами.

Создание подчиненных заданий и групп заданий



Возможность создания подчинённых заданий и/или группы заданий может быть ограничена при настройке Web-клиента.

Для создания подчиненного задания, пользователь должен обладать правами на изменение задания.

Чтобы создать подчинённое задание или группу заданий:

1. Откройте карточку задания, для которого создаётся подчинённое.
2. Перейдите к области *Создать подчинённое задание*.
3. Нажмите + и выберите вид создаваемого задания или группы заданий.

Будет открыта карточка *Задание* или *Группы заданий* с документом

основного задания.

4. Заполните задание/группу заданий и отправьте на исполнение.

> Подчинённые задания и группы заданий отображаются в дереве исполнения в разделе *Ход исполнения*.

Операции со связанными документами карточки Задание

Под связанными документами подразумеваются основные файлы карточки Задание.

Добавить документы в задание

Добавить документ в задание может любой участник задания: автор задания, исполнитель задания или контролёр.

Чтобы добавить документ:

1. Откройте карточку задания.
2. Перейдите к области *Документы*.
3. Нажмите +. Будет открыто окно выбора файла из файловой системы.
4. Выберите файл.

> Для файла будет создана карточка *Файл*, ссылка которую будет добавлена в раздел *Документы*.

Открыть карточку связанного документа из карточки Задание

1. Откройте карточку задания.
2. Перейдите к области *Документы*.
3. В строке открываемой карточки файла нажмите ⋮ **> Открыть карточку файла**. Будет открыта карточка выбранного файла.

Открыть файл связанного документа из карточки Задание

1. Откройте карточку задания.
2. Перейдите к области *Документы*.
3. В строке открываемого файла нажмите ⋮ **> Открыть**.

Файл будет открыт в редакторе файлов данного типа в режиме чтения или редактирования, если есть права на редактирования. При открытии в режиме редактирования файл будет заблокирован.

Сохранить файл задания на компьютер

1. Откройте карточку задания.
2. Перейдите к области *Документы*.
3. В строке сохраняемого файла нажмите ⋮ **> Скачать**.

Файл будет сохранен на компьютер пользователя.

Открепить связанный документ из карточки Задание

1. Откройте карточку задания.
2. Перейдите к области *Документы*.
3. В строке открепляемого файла нажмите ⋮ **> Открепить**.
4. Подтвердите открепление файла.

Файл и его карточка будут удалены.



Если задание создано дочерним к документу, то ссылку на родительский документ удалить нельзя.

Удаление задания

Карточка *Задание* может быть удалена пользователем при выполнении следующих условий:

- Карточка задания разблокирована.
- Карточка в состоянии `Подготовка` или `Отозвано`.
- Пользователь должен обладать достаточными правами на удаление карточки.

См. общую инструкцию по удалению в пункте Удаление карточки. При удалении задания ссылки на него будут удалены из других карточек.

Работа с группами заданий

Карточка *Группа заданий* предназначена для создания детализированных распоряжений по документу, разбитых на отдельные подзадачи. Каждая подзадача предназначена для определённого исполнителя или группы исполнителей, снабжена описанием работ, сроками их исполнения и другими параметрами, а также предусматривает необходимый контроль за ходом его реализации.

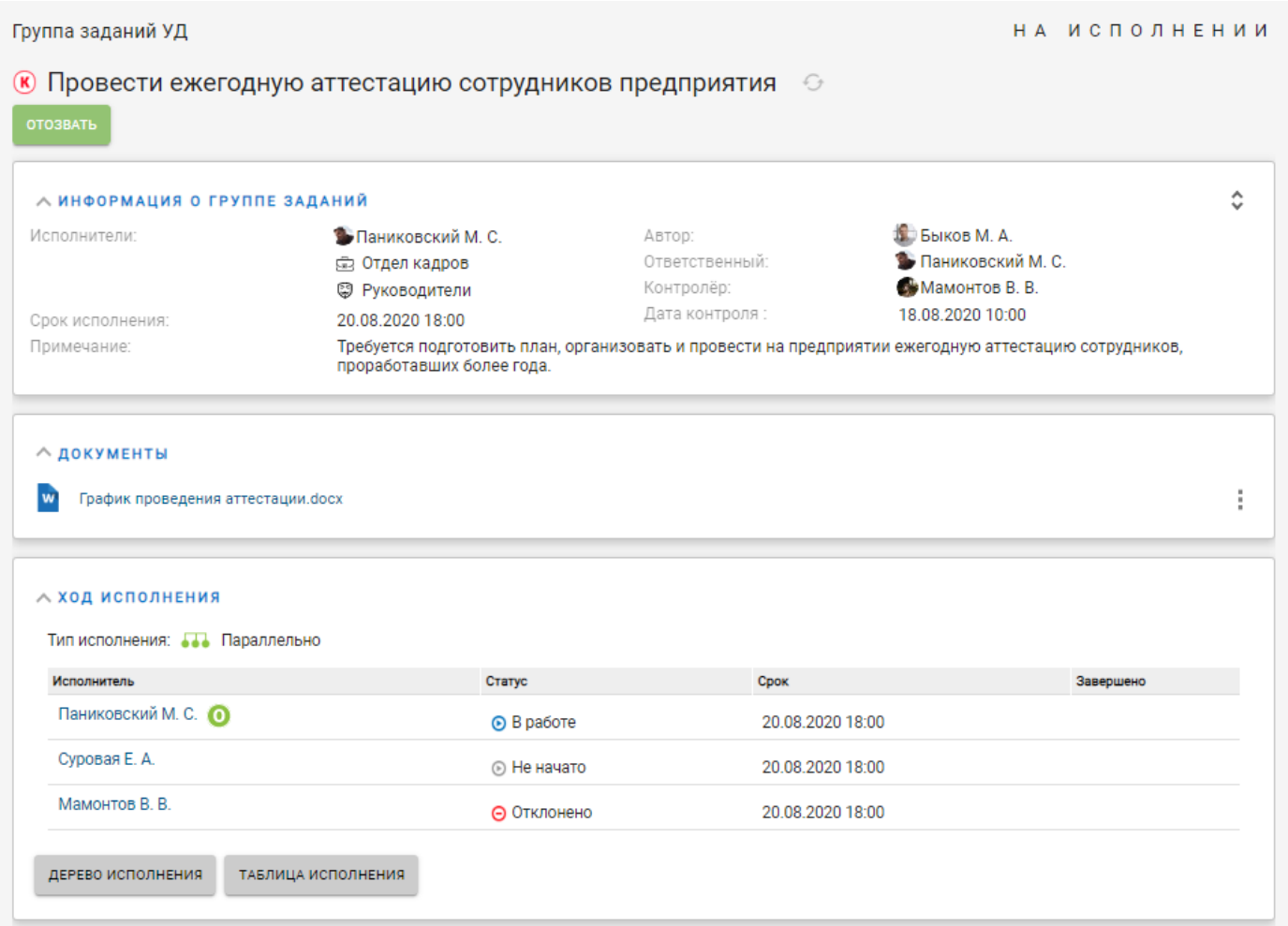


Рисунок 224. Группа задания, открытая у автора после формирования заданий для исполнителей

Создание группы заданий

1. Создайте карточку вида **Группа заданий › Группа заданий УД**.

   Будет открыто окно новой карточки.

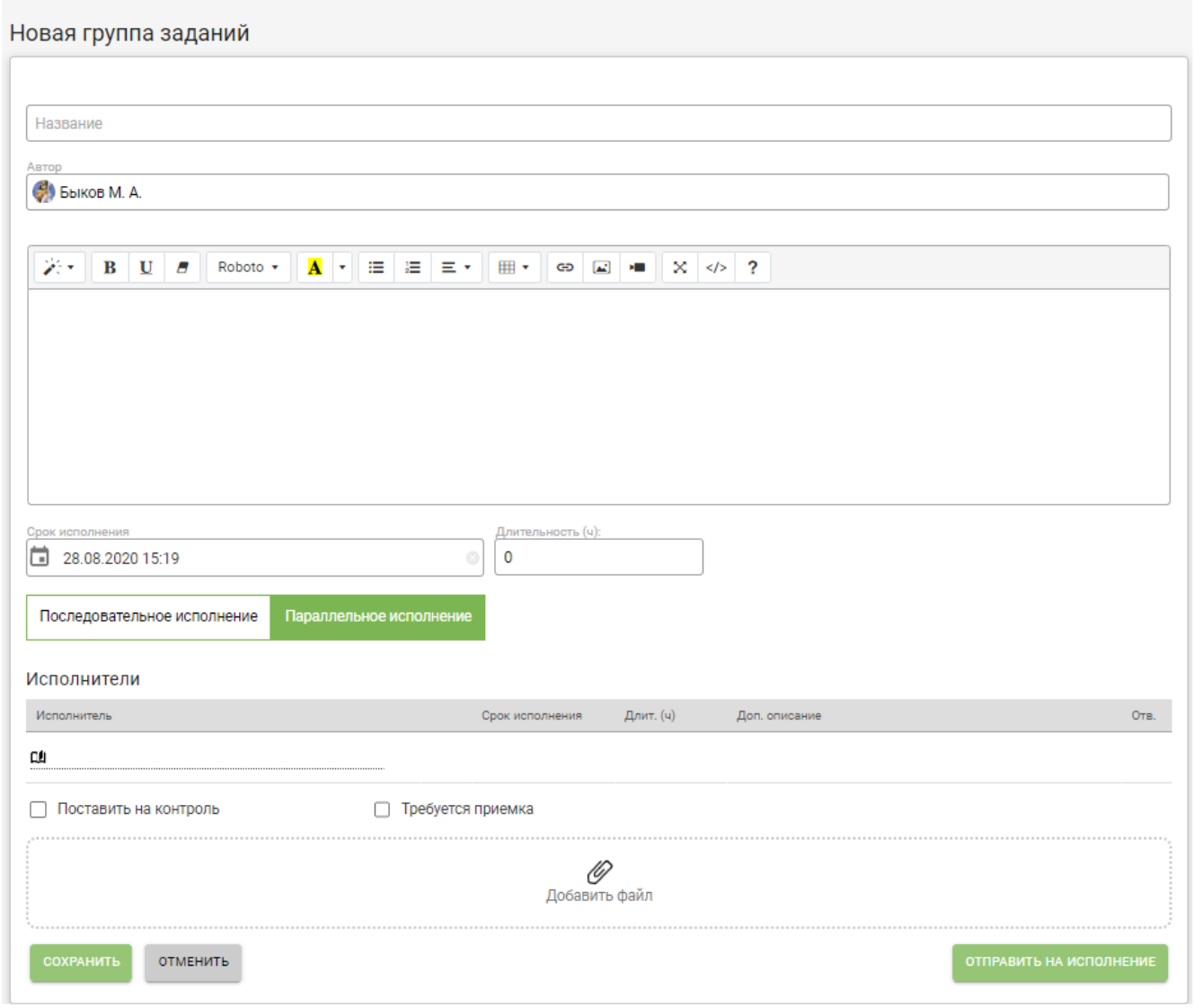


Рисунок 225. Новая группа заданий вида "Группа заданий УД"

2. Заполните поля группы заданий.
3. Нажмите кнопку **Отправить на исполнение**, чтобы отправить задания исполнителям или **Сохранить**, чтобы сохранить группы заданий с возможностью отправки. Карточка будет открыта в режиме просмотра.

> Задания для исполнителей формируются в течении некоторого времени.

Перечень полей карточки "Группа заданий"

Название

Название группы заданий.

Обязательно для заполнения.

Автор

Автор группы заданий. По умолчанию выбирается текущий сотрудник.

Обязательно для заполнения.

Примечание

Дополнительная информация по группе заданий.

Срок исполнения

Плановый срок исполнения для заданий группы. По умолчанию устанавливаются текущие дата и время.

Длительность исполнения пересчитывается с учётом бизнес-календаря:

- Для групп и ролей используется системный календарь.
- Для подразделений используется календарь подразделения или системный.
- Для сотрудников используется календарь сотрудника, подразделения или системный.

Длительность

Длительность исполнения заданий относительно текущей даты и времени.

При изменении длительности, срок исполнения будет пересчитан с учётом бизнес-календаря (см. правила выше).

Последовательное исполнение/Параллельное исполнение

Тип исполнения заданий группы:

- ***Параллельное исполнение***. Задания могут выполняться одновременно всеми исполнителями;
- ***Последовательное исполнение***. Задания будут выполняться исполнителями последовательно по отношению к порядку исполнителей в списке.

  При изменении типа исполнения нужно подтвердить пересчет даты исполнения заданий.

Исполнители

Исполнители группы заданий с индивидуальными настройками исполнения. См. пункт Выбор исполнителей группы заданий.

Обязательно для ввода. Должен быть добавлен хотя бы один исполнитель.

Поставить на контроль

Включает функцию контроля исполнения заданий группы ответственным сотрудником. При включённом контроле станут доступны параметры:

Контролёр

Сотрудник, который контролирует выполнение заданий.

Дата контроля

Дата, когда должен быть выполнен контроль выполнения заданий.

Требуется приёмка

Включает функцию приёмки выполненных заданий. При включении приёмки будет автоматически включён контроль.

Ознакомиться с особенностями контроля и приёмки можно в пункте Приёмка задания.

Добавить файл

Файлы, которые требуются для выполнения заданий. Файлы будут приложены ко всем заданиям исполнителей.

Файлы можно добавить с текущего устройства или со сканера (см. условия в описании элемента Список файлов).

Выбор исполнителей группы заданий

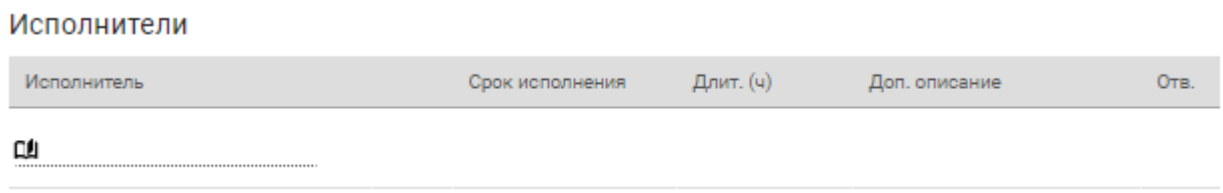


Рисунок 226. Элемент "Исполнители" в карточке Группа заданий

1. В столбце *Исполнитель* выберите исполнителя отдельного задания из группы заданий.

   В качестве исполнителя можно назначить (в любых комбинациях) *сотрудников*, *организации* и *подразделения*, *группы* и *роли*.

Можно добавлять исполнителей по одному или выбрать сразу несколько записей из Справочника сотрудников (каждая будет добавлена как отдельный исполнитель).



Чтобы удалить исполнителя из списка: наведите курсор на строку исполнителя и нажмите появившуюся кнопку ×.

2. Для исполнителя *Подразделение*, *Группа* и *Роль* выберите вариант рассылки заданий:
   - Отдельное задание для каждого сотрудника.
   - Одно задание для всех сотрудников.
   - Одно задание для руководителя (только для Организации или Подразделения).

Варианты рассылки заданий

Вариант рассылки заданий определяется позицией переключателя, представленного в виде круглого значка справа от названия подразделения, группы или роли:

Отдельное задание для каждого сотрудника

Для каждого сотрудника выбранного подразделения, группы или роли будет создано собственное задание.

Одно задание для всех сотрудников

Для сотрудников выбранного подразделения, группы или роли будет создано одно задание, фактическим исполнителем которого будет первый сотрудник, взявший данное задание в работу.

Одно задание для руководителя

Задание будет создано только для сотрудника, указанного руководителем данного подразделения или организации.

Данный вариант доступен только для подразделений и организаций. Если у подразделения не указан руководитель, при отправке группы заданий будет показана ошибка. В созданной группе заданий исполнителем будет указан сотрудник, а не подразделение.

Чтобы изменить вариант рассылки, нажмите на значок текущего

варианта рассылки.

Выполните следующие пункты, при необходимости

1. Измените порядок исполнения для группы заданий с последовательным исполнением. Для этого перетащите строку исполнителя на нужную позицию или наведите курсор на строку исполнителя и нажмите на одну из появившихся кнопок ↑ ↓.
2. Измените сроки исполнения для отдельных исполнителей в столбце *Срок исполнения*. Подробнее в пункте Настройка индивидуальных сроков исполнения.
3. Укажите дополнительное описание для задания (будет добавлено к основному описанию) в столбце *Доп. описание*.
4. Выберите сотрудника, ответственного за выполнение группы заданий, в столбце *Отв.*. Для этого нажмите на иконку в строке требуемого исполнителя.

Цвет иконки изменится на зеленый. Ответственным может быть назначен только сотрудник. Для подразделений, групп и ролей кнопка выбора ответственного будет скрыта.

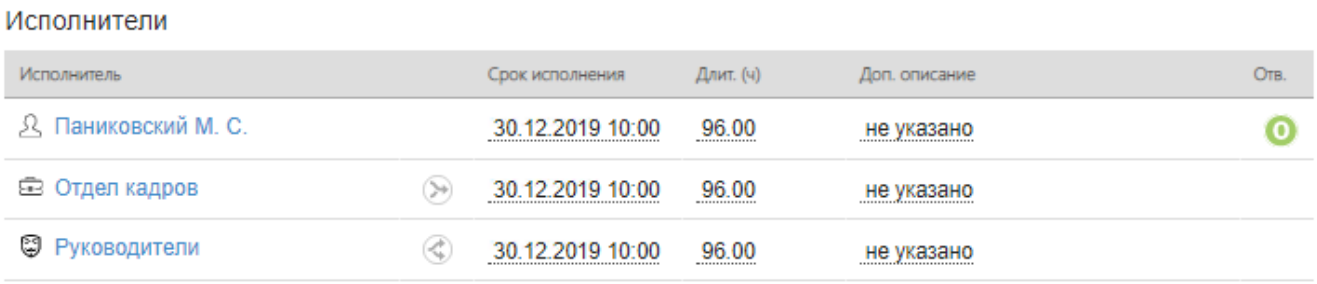


Рисунок 227. Назначение ответственного исполнителя

Сотрудник, назначенный ответственным за группу заданий, может получить доступ к заданиям всех остальных исполнителей и при необходимости завершить, отложить или изменить срок выполнения задания.

Особенности рассылки заданий исполнителям:

- При параллельном исполнении для сотрудника, указанного в списке несколько раз, например, в выбранном подразделении и выбранной

группе или роли, будет сформировано только одно задание. В приоритете будет задание, где выбран именно сотрудник, а не подразделение, группа, роль.

- При последовательном исполнении для сотрудника, указанного в списке несколько раз, будет создано соответствующее количество заданий.

Настройка индивидуальных сроков исполнения

Чтобы изменить индивидуальный срок исполнения заданий в группе заданий с последовательными исполнением:

1. Откройте окно редактирования сроков нажатием на поле со значением изменяемого срока исполнения или на поле с длительностью.

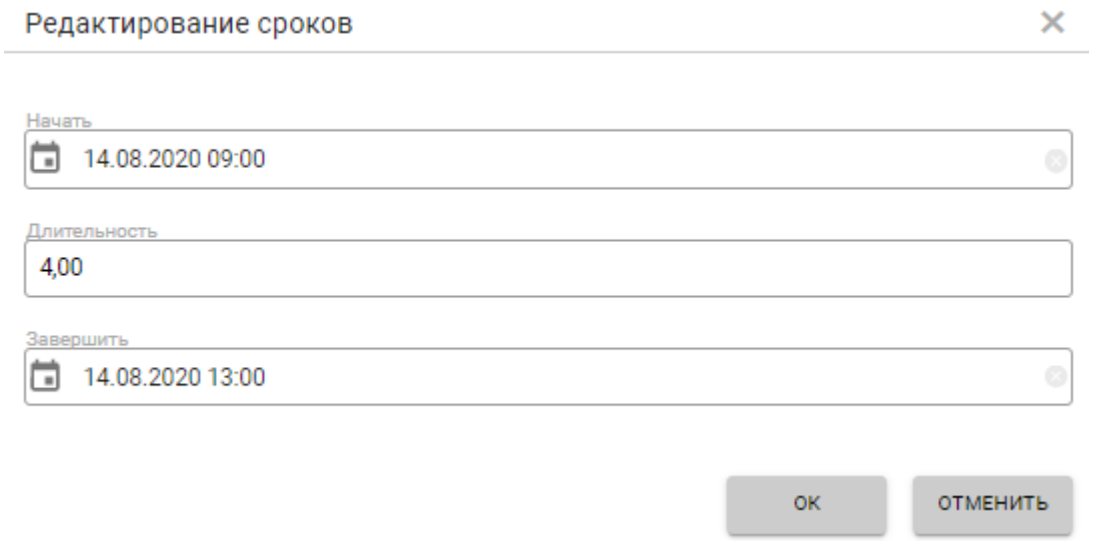


Рисунок 228. Изменение индивидуальных сроков исполнения заданий

2. Измените дату и время начала и/или завершения исполнения, или длительность. В зависимости от позиции исполнителя может быть недоступна возможность изменения начального или конечного срока.
3. Нажмите кнопку **ОК**, чтобы применить изменения.

При изменении индивидуального срока исполнения будет увеличен или уменьшен срок исполнения следующего исполнителя. Если этого недостаточно, будет предложено пересчитать сроки исполнения других исполнителей, следующих за текущим. Согласитесь с пересчетом.

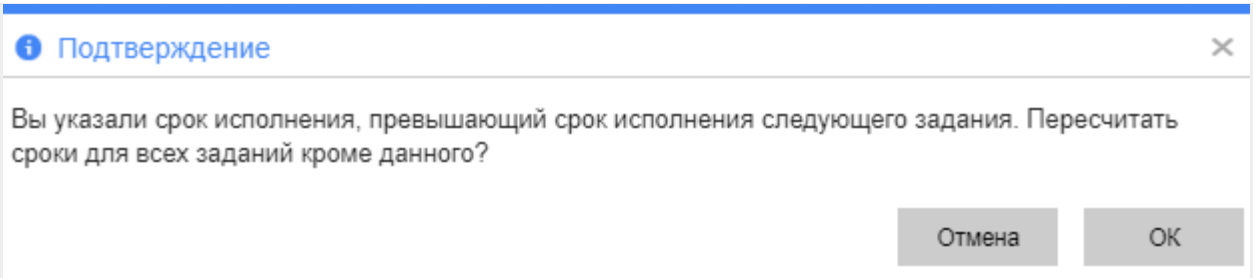


Рисунок 229. Запрос на пересчет сроков исполнителей

Если суммарные индивидуальные сроки исполнения для последовательного задания превысят установленный общий срок, будет показано предупреждение. Установите корректные индивидуальные сроки или измените общий срок исполнения, чтобы исправить ошибку.

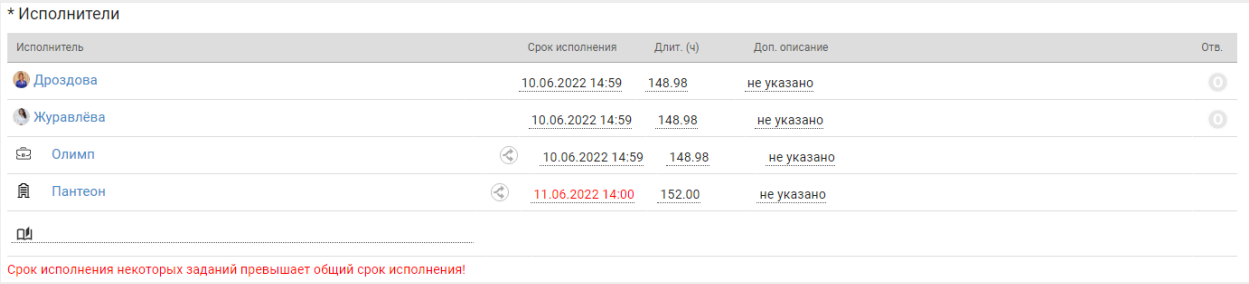


Рисунок 230. Предупреждение при превышении общего срока исполнения

Длительность исполнения пересчитывается с учётом бизнес-календаря:

- Для групп и ролей используется системный календарь.
- Для подразделений используется календарь подразделения или системный.
- Для сотрудников используется календарь сотрудника, подразделения или системный.

Особенности контроля заданий и передачи на приёмку

- При последовательном исполнении, пока предыдущее задание находится в состоянии На приёмке, следующее задание не будет выслано очередному исполнителю. Отправка произойдет только после перехода предыдущего задания в состояние Завершено.
- В экземплярах заданий, которые по завершении должны перейти в состояние На приёмке, также автоматически устанавливается флаг **Требуется приёмка**, а в поле Контролёр переносится значение либо из поля Контролёр (если он указан), либо из поля Ответственный исполнитель (если не указан контролёр, но указан ответственный исполнитель).

- Если одновременно указаны контролёр и ответственный исполнитель, контролёр будет проверять выполнение задания только ответственного исполнителя. Задания остальных исполнителей на приёмку после завершения не переходят.
- Если указан только Контролёр, а Ответственный исполнитель — нет, Контролёру на приёмку переходят задания всех исполнителей.
- Если указан только Ответственный исполнитель, задания всех исполнителей (кроме своего собственного) переходят ему на приёмку.

Изменение группы заданий перед отправкой

Автор группы заданий может изменить группу заданий до её отправки исполнителям.

1. Откройте сохранённую карточку группы заданий для редактирования.
2. Внесите требуемые изменения.
3. Нажмите кнопку **Сохранить** для сохранения изменений. Карточка будет открыта в режиме просмотра.

> Отправить группу заданий исполнителям можно из карточки, открытой для просмотра.

Отправка заданий исполнителям и мониторинг исполнения

Отправить группу заданий на исполнение

1. Откройте группу заданий в состоянии `Подготовка`.
2. На панели операций нажмите кнопку **Отправить на исполнение**. Состояние карточки будет `На исполнении`.

> Задания для исполнителей будут сформированы бизнес-процессом в течении некоторого времени.
>
> После формирования ссылки на задания будут добавлены в группу заданий, в список *Подчиненные задания*.
>
> Исполнители смогут увидеть свои задания в папке **Мои задания** › **Входящие**.

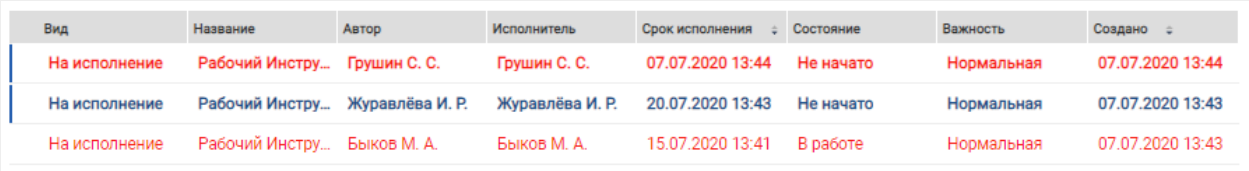
Отследить выполнение группы заданий

Отслеживать исполнение группы заданий можно двумя способами:

1. Из папки **Мои задания** › **Входящие**.

   Автор может отслеживать исполнение заданий группы из папки **Мои задания** › **Исходящие**, в которой отображаются задания и группы заданий автора. Для заместителя предназначена папка **Замещение** › **Исходящие**.

   Текущее состояние группы заданий и её заданий показано в колонке *Состояние*. При изменении задания (включая смену состояния) карточка будет выделена жирным шрифтом.

Вид	Название	Автор	Исполнитель	Срок исполнения	Состояние	Важность	Создано
На исполнение	Рабочий Инстру...	Грушин С. С.	Грушин С. С.	07.07.2020 13:44	Не начато	Нормальная	07.07.2020 13:44
На исполнение	Рабочий Инстру...	Журавлёва И. Р.	Журавлёва И. Р.	20.07.2020 13:43	Не начато	Нормальная	07.07.2020 13:43
На исполнение	Рабочий Инстру...	Быков М. А.	Быков М. А.	15.07.2020 13:41	В работе	Нормальная	07.07.2020 13:43

   Рисунок 231. Папка "Исходящие"

   Если задание было делегировано другому сотруднику, оно переместится в папку **Мои задания** › **Исходящие** › **Делегировано** у автора и в папку **Замещение** › **Исходящие** › **Делегировано** у его заместителя.

   завершённые задания будут показаны в папке **Мои задания** › **Исходящие** › **Завершено** и в папке **Замещение** › **Исходящие** › **Завершено** у заместителя.

2. Из раздела *Ход исполнения*, карточки Группа заданий:

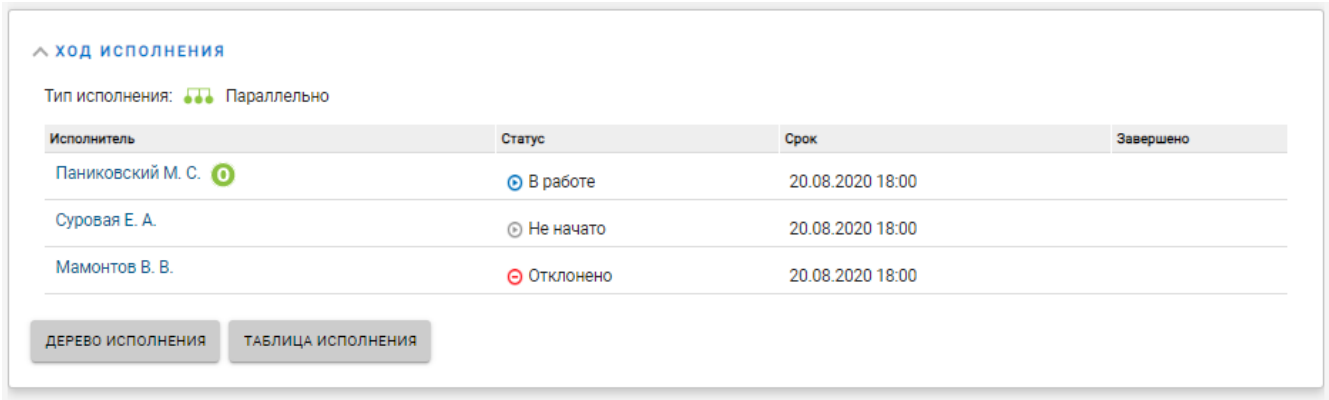
   

   Рисунок 232. Ход исполнения группы заданий

Данный раздел содержит:

- Тип исполнения задания (последовательно, параллельно).
- Таблицу с информацией об исполнении заданий группы: исполнители, статусы и сроки исполнения заданий.



Чтобы перейти к заданию, нажмите на ФИО исполнителя.

Исполнитель не отображается, если задание для него ещё не сформировано и не отправлено.

Кнопки:

- **Дерево исполнения**. Открывает дерево исполнения,
- **Таблица исполнения**. Открывает Таблица исполнения заданий.

Отозвать группу заданий

Правами на изменение состояния запущенной на исполнение карточки Группа заданий обладают пользователи с ролями *Автор* и *Ответственный исполнитель*.



Отозвать можно только незаблокированные задания. Также необходимо, чтобы все задания были закрыты.

Чтобы прекратить выполнение группы заданий:

1. Откройте карточку *Группа заданий* её автором.
2. На панели операций нажмите кнопку **Отозвать**. Состояние группы заданий и входящих в неё заданий изменится на `Отозвано`.



Решения исполнителей сбрасываются, если задание отозвано. Все остальные данные в карточке остаются без изменений: связанные документы остаются в карточке, предыдущие правки карточки остаются доступными в истории, сохраняются комментарии на соответствующей панели.

Исполнение заданий группы заданий

Работа с заданиями группы осуществляется аналогично работе с заданиями, созданными вне группы заданий. Инструкция по исполнению заданий приведена в пункте Работа с заданиями.

Удаление карточки Группа заданий

Карточка *Группа заданий* может быть удалена пользователем при выполнении следующих условий:

- Карточка задания разблокирована.
- Карточка находится в состоянии `Подготовка`.
- Пользователь является автором группы заданий.

> См. общую инструкцию по удалению в пункте Удаление карточки. При удалении группы заданий ссылки на неё будут удалены из других карточек.

Согласование карточек Документа

В данном разделе приведён пример согласования с использованием стандартного маршрута *Согласование с рецензированием и подписанием*, который включает этапы:

1. Согласование.
2. Консолидация.
3. Подписание.
4. Подведение итогов.

Согласование выполняется с помощью рассылаемых участникам согласования заданий видов: *На согласование*, *На консолидацию* и *На подписание*.

Состав маршрута согласования определяется при настройке согласования с использованием модуля *Конструктор согласований*. Пользователи, при наличии разрешения, могут вносить изменения в маршрут при создании и в процессе согласования.

Решения, принимаемые участниками согласования, могут носить положительный, условно-положительный и отрицательный характер, что определяется трактовкой решения в системе.

В процессе согласования отображаются локализованные решения. Локализация настраивается администратором модуля *Конструктор согласований*, см. подробнее.

После подведения итогов согласования, документ может быть согласован или возвращен на повторное согласование в новом цикле.

Усовершенствованное согласование

Усовершенствованное согласование изменяет поведение и вид некоторых элементов управления при работе с согласованием.

Запустить усовершенствованное согласование можно стандартным способом: нажатием на кнопку **На согласование** в карточке документа или при создании карточки. В списке выберите маршрут усовершенствованного согласования, настроенный в конструкторе согласований.



Рисунок 233. Управление усовершенствованным согласованием

Этапы могут выделяться цветом:

- Текущий этап выделяется сплошным синим цветом.

  Если активны несколько, другие активные этапы выделяются синим цветом со специальным фоном.

- Завершённые с отрицательной семантикой — красным.
- Завершённые с положительной семантикой — зелёным.
- Усовершенствованное согласование поддерживает повторение этапов — проходы. Номер прохода указывается рядом с названием этапа.



Завершённость этапа и семантика завершения определяются последним проходом этапа.

При использовании усовершенствованного согласования пользователю доступны следующие функции:

- Для согласования:
  - Завершить
  - Остановить
  - Отменить
- Для этапов:
  - Остановить этап.
  - Отредактировать этап.
  - Исключить этап из согласования.

Каждому отображаемому этапу могут соответствовать несколько экземпляров, но отображаются только данные последнего.

Цветовая индикация этапа зависит от результата последнего прохода. Если последний проход завершился положительно, этап будет отображаться как завершённый положительно вне зависимости от результата остальных проходов.

В названии этапов отображается номер прохода, начиная со второго прохода.

Подробнее о новой логике работы с усовершенствованным согласованием рассказывается в следующих пунктах:

- Ход согласования.
- Правила формирования версий файлов при согласовании.
- Консолидация.

Отправить документ на согласование и изменить маршрут согласования

Отправить документ на согласование

1. Создайте или откройте автором карточку согласуемого документа. Карточка должна быть в состоянии `Подготовка`.

- К документу должен быть приложен минимум один основной файл, который будет передан на согласование.
- В карточке должны быть указаны согласующие и, если нужно подписание, подписывающие. Список участников согласования можно изменить при отправке документа, если данная возможность была разрешена при настройке Docsvision.

2. Откройте панель подтверждения маршрута согласования нажатием кнопки **На согласование** на панели операций.

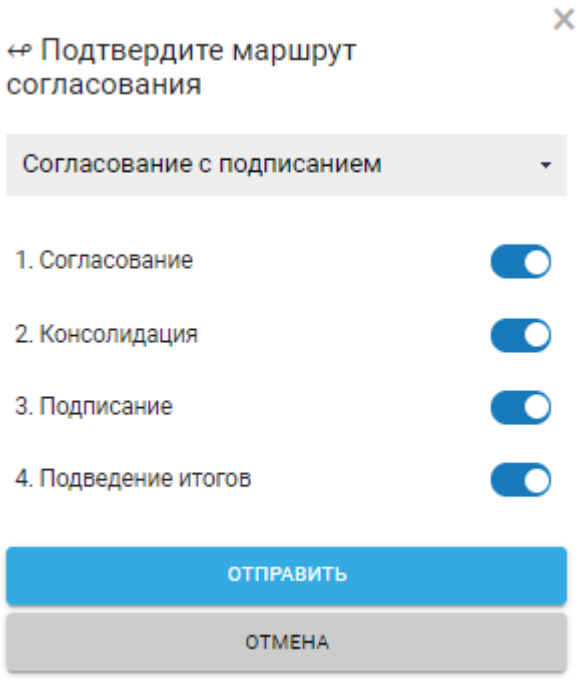


Рисунок 234. Панель подтверждения маршрута согласования в карточке "Документ"

Если по документу уже проводилось согласование, будет предложено предварительно очистить лист согласования от записей предыдущего согласования.

Если не очистить лист согласования, записи по новому согласованию будут добавляться к записям по предыдущему.

Вне зависимости от выбора результаты старого согласования не повлияют на результаты нового.

3. В списке маршрутов выберите маршрут, по которому пойдет согласование.

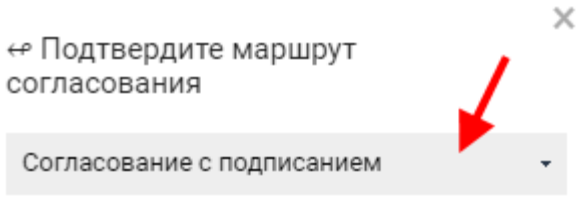


Рисунок 235. Маршрут согласования

> Маршруты для вида документов формируются при настройке Docsvision.

4. Если требуется, измените стандартные настройки маршрута согласования.
5. Нажмите кнопку **Отправить**, чтобы запустить согласование.
6. В появившемся окне нажмите кнопку **ОК**, чтобы подтвердить отправку.

> Будут сформированы задания для согласующих.
>
> Карточка Документа перейдет в состояние `На согласовании`
>
> Процесс отправки заданий может занять некоторое время.
>
> ⚠ На согласование поступают только *основные* файлы, приложенные к карточке. *Дополнительные* файлы не участвуют в согласовании, но будут также приложены к заданиям.

Изменить маршрут согласования

Пользователь может изменить настройки маршрута согласования, если данная возможность была разрешена при настройке согласования.

Исключить этап из согласования

Чтобы исключить этап из согласования, установите переключатель напротив этапа в положение **Выключено** (левое).

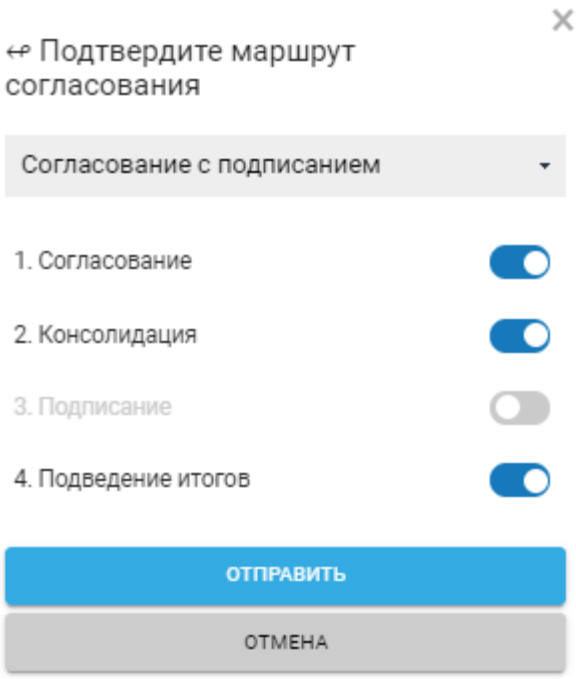


Рисунок 236. Маршрут с исключённым этапом "Подписание"



Данная возможность будет недоступна, если в настройках этапа снят флаг `Разрешить исключение этапа из маршрута`.



Рисунок 237. Этап согласования с заблокированным переключателем исключения этапа

Изменить состав согласующих

Пользователь может добавить, удалить или изменить порядок, а также временно исключить согласующих.

Чтобы изменить состав согласующих:

1. Откройте настройки этапа, выбрав этап из списка.

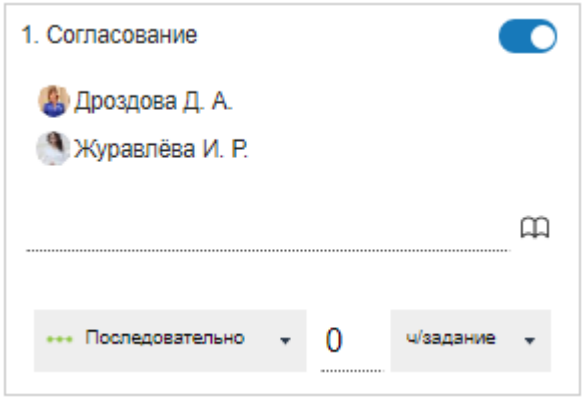


Рисунок 238. Настройка согласующих этапа "Согласование"

2. Измените состав согласующих как описано ниже:

Чтобы добавить согласующего:

a. Нажмите на кнопку .

b. Выберите согласующего из Справочника сотрудников.



Если выбрано подразделение, группа или роль, в согласование будут добавлены сотрудники, включённые в выбранный узел в *Справочнике сотрудников*.

Чтобы удалить или исключить согласующего:

a. Наведите курсор мыши на строку согласующего, чтобы отобразить функции удаления или исключения согласующего.

b. Чтобы временно исключить согласующего только из данного цикла согласования, снимите флаг в строке согласующего.

c. Удалите согласующего нажатием кнопки .

Чтобы изменить порядок согласующих:

a. Переместите участника выше или ниже, удерживая при этом левую кнопку мыши.

Особенности настройки списка согласующих:

Если в активном этапе не указаны согласующие, переключатель для исключения этапа будет выделен желтым. Укажите согласующих или исключите этап.

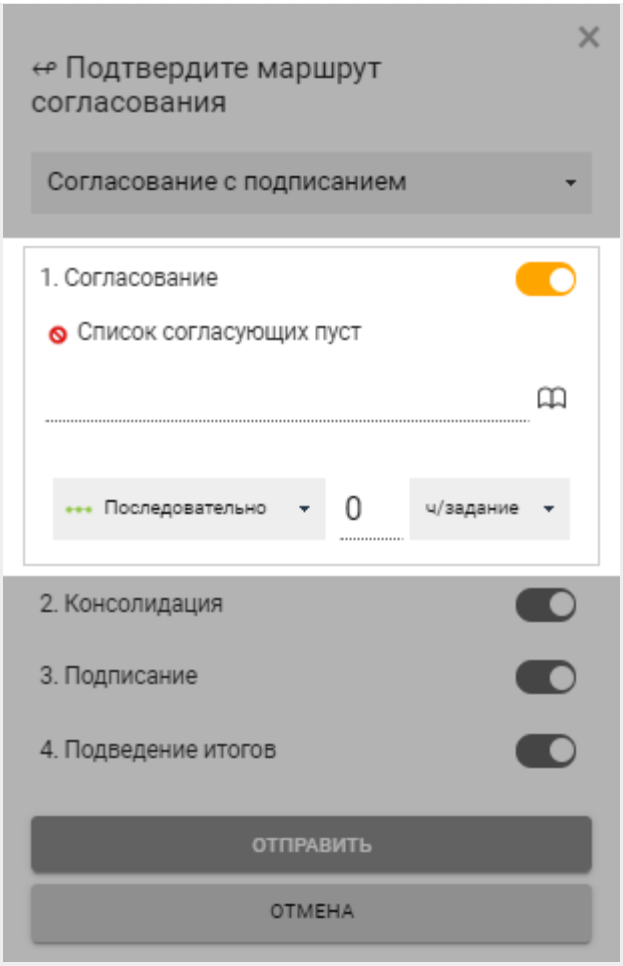


Рисунок 239. Этап "Согласования" без согласующих

Возможность изменить состав согласующих может быть заблокирована по двум причинам:

- Если в настройках этапа снят флаг `Разрешить изменение параметров этапа`. В этом случае блок с составом согласующих будет заштрихован:

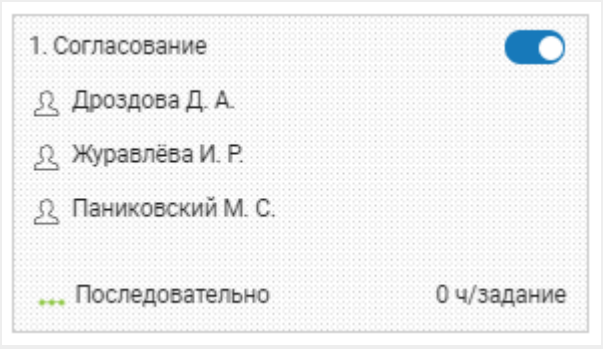


Рисунок 240. Этап согласования с заблокированным изменением настроек

- Если состав согласующих определяется с помощью бизнес-процесса. В данном случае этап будет помечен значком `A` (автоматически):

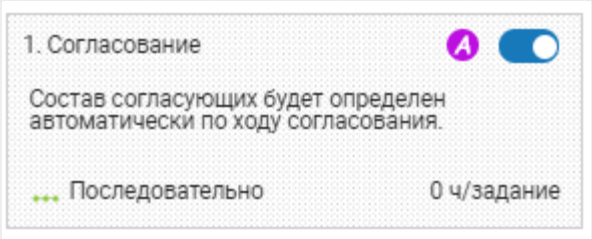


Рисунок 241. Этап согласования с автоматическим определением согласующих

Изменить тип маршрутизации

1. Откройте настройки этапа, выбрав его из списка.

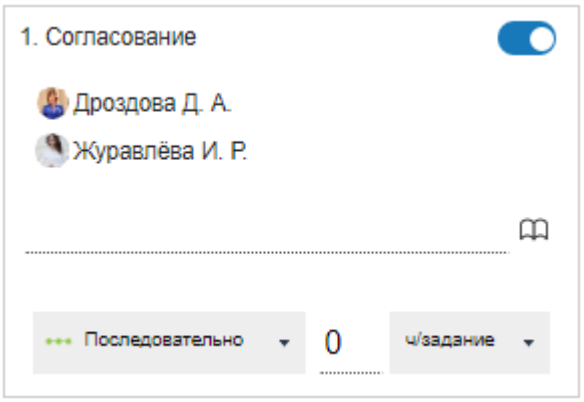


Рисунок 242. Настройка согласующих этапа Согласование

2. Выберите из списка новый тип маршрутизации заданий согласования:
   - ***Последовательно***. Участники согласования будут получать задания по очереди, следующий участник получит задание после выполнения задания предыдущим исполнителем.
   - ***Параллельно***. Все участники согласования получат задания одновременно.
   - ***Альтернативно***. Вместо отдельных заданий для каждого согласующего будет сформировано одно задание на всех выбранных согласующих. Исполнителем задания становится сотрудник, который первым возьмет задание в работу. У остальных согласующих задание будет удалено.



Данная возможность будет недоступна, если в настройках этапа снят флаг `Разрешить исключение этапа из маршрута`.

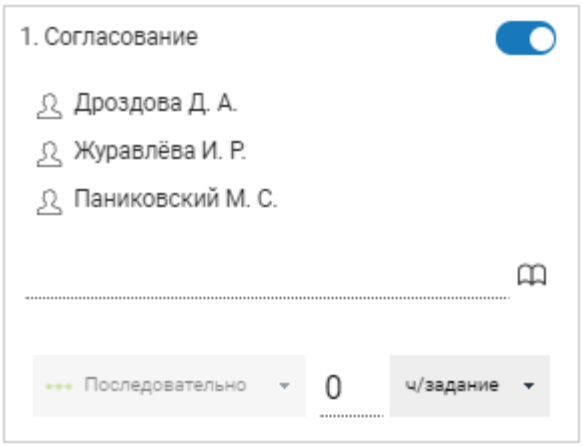


Рисунок 243. Этап согласования с заблокированной возможностью изменения типа маршрутизации

Ознакомиться с ходом согласования

1. Откройте карточку согласуемого документа.
2. Перейдите к разделу *Согласование* или нажмите кнопку **Ход согласования** на панели операций.

Ход согласования

У пользователей, имеющих права на просмотр согласования, на вкладке **Ход согласования** отображается список прошедших циклов согласования или текущий цикл, список прошедших этапов согласования или текущий этап. Для каждого этапа отображается решение и список новых версий файла, добавленных исполнителями заданий на согласование.

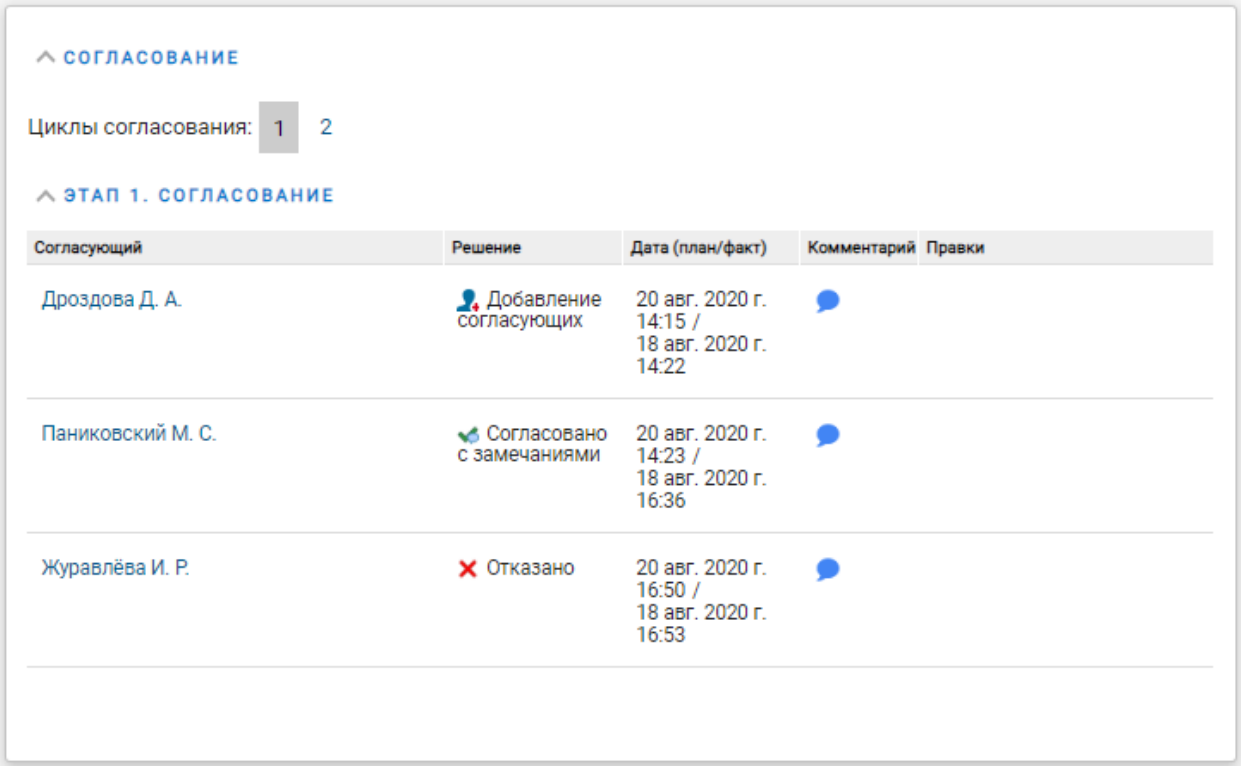


Рисунок 244. Раздел "Ход согласования" в карточке согласуемого Документа

Согласующий

В столбце указывается информация об исполнителе задания. Если задание было делегировано, в столбце указывается информация о назначенном (исходном) исполнителе и делегате.

Чтобы перейти к заданию согласования, нажмите на ссылку в ФИО исполнителя в столбце *Согласующий*. Для подчинённых заданий согласования ФИО не является ссылкой.

Отображаемое ФИО формирует БП согласования.

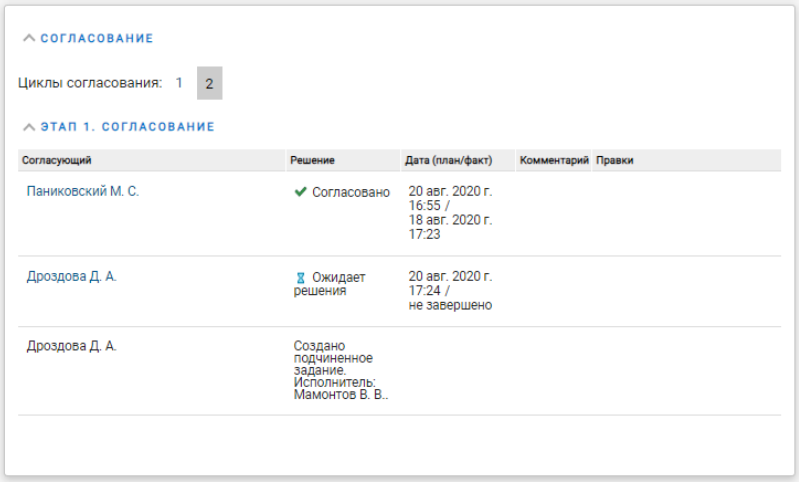


Рисунок 245. Подчинённое задание согласования в ходе согласования

Решение

В столбце *Решение* отображается решение исполнителя по своему заданию.

Если задание согласования было делегировано, в столбце будет указано `Делегировано [Кому делегировано]`

Дата (план/факт)	В следующем столбце отображаются планируемая и фактическая даты выполнения задания согласования. Если дата отсутствует, в столбце будет указано: не задана, не завершена. Для подчинённого задания дата не отображается.
Комментарий	Если по заданию был оставлен комментарий или файл комментария, в строке задания будет отображаться значок . при нажатии на значок, открывается комментарий. Если задание было делегировано, в столбце также отображается комментарий, указанный при делегировании.

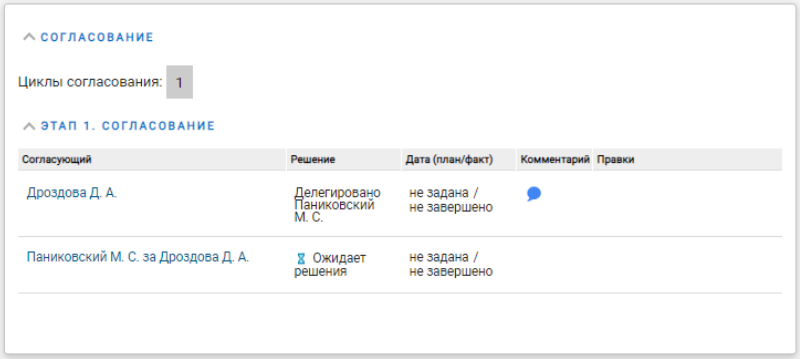


Рисунок 246. Записи о делегировании задания в Ходе согласования

3. Чтобы ознакомиться с результатами прошедших циклов согласования, нажмите на ссылку с номером нужного цикла.
4. Чтобы обновить журнал хода согласования, нажмите на появившуюся кнопку обновления, отображаемую в строке циклов согласования при наведении.

Если нужно распечатать результаты согласования, нажмите кнопку **Лист согласования** на панели операций.



Рисунок 247. Отправка результатов согласования на печать

Ход усовершенствованного согласования

Для усовершенствованных маршрутов в ходе согласования отображается проход этапа.

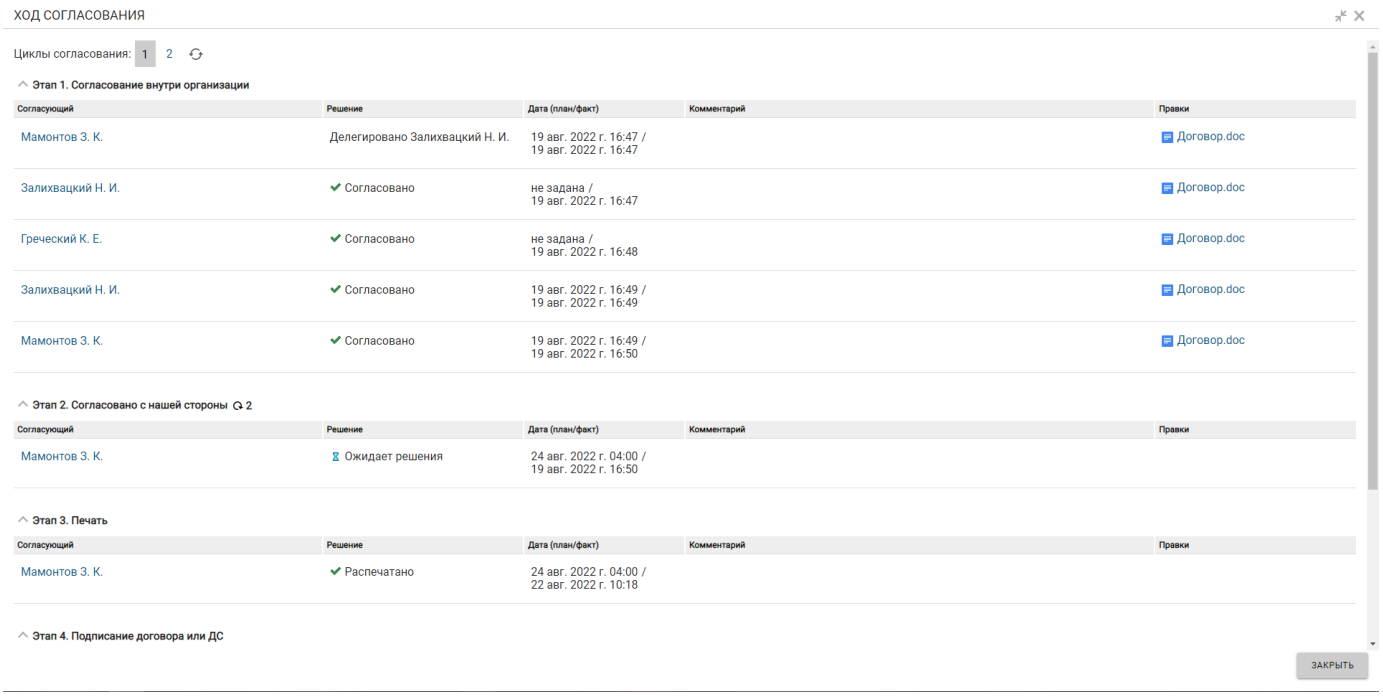


Рисунок 248. Этап 2, проход 2

Проход отображается рядом с названием этапа в виде иконки 2.

Остальная информация в ходе усовершенствованного согласования совпадает с информацией в ходе обычного согласования.

Полный ход согласования

По умолчанию в ходе согласования отображается текущее согласование, это ***Стандартный*** режим. При необходимости можно настроить ***Полный*** способ отображения хода согласования в свойствах ЭУ "Ход согласования". При ***Полном*** отображении согласования появляется возможность посмотреть все согласования карточки, включая отмененные.

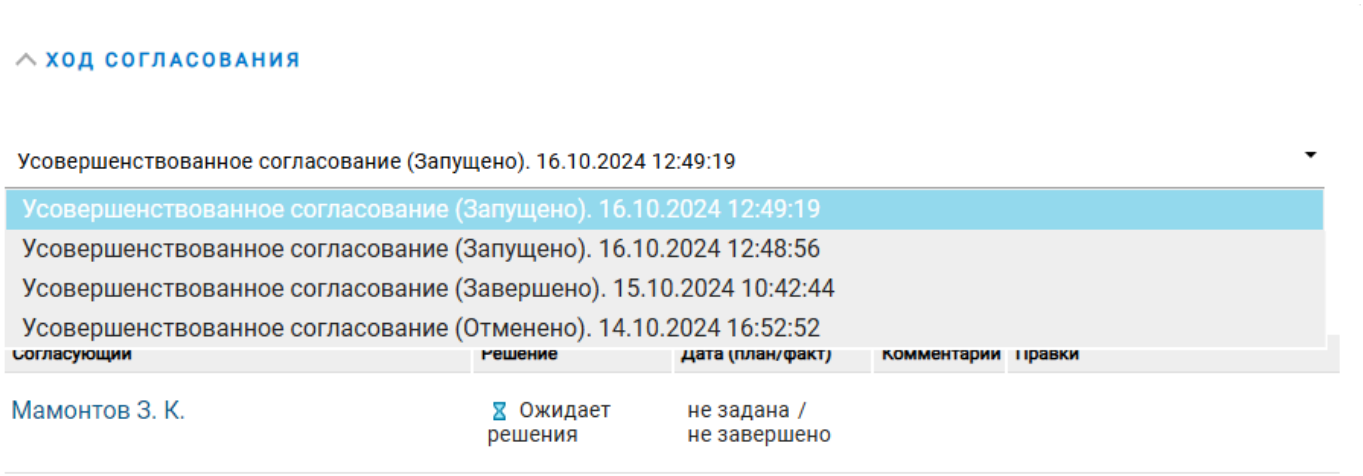


Рисунок 249. Выбор согласования при включенном режиме "Полный ход согласования"

Если выбран режим "Полный", в ЭУ "Ход согласования" отображается поле с раскрывающимся списком, в котором содержатся все согласования, отсортированные по дате старта согласования по убыванию, чем старше согласование, тем ниже его позиция в списке. По умолчанию выбрано самое последнее согласование. Согласование не отображается в списке, если у пользователя нет прав на его просмотр.

Имя согласования в списке формируется по принципу: `Название согласования (Состояние). Дата и время создания согласования`. Например, "Согласование с юристами (Согласовано) 16.12.2024 12:31:56".

Управление активным согласованием

Изменить параметров активного согласования

Автор может изменить параметры активного согласования из карточки

согласуемого документа.

Чтобы изменить параметры активного согласования:

1. Откройте карточку согласуемого документа.
2. Откройте панель управления активным согласованием нажатием кнопки **Управление** на панели операций.



Кнопка **Управление** может быть недоступна, если у пользователя отсутствуют права на управление согласованием.

3. Выберите этап согласования, настройки которого требуется изменить.
4. Измените настройки этапа или этапов.
5. Нажмите кнопку **Сохранить**.

Если изменённый этап ещё не выполнялся, то изменения будут применены в текущем цикле согласования, иначе — в следующем цикле.

Принудительно прекратить текущий этап

Автор может принудительно завершить текущий этап согласования из карточки согласуемого документа.

Чтобы принудительно завершить текущий этап согласования:

1. Откройте панель управления активным согласованием нажатием кнопки **Управление** на панели операций.



Кнопка **Управление** может быть недоступна, если у пользователя отсутствуют права на управление согласованием.

2. Нажмите кнопку **Завершить этап** в строке текущего этапа согласования (выделяется голубым цветом).

Для завершённого этапа будет установлена отрицательная семантика, все незавершённые задания участников согласования будут отозваны. Будет запущен следующий этап согласования в соответствии с маршрутом.

Приостановить согласование

Автор может на время приостановить активное согласование из карточки согласуемого документа.

1. Откройте панель управления активным согласованием нажатием кнопки **Управление** на панели операций.



Кнопка **Управление** может быть недоступна, если у пользователя отсутствуют права на управление согласованием.

2. Нажмите кнопку **Пауза**.

Процесс согласования будет приостановлен, рассылка новых заданий будет прекращена. Карточка будет доступна в режиме чтения для согласующих.

Чтобы вновь запустить приостановленное согласование, нажмите кнопку **Запустить**, которая появится на панели управления согласованием после приостановки.

Отменить согласование

Автор может отменить активное согласование из карточки согласуемого документа.

Чтобы отменить согласование:

1. Откройте панель управления активным согласованием нажатием кнопки **Управление** на панели операций.



Кнопка **Управление** может быть недоступна, если у пользователя отсутствуют права на управление согласованием.

2. Нажмите кнопку **Отменить**.

Карточка перейдет в состояние `Подготовка`. Все незавершённые задания текущего согласования будут отозваны.

Принудительно завершить согласование

Автор может принудительно завершить процесс активного согласования из

карточки согласуемого документа.

Чтобы принудительно завершить согласование:

1. Откройте панель управления активным согласованием нажатием кнопки **Управление** на панели операций.

   Кнопка **Управление** может быть недоступна, если у пользователя отсутствуют права на управление согласованием.

2. Нажмите кнопку **Завершить**.

   Карточка перейдет в финальное состояние, указанное администратором в маршруте. Все незавершённые задания согласования будут отозваны.

Задание на согласование

Согласование документа выполняется с помощью задания *На согласование*.

1. Перейдите в папку **Мои задания** › **Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение** › **Входящие**

2. Откройте полученное задание с описанием *Согласование документов* в состоянии `Не начато`.

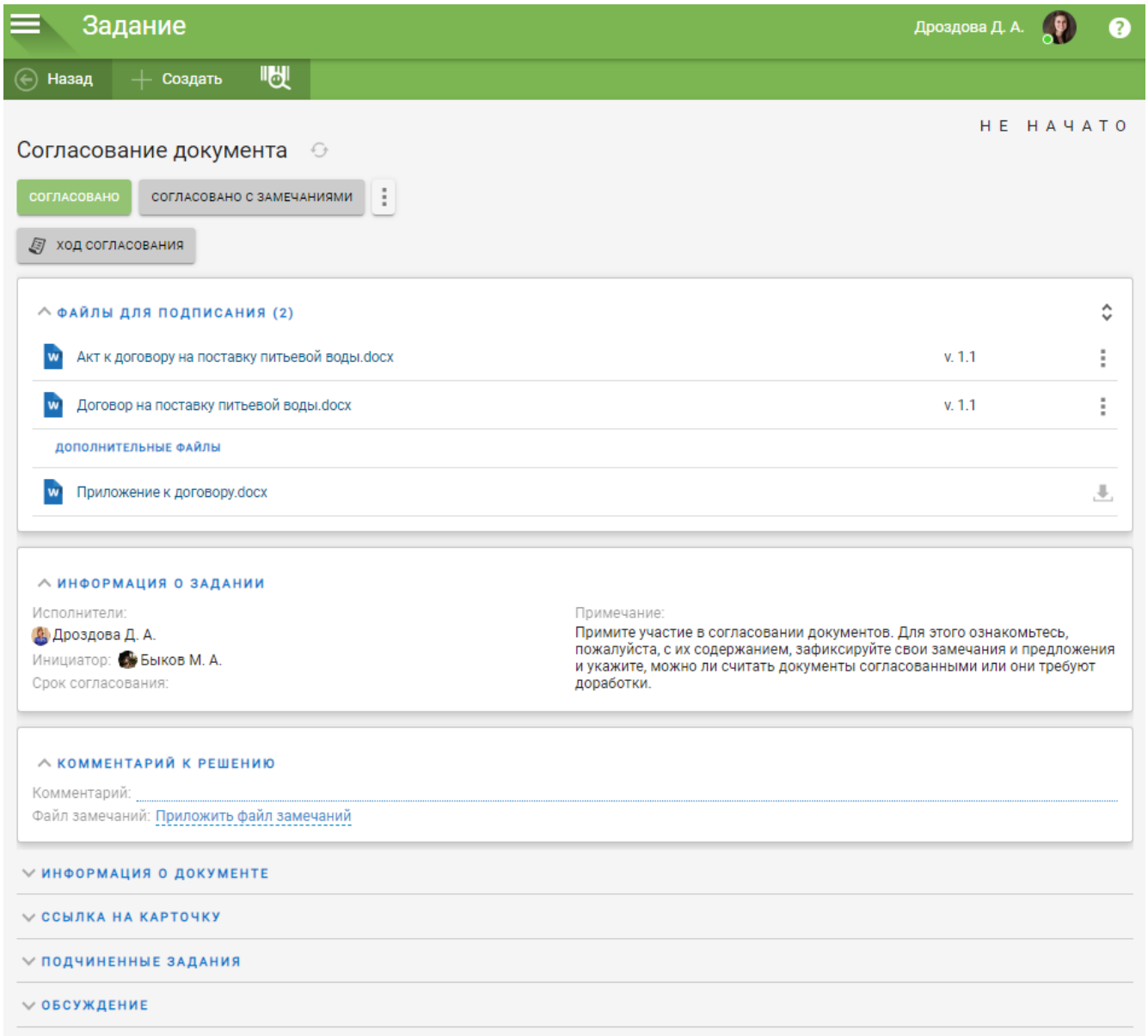


Рисунок 250. Карточка задания На согласование, открытая у согласующего

Задание содержит разделы:

Файлы для согласования (N) или Файлы для подписания (N)

Согласуемые основные файлы документа. Приложенные дополнительные файлы документа в согласовании не участвуют.

Информация о задании

Содержит общую информацию о задании согласования, включая срок выполнения.

Комментарий к решению

Комментарий согласующего по принятому решению.

Информация о документе

Содержит общую информацию о карточке Документ, поступившей на согласование.

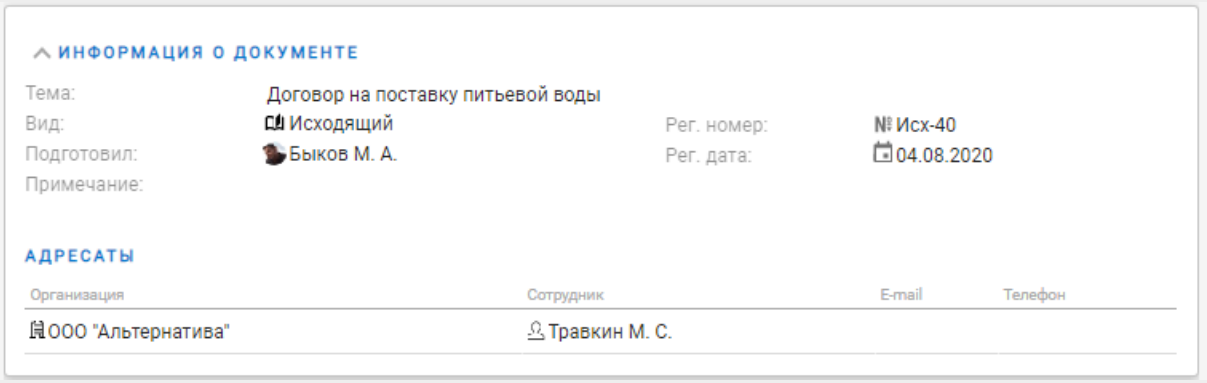


Рисунок 251. Информация о документе

Ссылка на карточку

Содержит ссылку для перехода к карточке согласуемого документа.

Подчинённые задания

Содержит ссылки на задания, созданные по данному заданию согласования.

Обсуждение

Комментарии по данному заданию. Подробнее в пункте Обсуждение согласования.

3. Ознакомьтесь с составом согласуемых и дополнительных файлов.
4. Добавьте собственную версию согласуемого файла.
5. Добавьте комментарий к версии.



Особенности формирования версий файлов приведены в пункте Правила формирования версий файлов при согласовании.

6. Если требуется, создайте подчиненные задания согласования для других пользователей.

Подчиненные задания позволяют получить от других сотрудников информацию, необходимую для принятия решения по согласованию. Работа с подчиненными заданиями рассмотрена в пункте Подчиненные

задания согласования.

7. Примите решение по согласованию. Нажмите соответствующую кнопку принятия решения на панели операций.

Согласовано	Положительное решение. Документ будет согласован и передан следующему исполнителю.
Согласовано с замечаниями	Условно-положительное решение. Документ будет согласован и направлен следующему участнику согласования с комментарием к решению.
Отказано	Отрицательное решение. Документ не будет согласован, карточка будет отправлена следующему исполнителю, либо произойдет переход к новому циклу согласования.

Также могут быть доступны дополнительные варианты:

Делегировать	Делегирует задание другому исполнителю.
Добавить согласующих	Добавляет новых участников в согласование. Инструкция по работе с дополнительными согласующими приведена в Дополнительные согласующие.

Результат согласования будет отображен в компоненте *Ход согласования*.

8. Введите причину, по которой было принято соответствующее решение в появившемся окне, если принятое решение по согласованию предполагает ввод комментария.
9. Нажмите кнопку **Подтвердить**, чтобы добавить комментарий.



Рисунок 252. Добавление комментария к решению

Если согласующим был добавлен комментарий к загруженной версии согласуемого файла, этот комментарий будет автоматически скопирован в поле комментария к решению.

Задание будет завершено. Описание решения, принятого согласующим, появится в верхней части экрана.

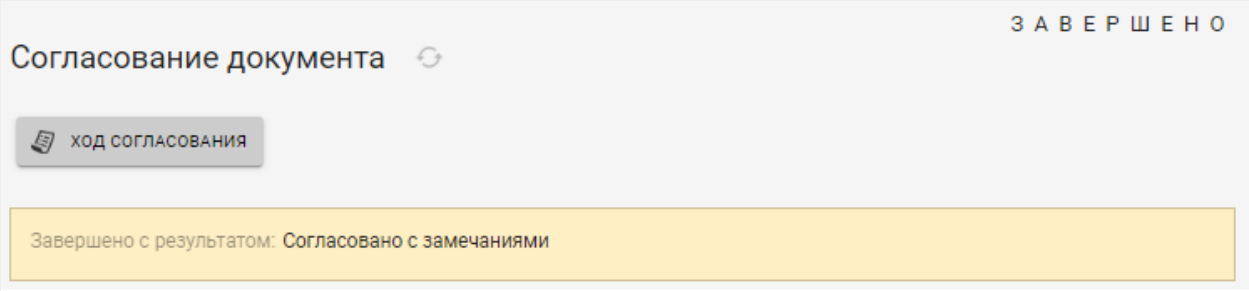


Рисунок 253. Интерфейс карточки завершённого задания на согласование

Работа с файлами согласования

Ознакомиться с составом согласуемых и дополнительных файлов

Согласуемыми файлами являются основные файлы согласуемой карточки Документ. Дополнительные файлы не участвуют в процессе согласования и могут быть приложены к документу только для ознакомления.

Чтобы ознакомиться с файлами в задании на согласование:

1. Откройте карточку полученного задания на согласование.
2. Ознакомьтесь с текущими версиями согласуемых файлов, представленных в разделе *Файлы для согласования*.

   При необходимости ознакомьтесь с версиями, созданными в процессе согласования, и выполните доступные операции с файлами, поступившими на согласование: просмотрите список доступных версий, создайте собственную

версию или комментарий к версии.

Файлы, представленные в списке *Дополнительные файлы*, являются дополнительными файлами согласуемого документа, и не участвуют в согласовании, но могут быть полезны при принятии решения.

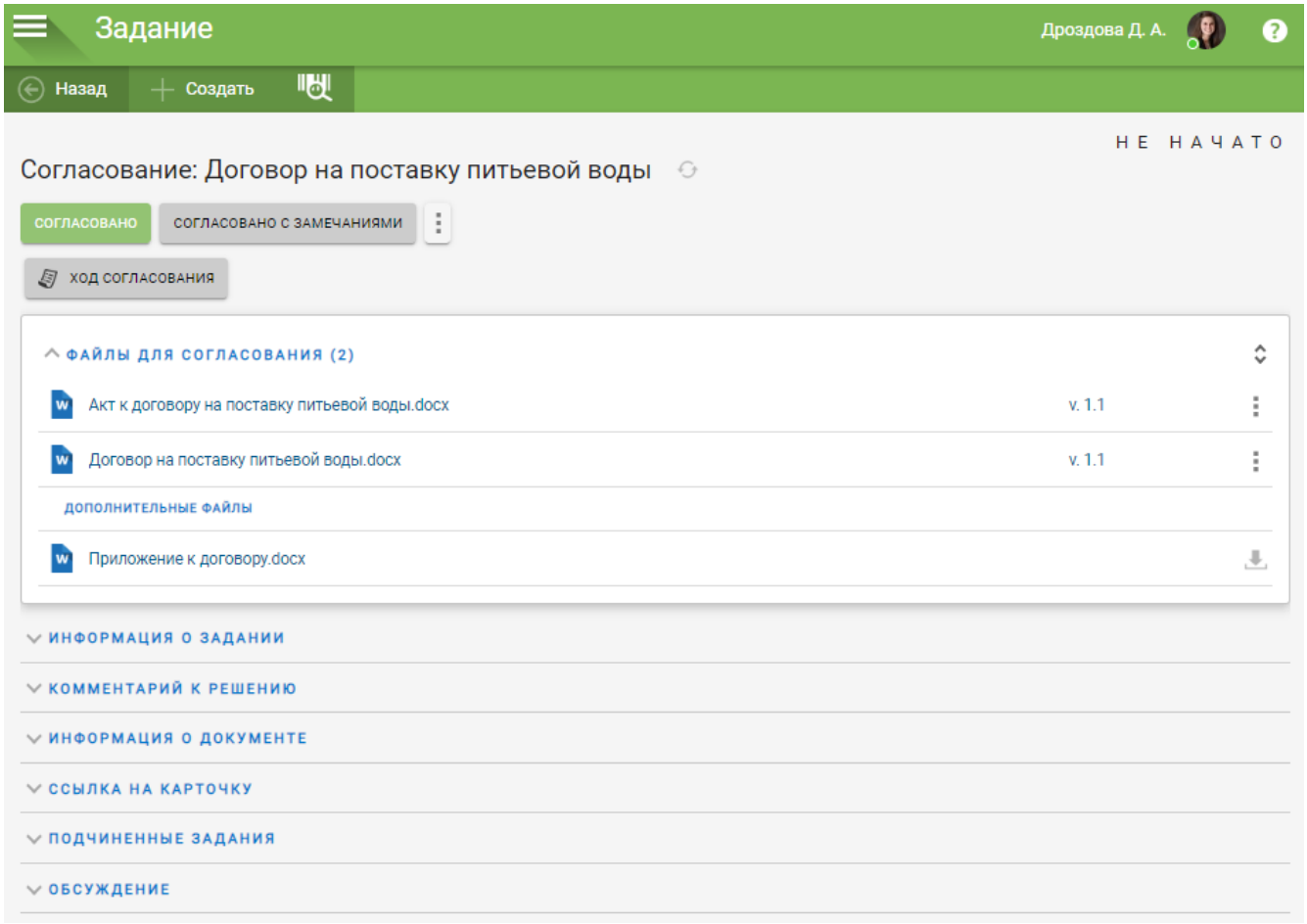


Рисунок 254. Список файлов для согласования в карточке задания "На согласование"

Правила формирования версий файлов при согласовании

В процессе согласования к документу или заданию могут быть приложены собственные версии согласующих, добавленные загрузкой или изменённые через приложение, а также версии, созданные автоматически бизнес-процессом согласования. Версии формируются и учитываются отдельно для каждого приложенного файла.

При создании нескольких ветвлений версий файла на параллельных этапах, если на параллельных этапах не производилась консолидация, на последующий общий этап будет передана версия файла до ветвления. При этом, если между последовательными этапами есть прямая связь, между

ними передаётся изменённая версия файла.

Обычное согласование

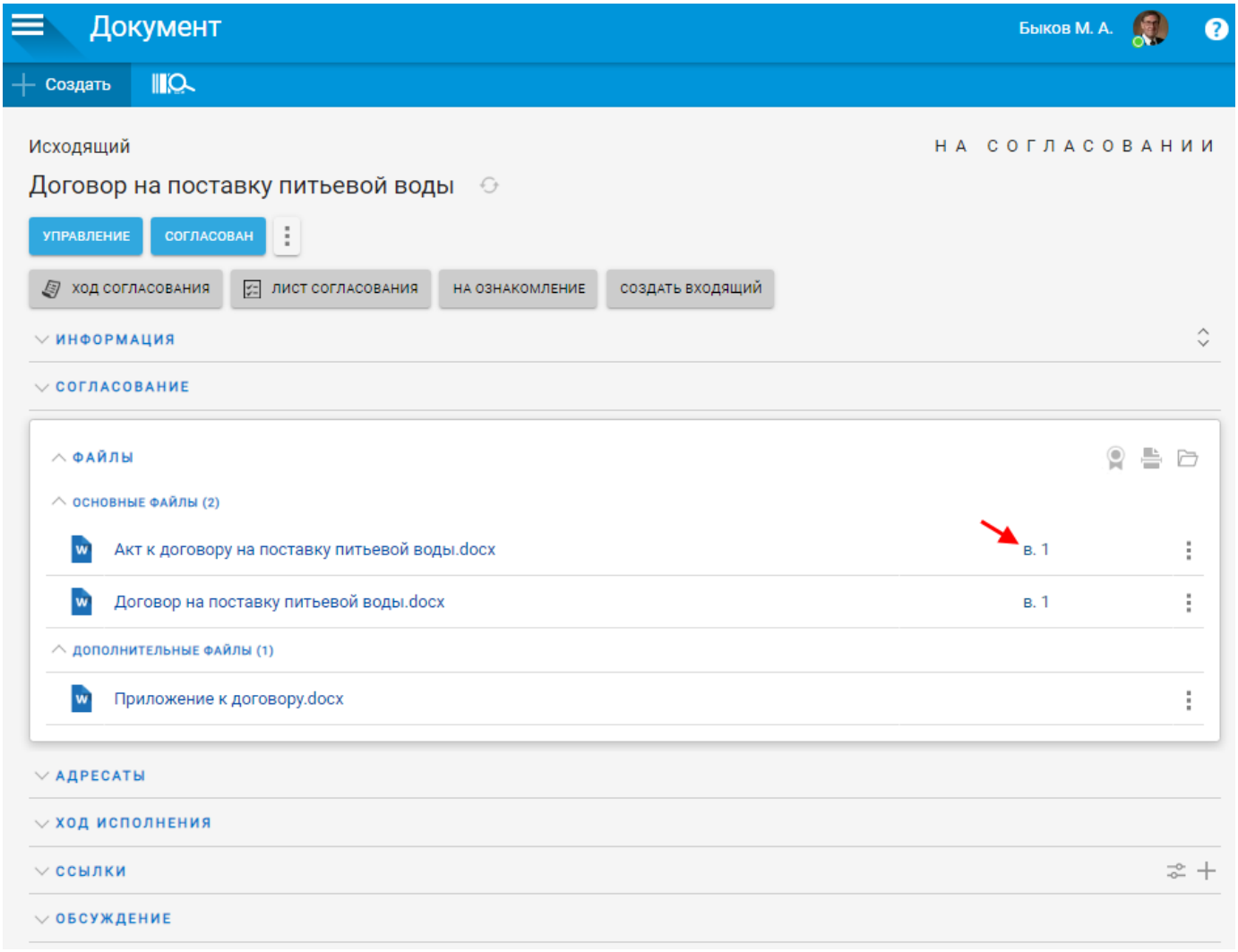


Рисунок 255. Создание исходной версии документа

При первичной загрузке *основного* файла к карточке документа версии файла будет присвоен номер `в.1`. Наименование следующих версий может отличаться в зависимости от вида маршрута согласования.

После отправки документа по маршруту согласования, исполнители первого этапа получат для согласования копию версии файла `в.1` с названием `в.1.1`. Вторая цифра после точки будет соответствовать номеру этапа (в данном случае — первому этапу).

Если на каком-либо этапе согласующий создаст собственную версию, её номер будет сформирован как `в.1.1.N`. `N` — порядковый номер версии, созданной данным согласующим. Первая созданная версия будет `в.1.1.1`, следующая —

в.1.1.2 и так далее.

Если в этапе участвуют два согласующих, и второй согласующий создаст собственную версию, её номер версии будет образован как номер версии предыдущего согласующего + 1. Например, в.1.1.2.

При переходе к следующему этапу согласования данного маршрута, согласующему будет отправлена копия версии файла в.1 с названием в.1.2.N. Соответственно, нумерация версий согласующих будет выглядеть как: в.1.2.N.

Аналогичным образом будут формироваться версии всех последующих этапов согласования.

На втором цикле согласования исполнители первого этапа получат для согласования версию с названием в.2.1. Первая цифра соответствует номеру цикла, вторая — номеру этапа. Аналогичным образом будут формироваться версии всех последующих циклов до наступления первого этапа консолидации.

Усовершенствованное согласование

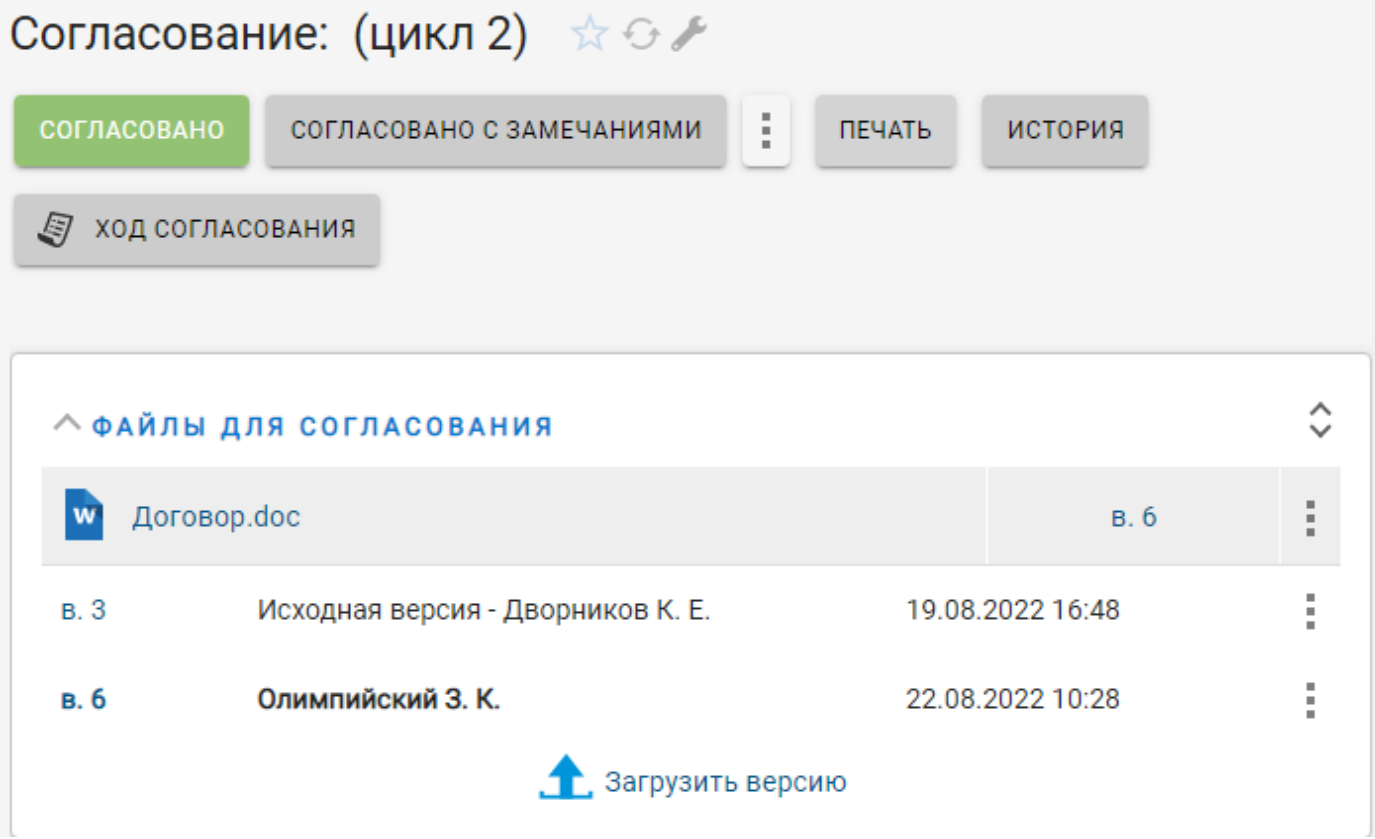


Рисунок 256. Версии файлов согласования

При усовершенствованном согласовании номера версий назначаются по порядку. Номер для начальной версии не отображается. Новая версия файла

имеет номер в. 1 и создаётся, только если он был изменён. Следующая версия будет иметь вид в. 2 и далее по порядку.

Номер версии не зависит от этапа или цикла. В списке версий отображаются все версии, созданные из этапа или прохода этого задания и исходная версия этапа. Версии упорядочены в списке по порядку добавления. Первой отображается исходная версия этапа.

Для всех версий доступны следующие операции:

- **Открыть**
- **Скачать**
- **Комментировать**

Операции для версий, созданных из этого задания:

- **Удалить**.
- **Редактировать**.

Просмотреть версии согласуемых файлов

Чтобы посмотреть список доступных версий файлов на согласовании:

1. Откройте карточку задания на согласование.
2. Перейдите к разделу *Файлы для согласования*.

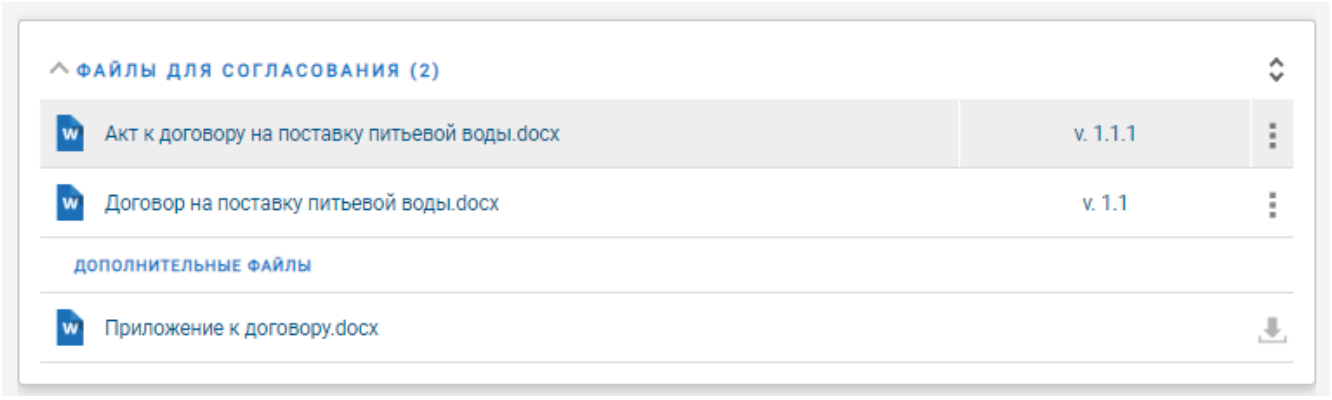


Рисунок 257. Файлы для согласования

3. Нажмите на согласуемый файл, чтобы раскрыть список доступных версий файла.

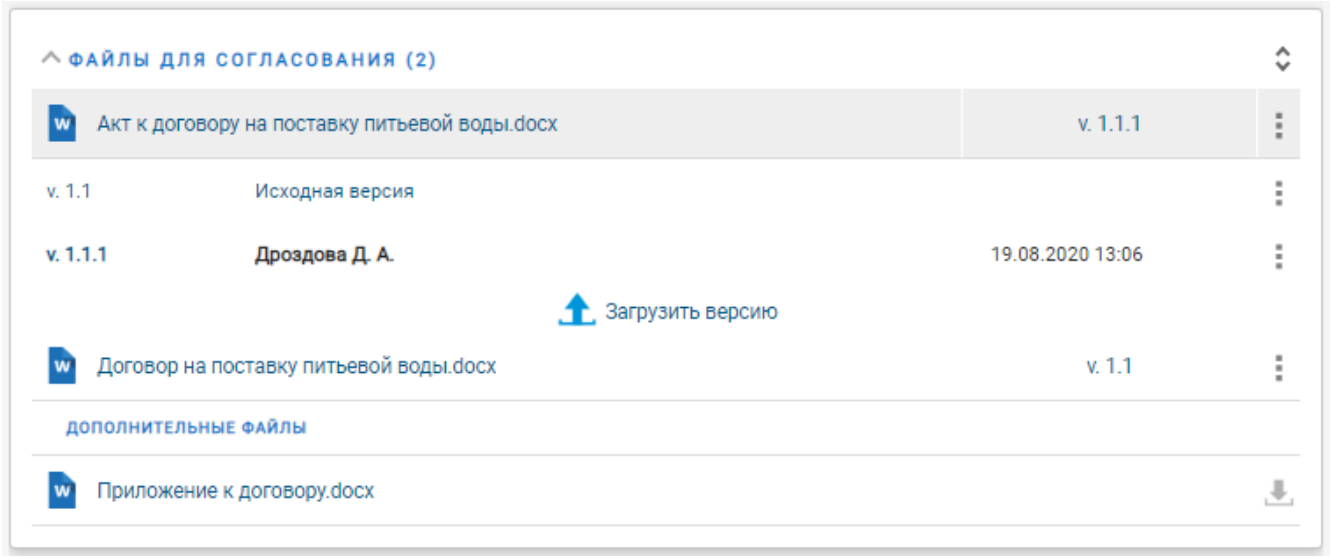


Рисунок 258. Текущая версия файла выделяется жирным шрифтом

> Подробнее о формировании номера версии см. выше. В описании версии будет отображаться дата и время добавления версии.

Создать новую версию согласуемого файла

Для создания собственной версии согласуемого файла, выполните следующие действия:

1. Откройте карточку согласуемого документа.
2. В разделе **Файлы для согласования** выберите версию, на основе которой требуется создать собственную версию файла.
3. Нажмите ⋮ справа от названия файла и выберите команду **Скачать версию**, чтобы сохранить файл на компьютер.
4. Создайте новую версию файла одним из доступных способов:
   - Нажмите кнопку **Скачать версию**, чтобы загрузить версию себе на компьютер и создать новую версию на основе существующей.

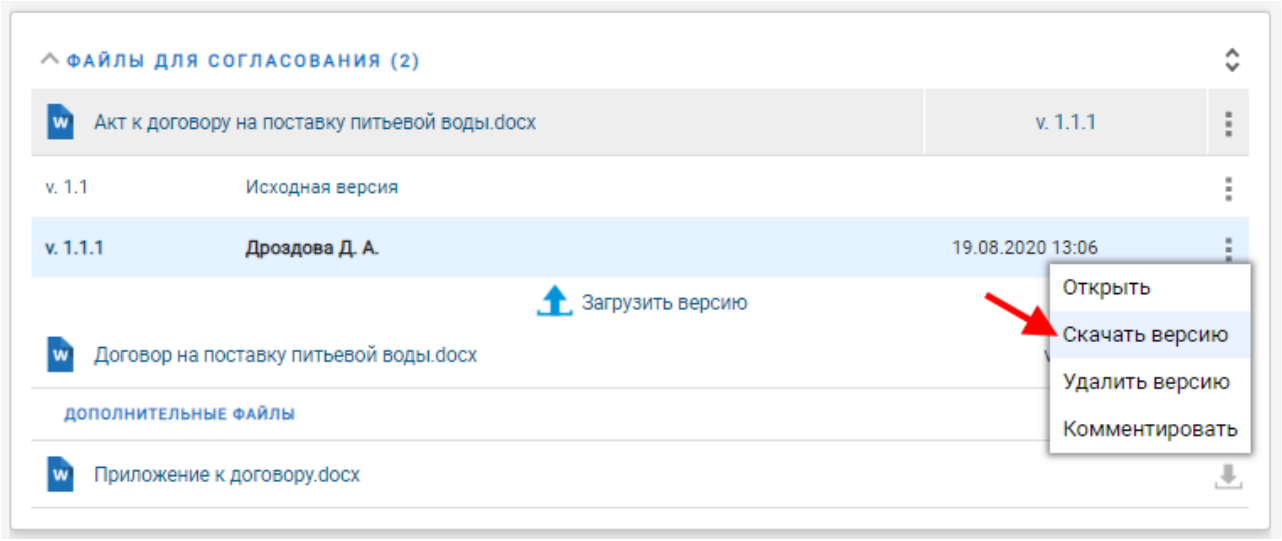


Рисунок 259. Команда скачивания версии в контекстном меню версии файла

- Нажмите кнопку **Открыть**, чтобы открыть версию на редактирование в офисном редакторе. Потребуется DVWebTool или DVSupService, см. подробнее "Установка и запуск программы DVWebTool" или "Установка и запуск службы DVSupService".

5. Внесите требуемые изменения в файл.
6. Сохраните изменения в файле.
   - Если версия была скачана на устройство, после сохранения изменений нажмите кнопку ⋮ справа от названия (не версии) согласуемого файла и выберите команду **Загрузить версию**. Затем выберите загружаемый файл с файловой системы.

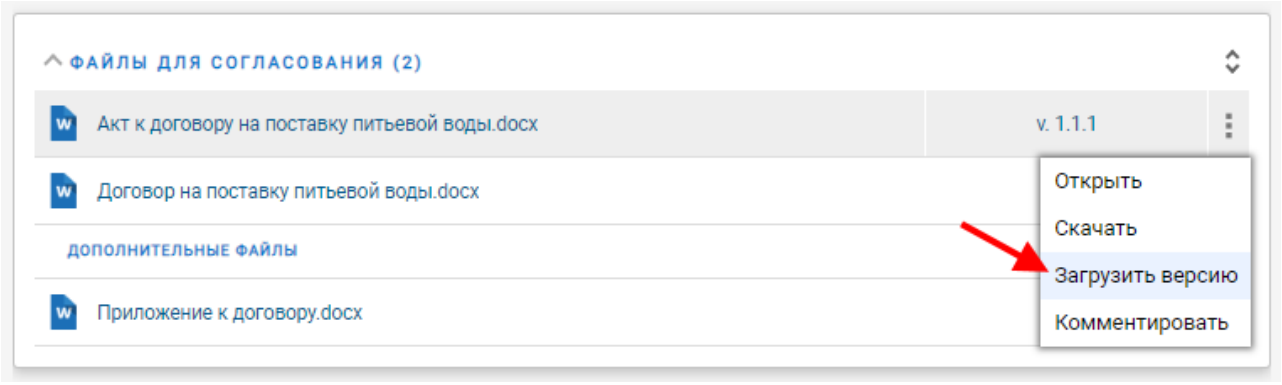


Рисунок 260. Команда загрузки версии в контекстном меню файла для согласования

> Название загружаемого файла должно совпадать с названием файла, для которого загружается новая версия.
>
> Новая версия файла будет добавлена в список.

- Если версия была открыта в офисном редакторе, новая версия будет автоматически создана в задании сразу после закрытия окна редактора.

Оставить комментарий к версии файла

Согласующий может оставить комментарий к версии файла. Редактировать или удалять свой комментарий можно до появления нового комментария или до принятия решения по согласованию.

Чтобы добавить комментарий:

1. Откройте карточку задания на согласование.
2. В разделе **Файлы для согласования** нажмите на номер версии и выберите версию файла, к которой требуется добавить комментарий.
3. Выполните любое из действий:
   - Нажмите на иконку справа от названия файла. Иконка скрыта, если по данной версии нет комментариев;
   - Нажмите ⋮ **> Комментировать**, чтобы открыть окно добавления комментария.

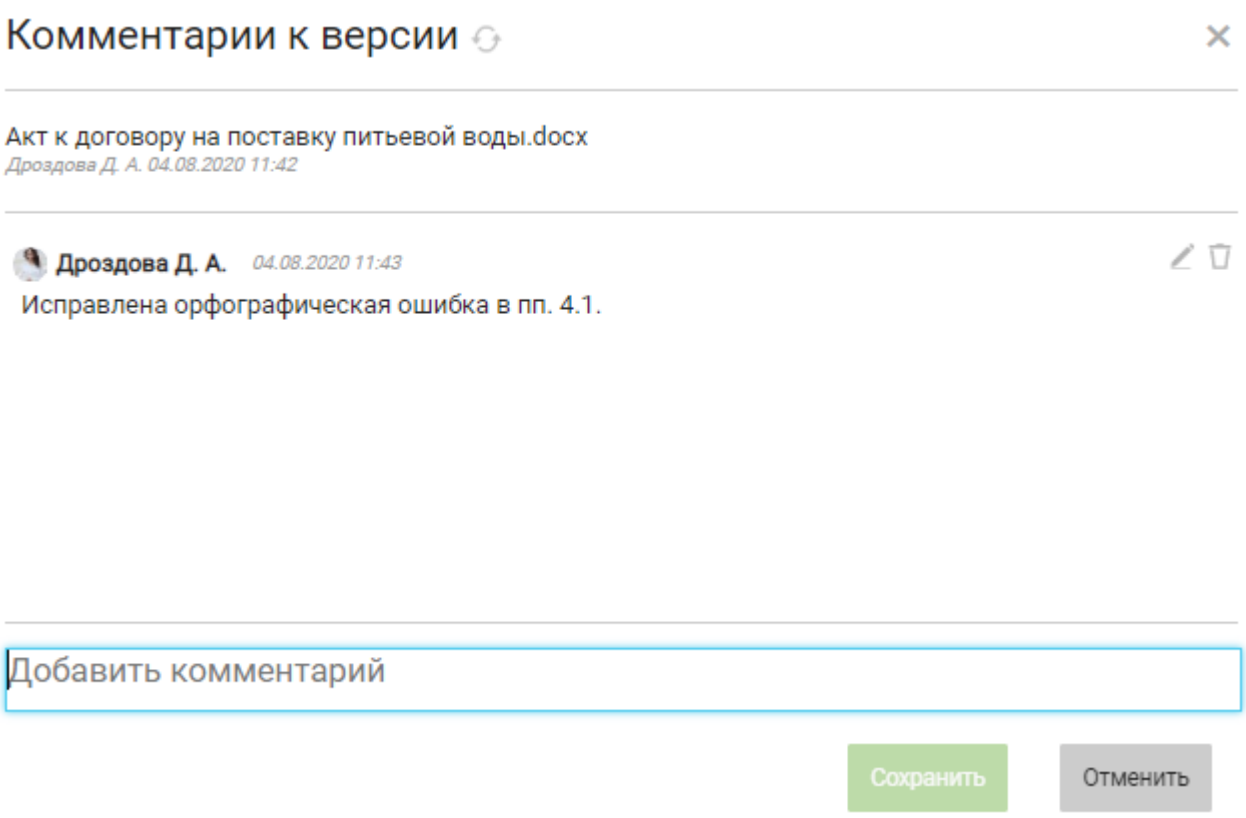


Рисунок 261. Добавление комментария к версии согласуемого файла

4. Введите текст комментария и нажмите **Сохранить**, чтобы добавить

комментарий к версии файла.

> Чтобы посмотреть комментарии, нажмите . Справа от кнопки отображается количество комментариев.

Задание на консолидацию

Основная задача консолидатора — сформировать итоговую версию согласуемого документа из всех версий, добавленных согласующими. В дальнейшем итоговая версия может быть отправлена на окончательное подписание, или быть использована в качестве стартовой в новом цикле согласования. Без выполнения консолидации или копирования какой-либо версии в итоговую согласующие или подписывающие в заданиях на согласование или на подписание всегда будут получать копию файла, отправленного при старте согласования.

Консолидатор может выполнять следующие действия в карточке задания на консолидацию:

- Ознакомиться с составом согласуемых и дополнительных файлов.
- Работать с версиями согласуемых файлов — создавать и редактировать собственную версию файла, просматривать версии остальных участников согласования.
- Создать объединенную версию.
- Работать с областью комментариев на протяжении всей работы с заданием.
- Ознакомиться с ходом согласования, решениями других участников.
- Завершить задание путем выбора одного из доступных вариантов решений.

При работе с файлами *Консолидатор* дополнительно может назначить любую версию в дереве итоговой, для подведения итогов определённой стадии согласования.

Чтобы получить документ на консолидацию:

1. Перейдите в папку **Мои задания** › **Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение** › **Входящие**

2. Откройте полученное задание *На консолидацию* с состоянием `Не начато`.

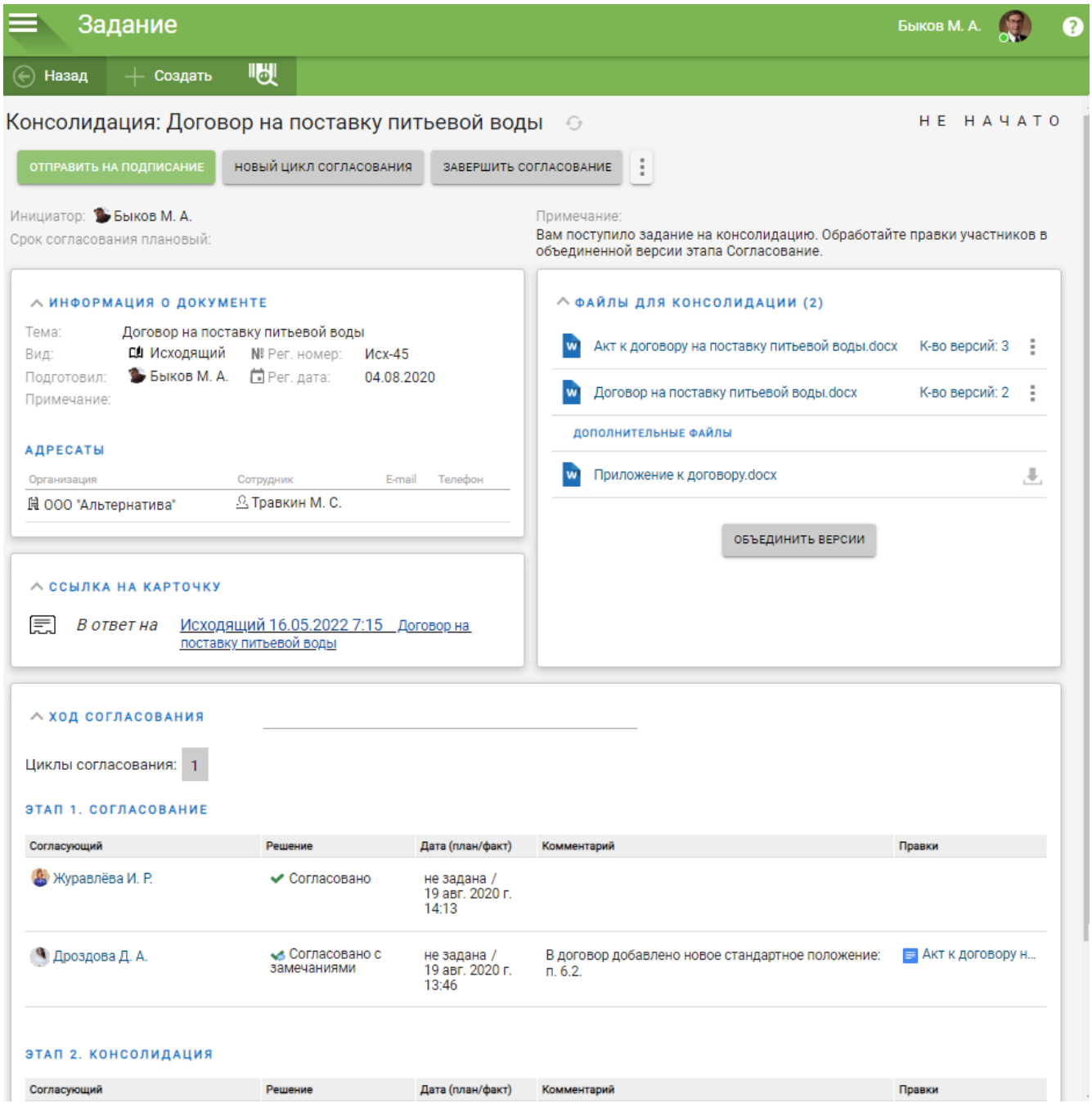


Рисунок 262. Задание на консолидацию

Описание разделов карточки задания:

Информация о документе

Содержит общую информацию о карточке Документ, файлы которой поступили на консолидацию после согласования.

Ссылка на карточку

Содержит ссылку для перехода к карточке согласуемого документа.

Файлы для консолидации (N)

Согласуемые файлы. Консолидатору доступны версии:

Исходная версия цикла

Версия, поступившая в цикл согласования с предыдущего цикла, или исходная версия, если текущий цикл согласования является первым.

Версии участников

Итоговая версия этапа

Версия для использования в качестве результата выполнения консолидации (выделена жирным шрифтом). .В полученном задании итоговой версией:

- Будет считаться исходная, если на предыдущем этапе не было правок.
- Будет считаться версия согласующего, если были правки только от одного согласующего.
- Будет считаться версия последнего согласующего, если предыдущий этап был последовательный.
- Будет считаться исходная версия предыдущего этапа, если предыдущий этап был параллельный.

Ход согласования

Содержит информацию об этапах согласования: список согласующих и принятые ими решения, комментарии к решениям и добавленные версии файлов. К комментарию может быть приложен файл замечаний.

Обсуждение

Комментарии по данному заданию. Подробности в пункте Обсуждение согласования.

3. Выполните требуемые операции с версиями согласуемых файлов.

Операции с версиями согласуемых файлов

- Ознакомьтесь с составом согласуемых и дополнительных файлов.

- Добавьте собственную версию согласуемого файла.
- Создайте объединенную версию.
- Добавьте комментарий к версии.

После изучения версий согласующих, консолидатор должен определить версию, которую планируется использовать либо как итоговую по результатам согласования, либо как стартовую в следующем этапе/цикле согласования. В качестве итоговой консолидатор может выбрать версию любого из согласующих на любом этапе/цикле, либо собственную версию.

В дальнейшем процессе согласования будет использоваться итоговая версия. Эту версию консолидатор может отправить на подписание.

Если инициатор отправляет документ на новый цикл согласования, итоговая версия, выбранная при консолидации, становится исходной для нового цикла согласования.

Консолидатор в списке всех версий увидит выделенную итоговую версию предыдущего цикла, которая является текущей на данный момент. В области *Основные файлы* будет отображаться текущая версия, являющаяся итоговой версией предыдущего цикла. Инициатор консолидирует замечания согласующих и помечает отредактированную версию как текущую. Она становится итоговой.

В списке версий основного документа итоговая версия выделена жирным шрифтом.

В области *Основные файлы* отображается текущая версия основного документа, а не последняя прикрепленная.

4. Выберите итоговую версию.

   В разделе *Файлы для консолидации* выберите версию.

   Итоговая версия будет считаться результатом выполнения консолидации. При переходе к следующему этапу маршрута согласования, данная версия будет скопирована в задания согласующих.

5. Примите решение по консолидации. Для этого нажмите соответствующую кнопку принятия решения.

Отправить на подписание	Документ будет отправлен на этап подписания.
Новый цикл согласования	Будет запущен новый цикл согласования.
Завершить согласование	Согласование будет принудительно завершено. Документ перейдет в состояние `Согласован` или `Подписан` с сохранением результатов текущего согласования при отсутствии отрицательных решений по согласованию. Незавершённые задания исполнителей будут отозваны.
Отменить согласование	Согласование будет отменено. Документ перейдет в состояние `Подготавливается`, если данное состояние определено для вида карточки в *Конструкторе состояний*. Незавершённые задания исполнителей будут отозваны.

Также могут быть доступны дополнительные варианты:

Делегировать	Делегирует задание другому исполнителю.
Добавить согласующих	Добавляет новых участников в согласование. Инструкция по работе с механизмом дополнительных согласующих приведена в Дополнительные согласующие.

Результат согласования будет отображен в блоке *Ход согласования*.

6. Если принятое решение по согласованию предполагает ввод комментария, в открывшемся окне введите причину, по которой было принято соответствующее решение, и нажмите кнопку **Подтвердить**.

Задание перейдет в состояние Завершено.

Добавленный к решению комментарий будет отображаться в блоке *Ход согласования*.

Усовершенствованное согласование

При усовершенствованном согласовании внешний вид задания консолидации не изменяется.

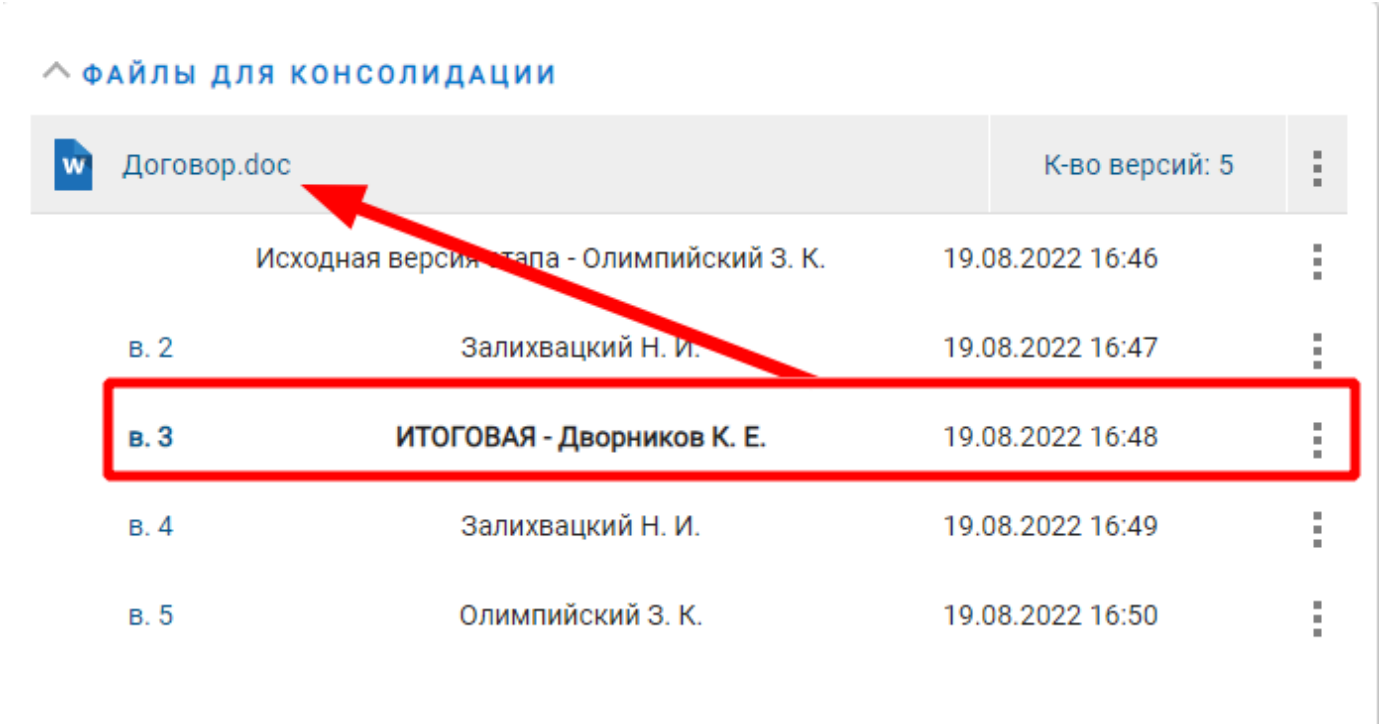


Рисунок 263. Итоговая версия в задании на консолидацию

В заголовке отображается итоговая версия.

Список версий выводится в следующем порядке:

- Исходная версия этапа консолидации. Строчка добавляется, только если она не является одной из версий из очереди консолидации.
- Версии из очереди консолидации в порядке их создания.
- Версии, добавленные на этапе консолидации.

Итоговая версия этапа выделяется жирным шрифтом, в описании к ней указан префикс **ИТОГОВАЯ -**.

Все версии отображаются с порядковыми номерами, см. подробнее Правила формирования версий файлов при усовершенствованном согласовании.

Загруженная версия назначается итоговой.

Набор операций, доступных для всех версий:

- **Скачать** и **Комментировать**.
- **Открыть**. При редактировании создаётся новая версия консолидатора.
- **Назначить итоговой**. Для всех версий, кроме итоговой. При выполнении данной операции выбранная становится итоговой, новая версия не создаётся.

Операции для версий, созданных из этого задания:

- Удалить.

Создание объединенной версии

Консолидатор может создать версию, включающую изменения других согласующих, воспользовавшись функцией автоматического объединения версий — функцией автоконсолидации.

Требования для использования автоконсолидации:

1. Автоконсолидация версий выполняется при помощи "Р7-Офис. Сервер документов".
2. Объединяемые файлы должны быть одного из допустимых форматов: `.docx`, `.docm`, `.doc`, `.rtf`, `.dotx`, `.dotm`, `.dot`, `.odt`.

Чтобы создать объединенную версию:

1. Получите задание на консолидацию.
2. Нажмите кнопку **Объединить версии**.

   Будет запущен процесс создания объединённой версии, которая будет назначена итоговой.

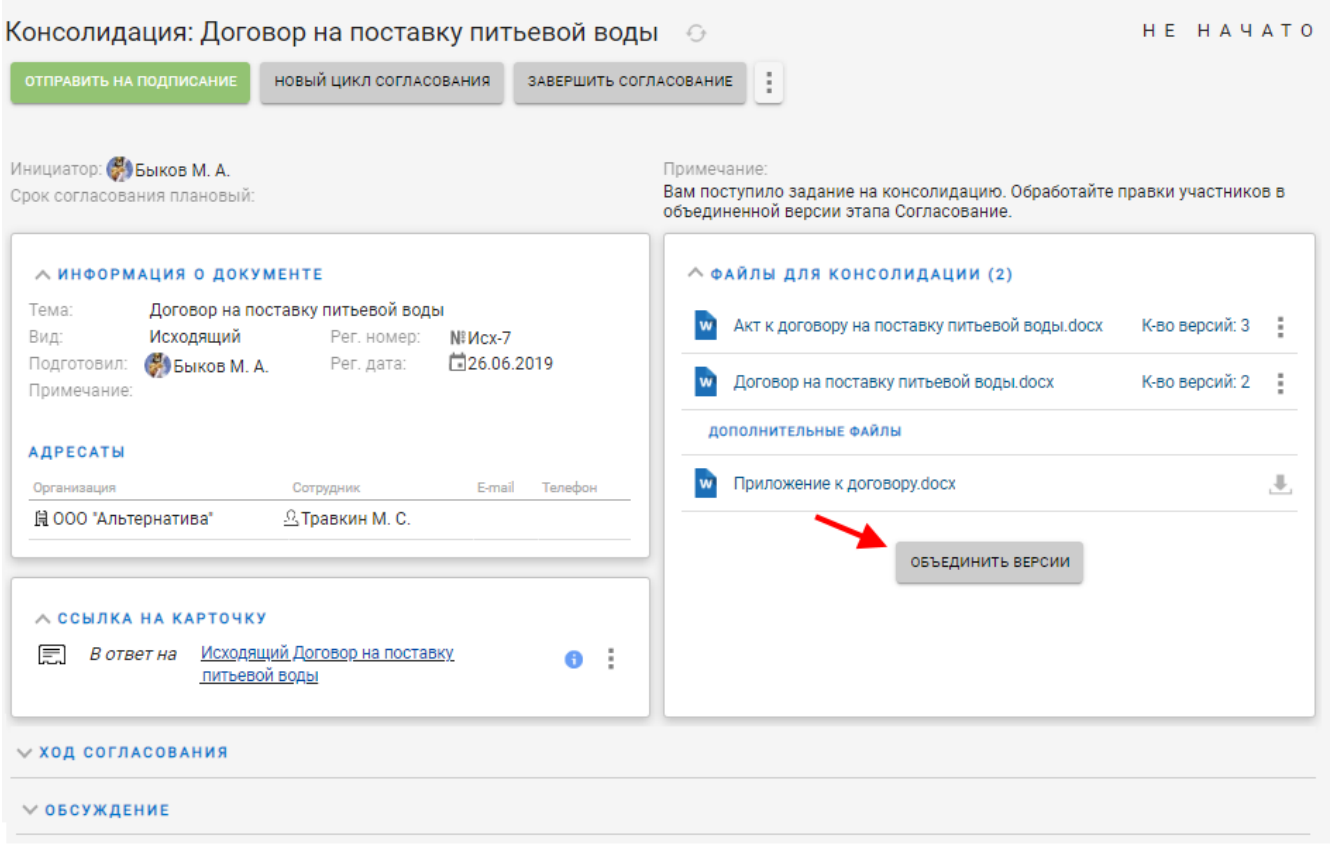


Рисунок 264. Кнопка "Объединить версии" в задании консолидации

На время выполнения операции пользовательский интерфейс блокируется. Блокировка будет снята, когда появится сообщение о завершении объединения.

Задание на подписание

Этап подписания предназначен для добавления электронной подписи к согласуемому документу. На документ может быть наложена простая, усиленная и квалифицированная электронная подпись. Возможность наложения того или иного вида подписи регулируется настройками системы Docsvision и наличием необходимых программных средств на компьютере пользователя.

На этапе подписания не предполагается добавление новых версий файлов.

Чтобы получить задание на подписание, выполните следующие действия:

1. Перейдите в папку **Мои задания** › **Входящие**

   Если вы являетесь заместителем, прейдите в папку **Замещение** › **Входящие**

2. Откройте полученное задание *На подписание* с состоянием `Не начато`.

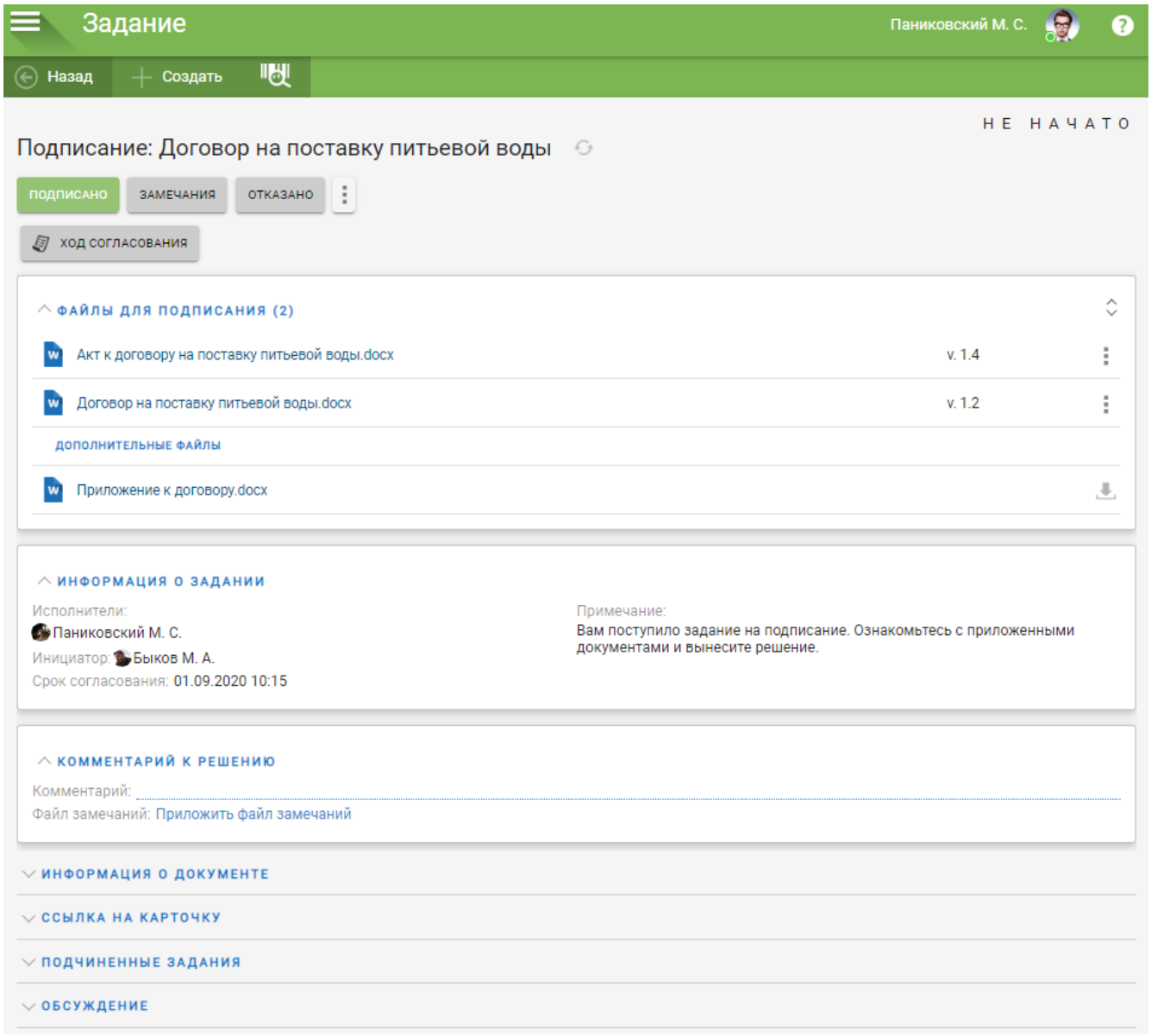


Рисунок 265. Пример открытого задания на подписание

Задание содержит разделы:

Файлы для согласования (N) или Файлы для подписания (N)

Согласуемые основные файлы документа. Приложенные дополнительные файлы документа в согласовании не участвуют.

Информация о задании

Содержит общую информацию о задании согласования, включая срок выполнения.

Комментарий к решению

Комментарий согласующего по принятому решению.

Информация о документе

Содержит общую информацию о карточке Документ, поступившей на согласование.

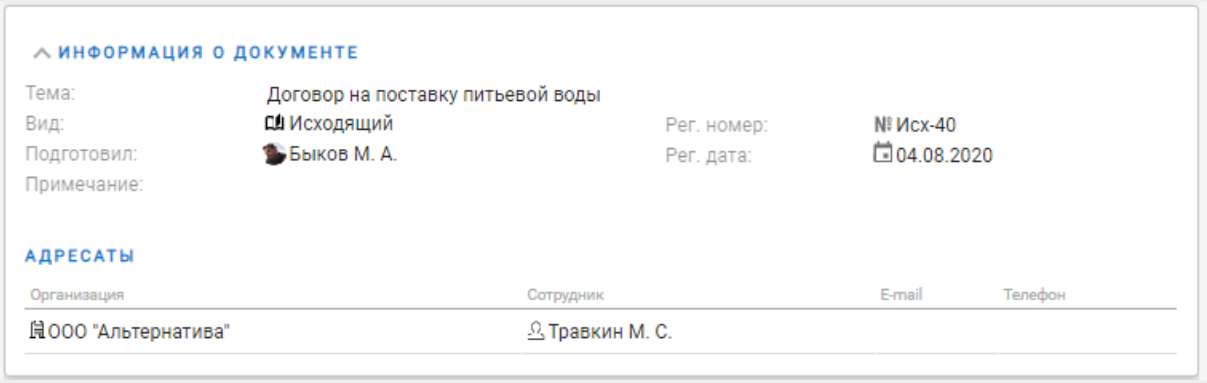


Рисунок 266. Информация о документе

Ссылка на карточку

Содержит ссылку для перехода к карточке согласуемого документа.

Подчинённые задания

Содержит ссылки на задания, созданные по данному заданию согласования.

Обсуждение

Комментарии по данному заданию. Подробнее в пункте Обсуждение согласования.

3. Выполните требуемые операции с версиями подписываемых файлов.

- Ознакомьтесь с составом подписываемых и дополнительных файлов;
- Добавьте комментарий к версии.

Подробнее о работе с версиями см. раздел: Правила формирования версий файлов при согласовании.

1. Примите решение по согласованию нажатием соответствующей кнопки принятия решения.

Подписано	Положительное решение. Документ будет подписан и передан следующему исполнителю согласно маршруту. электронная подпись будет добавлена ко всем основным файлам карточки. Если в настройках этапа был указан тип метки, он будет включен в подпись. В истории появится запись о добавленной подписи.
Замечания	Условно-положительное решение. Документ будет подписан и направлен следующему участнику согласования с комментарием к решению.
Отказано	Отрицательное решение. Документ не будет подписан, согласование будет отменено.

2. Если принятое решение по подписанию предполагает ввод комментария, в открывшемся окне введите причину решения.
3. Подтвердите ввод комментария нажатием кнопки **Подтвердить** и выберите способ наложения электронной подписи. При наличии доступного сертификата может быть наложена усиленная электронная подпись, при отсутствии сертификата — простая подпись (если она не запрещена настройками этапа).



Для формирования усиленной подписи с использованием сертификатов ЭП должны быть установлены компоненты КриптоПро.

Задание будет завершено. Добавленный к решению комментарий будет отображаться в области **Ход согласования**

После завершения согласования итоговая версия файла будет перенесена в согласуемый документ вместе с подписями данной версии. До завершения согласования подписи файла в Журнале подписей не отображаются. Подписи атрибутов карточки Документ отображаются сразу после подписания и не зависят от итоговой версии файла.



Ограничение на размер подписываемого файла в Docsvision отсутствует, однако при подписании файлов в Web-клиенте могут быть ограничения, связанные с плагином КриптоПро,

см. подробнее в разделе "Подписание карточек типа "Документ"".

Работа с ЭП на iOS и Android через интерфейс Web-клиента не поддерживается без специального мобильного приложения.

Делегировать задание

Задания на согласование, подписание и консолидацию документов можно делегировать другим исполнителям также, как и обычные задания. Возможность повторного делегирования задания определяется его настройками.

Чтобы делегировать задание согласования:

1. Откройте карточку задания.
2. Откройте панель делегирования нажатием на кнопку **Делегировать** на панели операций.

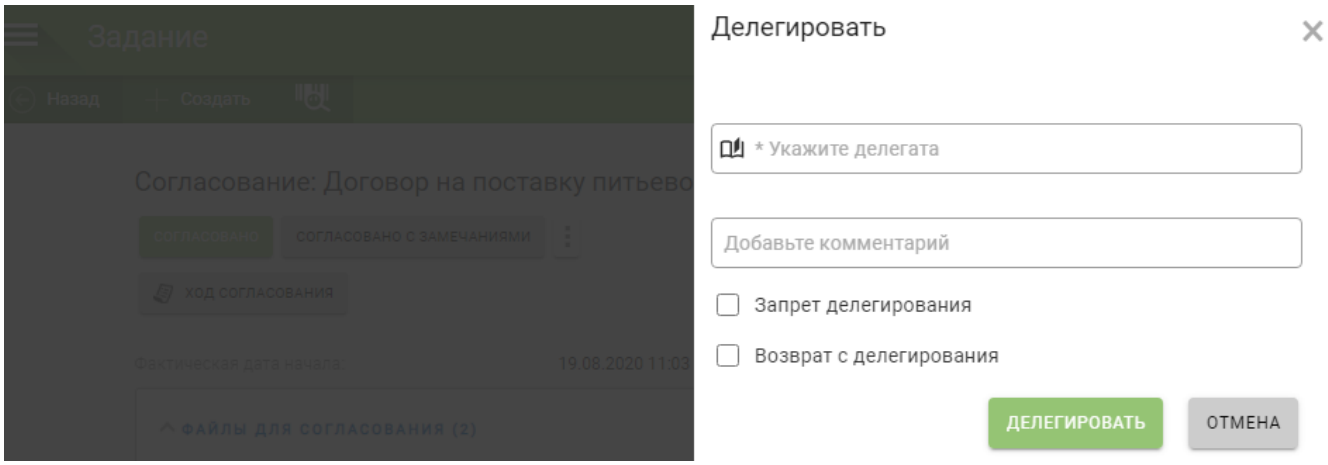


Рисунок 267. Делегирование задания на подписание

3. Выберите сотрудника, которому делегируется задание.
4. Добавьте комментарий к делегированию.
5. Чтобы запретить повторное делегирование задания, установите флаг `Запрет делегирования`.
6. Чтобы вернуть себе задание после его завершения делегатом, установите флаг **Возврат с делегирования**.

   После завершения задания карточка вернется делегировавшему в состоянии `Возвращено с делегирования`.

7. Нажмите кнопку **Делегировать**.

Состояние задания изменится на Делегировано. Задание будет перемещено в папку **Мои задания** › **Исходящие** › **Делегировано**. *Делегат* увидит данное задание в папке **Мои задания** › **Входящие**.

В верхней части страницы задания появится блок с информацией о последнем делегировании.

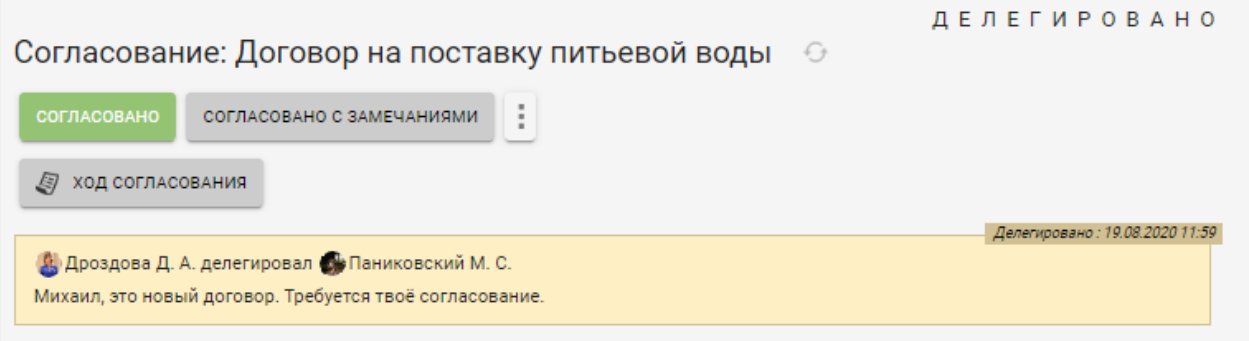


Рисунок 268. Информация о ходе делегирования в карточке задания

В ход согласования будет добавлена запись о делегировании.

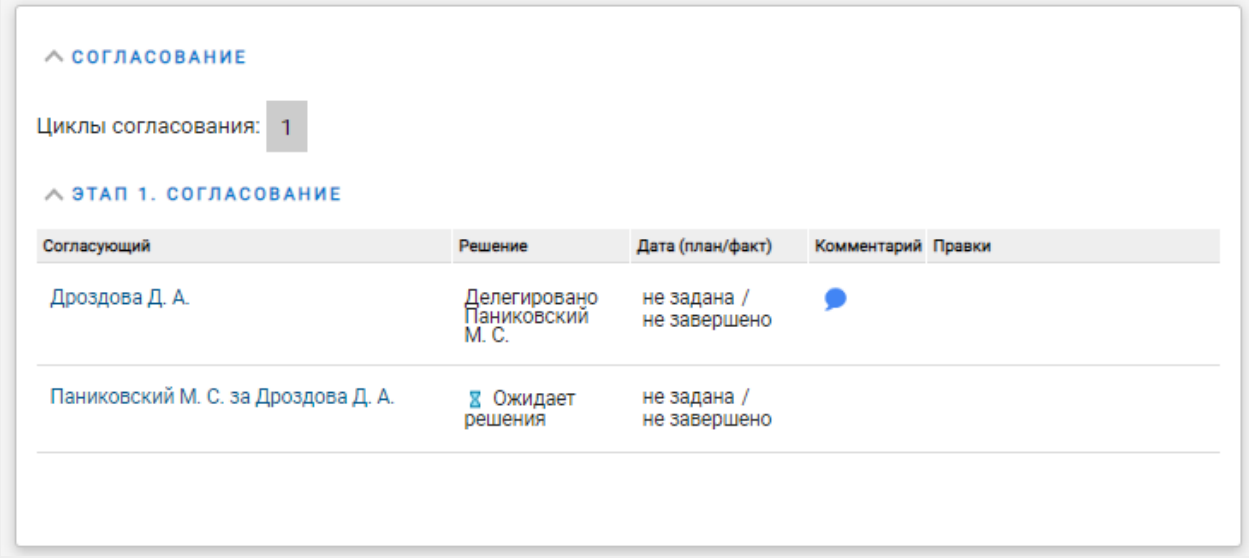


Рисунок 269. Записи о делегировании задания в Ходе согласования

Обсудить согласование

Обсуждение согласования ведется с помощью компонента Обсуждение, который доступен во всех карточках заданий согласования.

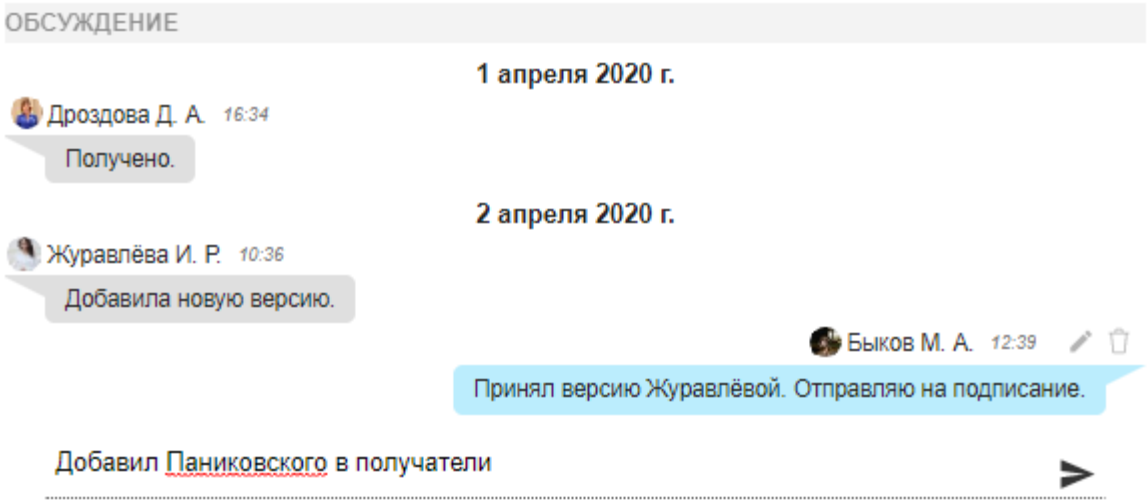


Рисунок 270. Обсуждение в задании консолидатора

В рамках одного согласования действует общий список комментариев. Комментарии, оставленные участниками согласования в своих заданиях, видны другим участникам в их заданиях. Например, комментарий, оставленный согласующим, будет виден в задании консолидатора.

Добавить дополнительных согласующих

Дополнительные согласующие — участники, добавленные другими согласующими в процессе согласования. Дополнительные согласующие автоматически добавляются в следующий цикл согласования, становясь тем самым полноправными участниками.

Добавление согласующих является вариантом завершения задания согласования, возможность принятия которого определяется настройками согласования.

Дополнительные согласующие добавляются в согласование сразу или после их утверждения инициатором согласования. Необходимость утверждения определяется настройками согласования.

В последовательном согласовании дополнительные участники согласуют сразу после сотрудника, который их добавил. В параллельном согласовании — (после добавления) вместе с остальными участниками.

Согласование может вернуться к участнику, который добавил согласующих, если это предусмотрено настройками согласования. После принятия решения дополнительными согласующими или, если инициатор не утвердил добавление согласующих.

Добавление новых согласующих в согласование

Чтобы добавить дополнительных согласующих

1. Откройте полученное задание согласования.
2. Примите решение **Добавление согласующих** (название может отличаться).

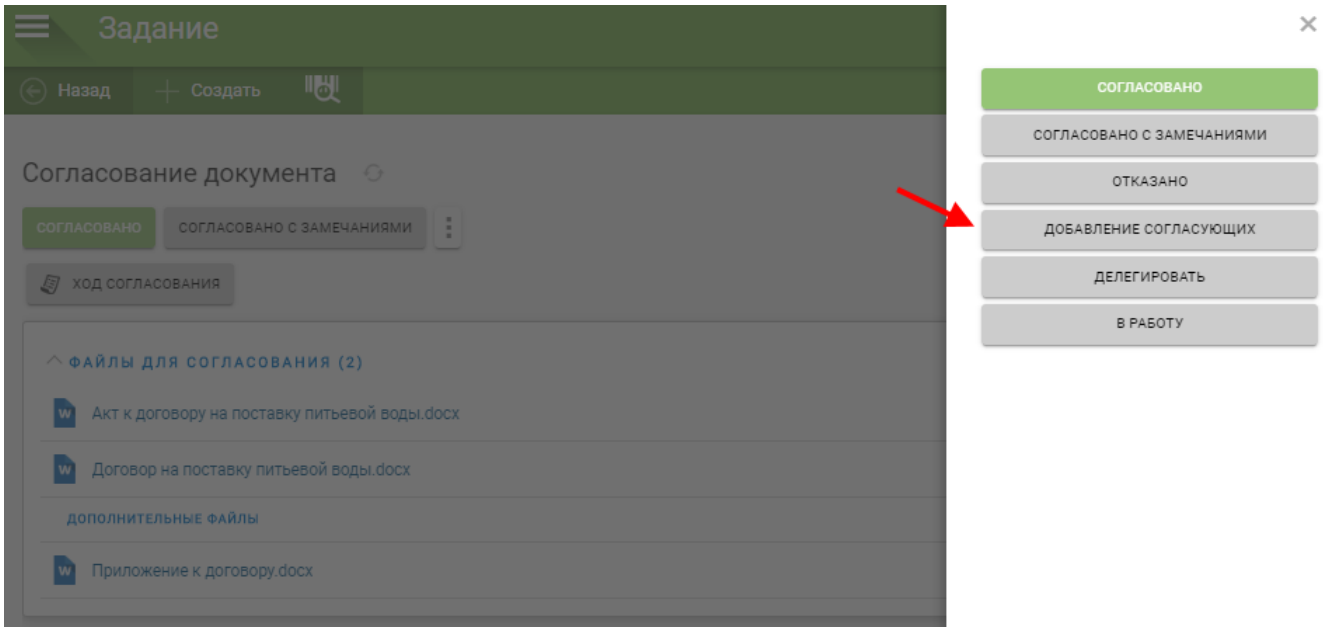


Рисунок 271. Кнопка принятия решения "Добавить согласующих"

Будет предложено выбрать новых участников согласования и добавить комментарий.

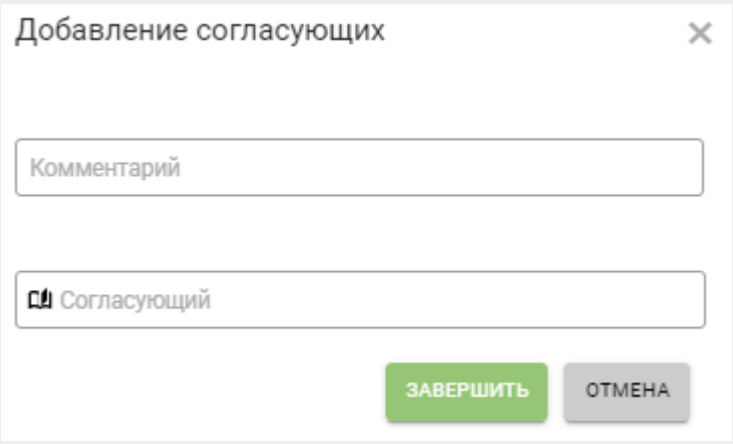


Рисунок 272. Выбор дополнительных согласующих

3. Выберите дополнительных согласующих, введите комментарий и нажмите **Завершить**.

Задание будет завершено с результатом *Добавление согласующих*.

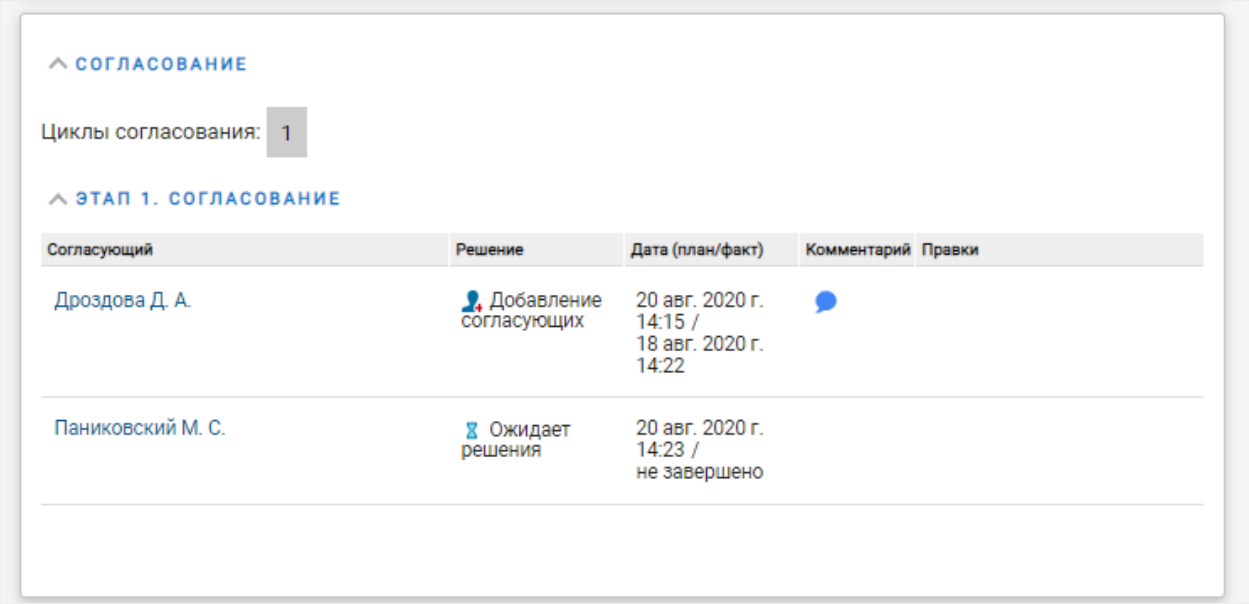


Рисунок 273. Ход согласования с решением "Добавление согласующих"

В зависимости от настроек согласования новые участники будут добавлены сразу или после их утверждения инициатором согласования.

Утверждение дополнительных согласующих

Если это предусмотрено настройками согласования, после добавления новых согласующих участником, инициатор получит задание на их утверждение.

Чтобы принять решение по новым участникам:

1. Откройте задание с названием *Запрос на добавление участников согласования*.

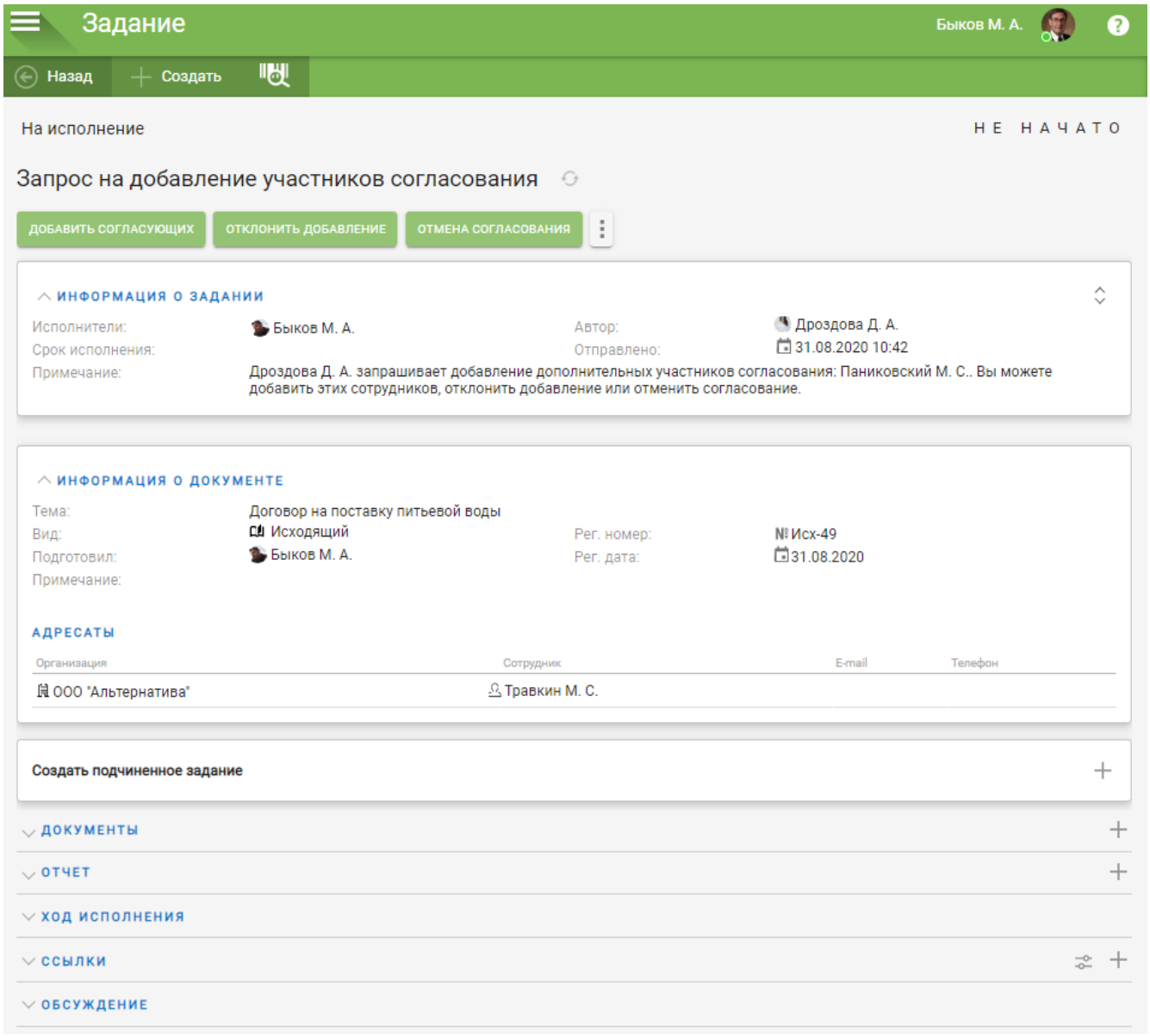


Рисунок 274. Задание на утверждение дополнительных согласующих



Список добавляемых согласующих приведён в описании задания.

2. Примите решение по заданию, нажав соответствующую кнопку принятия решения на панели операций:

Добавить согласующих	Дополнительные согласующие будут добавлены в согласование, согласование продолжится; если согласование перейдет на новый цикл, указанные сотрудники будут добавлены в него автоматически без повторного утверждения.
Отклонить добавление	Дополнительные согласующие не будут добавлены в согласование, согласование продолжится.
Отмена согласования	Согласование будет остановлено.

Подчиненные задания согласования

Подчиненное задание согласования — особый класс заданий. Подчинённые задания могут быть созданы из задания согласования с целью сбора информации, связанной с согласованием. Сотрудники-исполнители подчинённых заданий непосредственно не участвуют в согласовании. Результаты подчинённых согласований не влияют на ход согласования и используются только участником согласования для принятия решения. В подчиненные задания прикладываются согласуемые версии файлов.

Участник согласования может:

- Создать одно или несколько подчиненных заданий согласования к своему заданию для одного или нескольких исполнителей. Такие задания никак не связаны между собой и всегда рассылаются параллельно.
- Скопировать комментарии из подчиненных заданий в согласование.
- Отзывать подчинённые задания.
- Завершить своё задание согласования без учёта состояния заданий подчинённых.

Поведение подчиненных заданий при отзыве задания согласования определяется настройками вида задания согласования.

Создать подчиненное задание согласования

Подчиненное задание может быть создано в задании согласования, если в настройках этапа включена данная возможность.

Чтобы создать подчиненное задание:

1. Откройте полученное задание согласования.
2. Перейдите к раздел *Подчиненные задания* и нажмите кнопку +.
3. В зависимости от настроек может быть предложено выбрать вид создаваемого задания. В этом случае выберите вариант Подчиненное к согласованию.
4. Если вид подчиненного задания не может быть определён однозначно, будет предложено подтвердить создание задания с видом текущего задания. Подтвердите использование указанных настроек, чтобы продолжить.

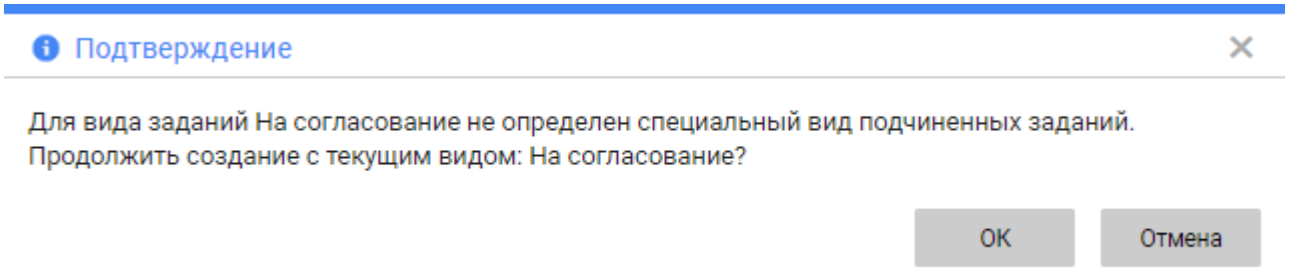


Рисунок 275. Диалоговое окно подтверждения создания подчиненного задания с текущим видом

Будет открыто диалоговое окно создания подчиненных заданий.



Рисунок 276. Диалоговое окно создания подчиненных заданий

В зависимости от настроек родительского задания, некоторые поля могут быть скопированы в подчиненное задание.

5. Заполните требуемые поля подчинённого задания: Название, Описание, "Записи справочника сотрудников" (если выбрано несколько исполнителей, им будут отправлены отдельные задания), Срок исполнения.
6. Нажмите кнопку **Создать**, чтобы отправить задания.

В разделе *Подчиненные задания* появится запись о созданном задании.

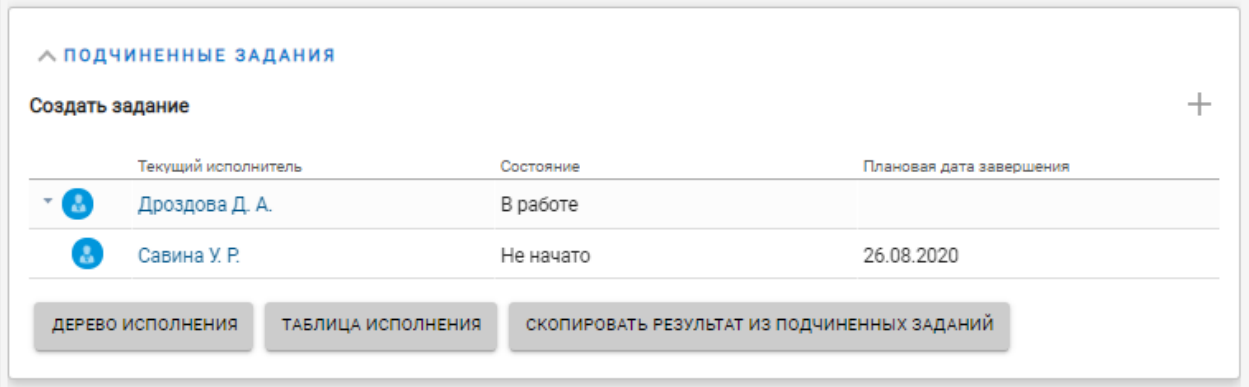


Рисунок 277. Информация о подчиненных заданиях

Выполнить подчиненное задание согласования

Для работы с подчиненными заданиями в Web-клиенте используется карточка задания *На согласование* со следующими особенностями:

- В списке *Файлы для согласования* отображаются версии родительского задания.
- Нельзя загружать новые версии файлов и изменять существующие.
- Результаты завершения подчиненного задания не записываются в *Ход согласования* и *Лист согласования*.
- При завершении подчиненного задания вне независимости от настроек этапа не запрашивается электронная подпись.
- В стандартных решениях отчёт в подчиненном задании согласования добавить нельзя.

В остальном работа с подчиненным заданием согласования не отличается от работы с заданиями согласования: можно взять задание в работу, делегировать его другому сотруднику, прокомментировать предоставленную версию файла, выполнить.

Скопировать комментарии из подчиненного согласования

Согласующий-автор подчиненных заданий, может перенести комментарии и файл комментария из подчиненных заданий в своё согласование.

Чтобы перенести комментарии из подчиненного согласования:

1. Дождитесь выполнения подчиненного задания.
2. Откройте задание согласования, для которого было создано подчиненное задание.
3. Перейдите к разделу *Подчиненные задания*.

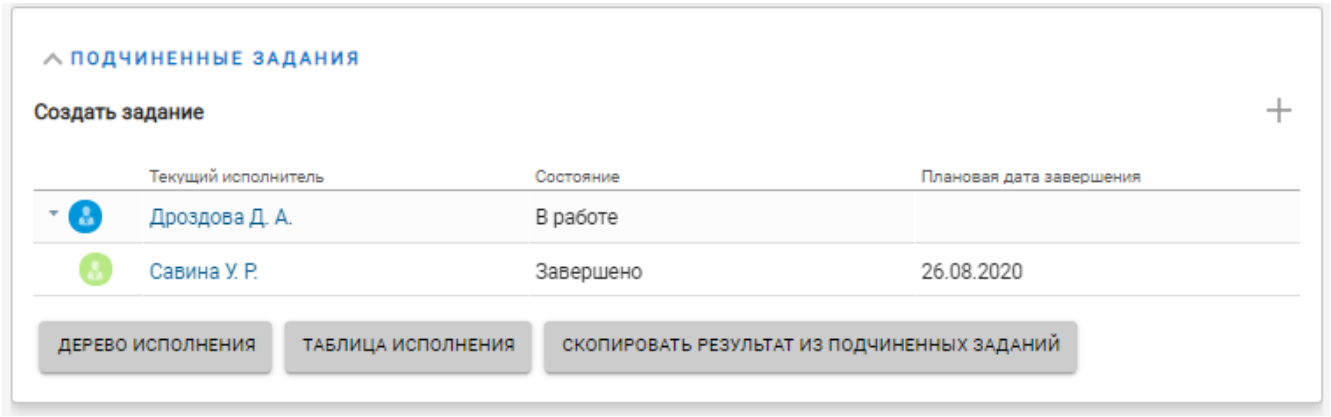


Рисунок 278. Страница "Подчиненные задания" в карточке задания

4. Нажмите кнопку **Скопировать результат из подчиненных заданий**. Будет открыто диалоговое окно копирования комментариев.

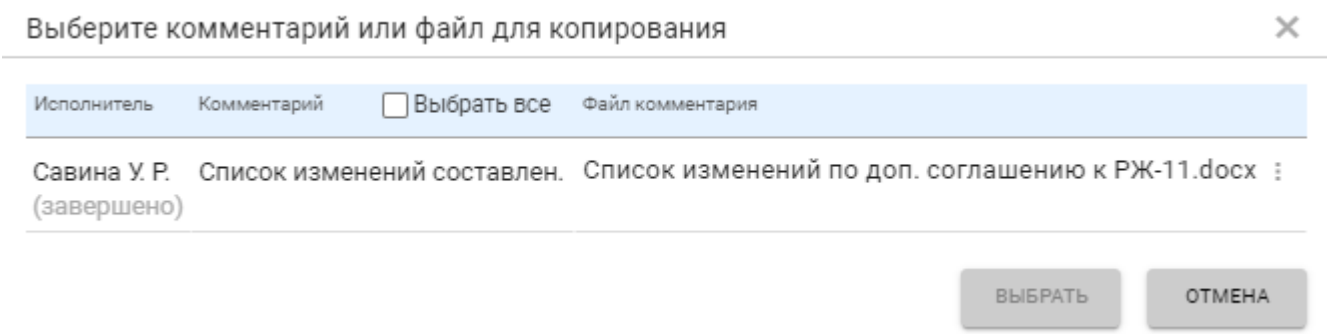


Рисунок 279. Диалоговое окно копирования комментариев подчиненного согласования

5. Выберите файл и/или комментарий, которые требуется скопировать, и нажмите кнопку **Выбрать**.

> Чтобы выбрать файл или комментарий, нажмите на него. Чтобы снять выбор, нажмите повторно.
>
> Можно выбрать несколько комментариев, но только один файл. При

повторном переносе комментарии будут добавлены к перенесённым ранее, а файл будет заменён.

Чтобы просмотреть файл комментария или сохранить его на диск, нажмите кнопку ⋮.

Выбранные комментарии будут добавлены в комментарии к решению по согласованию.

Настройка безопасности объектов

Система Docsvision поддерживает несколько моделей безопасности, позволяющих настраивать права доступа к объектам:

- Ролевую,
- Мандатную,
- Дискреционную.

Настройка безопасности с использованием первых двух моделей осуществляется специалистом при настройке системы.

Возможность настройки дискреционной безопасности доступна пользователям при наличии права владения объектом.

Web-клиент предоставляет возможность настройки дискреционных прав доступа с использованием диалога настройки.

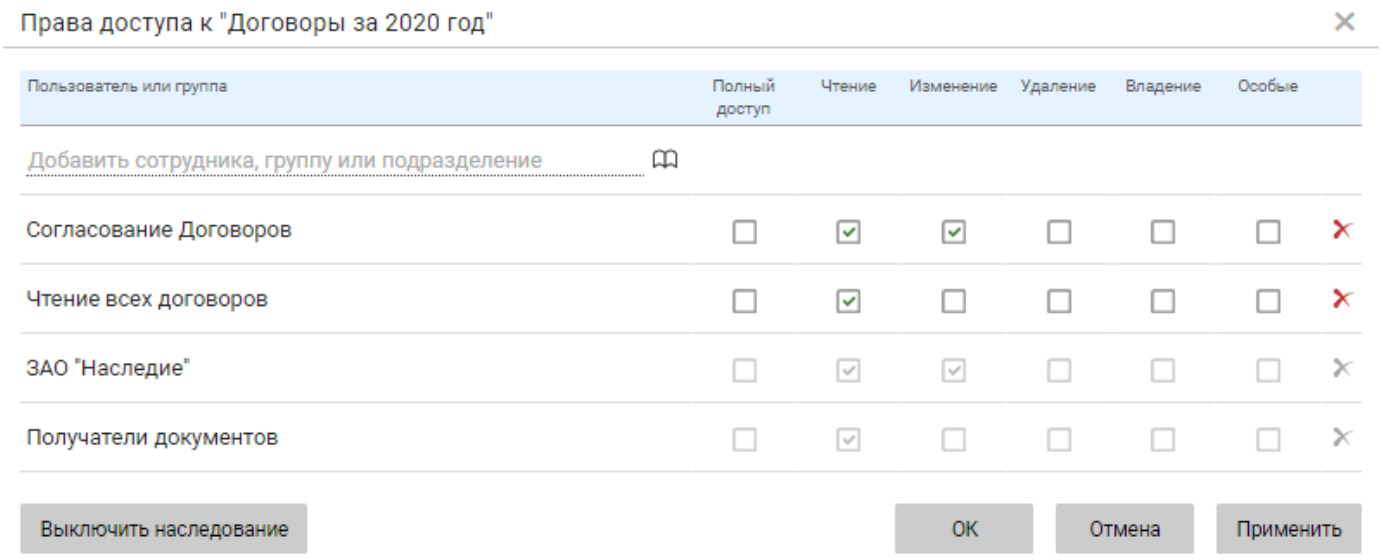


Рисунок 280. Диалог назначения дискреционных прав доступа

Данный диалог позволяет:

- Изменять существующие права доступа.
- Предоставлять доступ субъектам из Справочника сотрудников.
- Включать и выключать наследование прав доступа от родительского объекта.

Субъекты могут быть выбраны только из Справочника сотрудников. При этом также отображаются и могут быть изменены права, назначенные для субъектов из Active Directory.



Права могут быть предоставлены отдельным сотрудникам, группам сотрудников и/или организациями и подразделениям.

При одновременном редактировании прав доступа к справочнику с разных окон или вкладок браузера изменения сохранятся только из последнего закрытого окна.

Права доступа, отображаемые в диалоге, разделены на две группы:

Непосредственные	Права доступа, назначенные для данного объекта.
Унаследованные	Права доступа, назначенные для родительского объекта и унаследованные текущим, если наследование прав не отключено.

Унаследованные права доступа изменить нельзя, соответствующие элементы управления будут заблокированы. Унаследованные права доступа нужно изменять непосредственно в родительском объекте, от которого они наследуются.

Возможность изменения прав доступа также будет недоступна, если у пользователя отсутствуют права на изменение прав доступа.

Права доступа включают следующие категории:

Полный доступ	Предоставляются права на чтение, изменение, удаление и владение объектом;
Чтение	Предоставляет доступ на просмотр содержимого объекта.
Изменение	Предоставляет доступ на изменение содержимого объекта.
Удаление	Предоставляет доступ на удаление объекта.

Владение	Предоставляет доступ на изменение прав доступа к объекту.
Особые	Используются расширенные настройки доступа (могут быть установлены и изменены в Windows-клиенте).

Право на выполнение операции в определённой категории может иметь одно из трёх значений:

☐ **Разрешение отсутствует**	Субъекту будет запрещено выполнение операции, если разрешение не получено от других субъектов (например, пользователь может входить в группу, которой предоставлено разрешение).
☑ **Разрешено**	Субъекту будет разрешено выполнение операции, если нет явного запрета от других субъектов.
☒ **Запрещено**	Субъекту будет запрещено выполнение операции, в т. ч. если предоставлено разрешение от других субъектов.

Правила работы с диалогом назначения дискреционных прав доступа

Далее рассмотрены общие правила работы с диалогом назначения дискреционных прав доступа. Инструкция по настройке прав для папок приведена в соответствующем пункте.

Чтобы назначить права доступа к объекту:

1. Откройте диалог назначения дискреционных прав доступа к объекту, для которого настраиваются права.

   При отсутствии права владения объектом диалог назначения прав доступа открывается в режиме чтения, без возможности изменения настроек безопасности.

2. Если требуется, измените наследование прав.

3. Если требуется, добавьте нового субъекта:
4. Измените права доступа для субъекта.
5. Удалите права доступа нажатием на кнопку ✕ в строке субъекта.

   Разрешение удаляется без подтверждения.

   > ⚠ Если у объекта были удалены все права доступа, то добавить права сможет только администратор безопасности (член группы Docsvision Security Administrator).

6. Нажмите кнопку **ОК** в окне диалога назначения дискреционных прав доступа, чтобы применить изменения.
7. Чтобы отменить ещё не применённые изменения, нажмите кнопку **Отмена**.

Отключить наследование прав доступа

В зависимости от выбранного варианта отключения наследования, права, полученные текущим объектом от родительского объекта будут сохранены или удалены.

Чтобы восстановить наследование, нажмите кнопку **Включить наследование**.

Настройки применятся сразу после нажатия кнопки.

Чтобы отключить наследование прав доступа от родительского объекта:

1. Нажмите кнопку **Выключить наследование** и выберите способ отключения наследования:

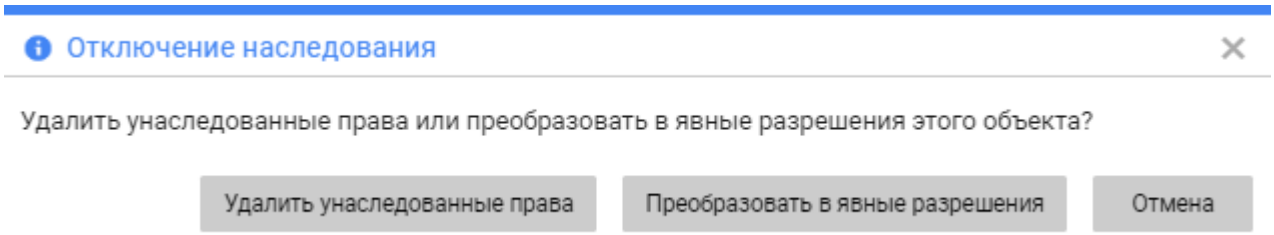


Рисунок 281. Диалог отключения наследования

2. Нажмите любую из кнопок:

   Удалить унаследованные права

   Права доступа, установленные для родительского объекта, будут удалены из текущего объекта.

Наследование изменений прав родительского объекта будет отключено.

Преобразовать в явные разрешения

Права доступа, установленные для родительского объекта, будут применены к текущему объекту.

Наследование изменений прав родительского объекта будет отключено.

Отмена

Отменить операцию отключение наследования.



Настройки наследования применяются сразу после нажатия кнопки.

Добавить нового субъекта

Чтобы добавить нового субъекта:

1. Откройте окно выбора записи из Справочника сотрудников нажатием на кнопку в первой строке списка прав доступа.
2. Добавьте новую запись в список прав, выбрав из справочника сотрудника группу или подразделение.

   По умолчанию права не предоставляются.

Изменить права доступа для субъекта

1. Чтобы разрешить субъекту выполнять операции, нажмите несколько раз на флаг в соответствующей категории, пока флаг не перейдёт в состояние .
2. Чтобы запретить субъекту выполнять операции, нажмите несколько раз на флаг в соответствующей категории, пока флаг не перейдёт в состояние .
3. Чтобы отменить запрет и разрешение на выполнение операции для субъекта, нажмите несколько раз на флаг в соответствующей категории, пока флаг не перейдёт в состояние .

   Разрешение, установленное в категории *Полный доступ*, будет применено ко всем другим категориям.

Работа со справочниками

Работа в Web-клиенте

Работа со справочниками

Описание работы со справочниками Web-клиента: взаимодействие со справочниками, настройка и администрирование.

- Справочник контрагентов
- Справочник сотрудников
- Больше информации в навигационном меню слева…

Справочник контрагентов

Справочник контрагентов позволяет создавать, редактировать и просматривать иерархическую структуру узлов организаций и подразделений, групп контрагентов. Справочник позволяет создавать и объединять сотрудников в группы внешних организаций, с которыми взаимодействует компания.

Пример пользовательского интерфейса справочника приведён ниже.

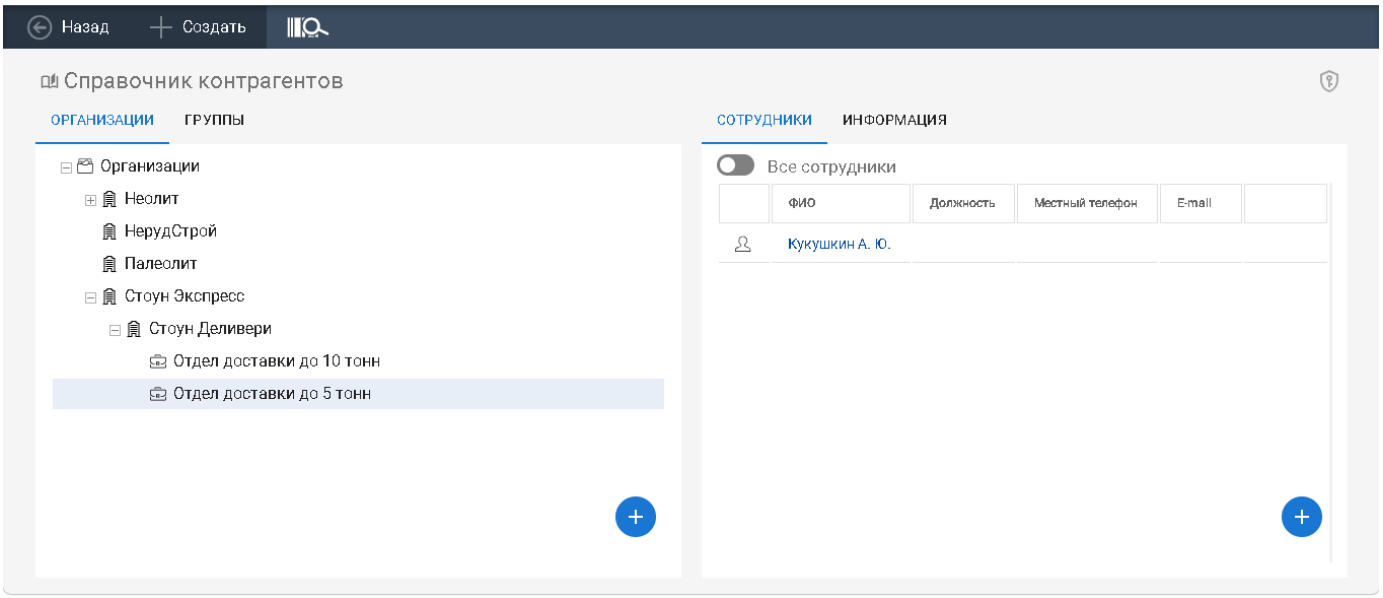


Рисунок 282. Общий вид справочника контрагентов

Справочник визуально поделён на две области. При наведении на границу областей изменяется вид указателя, границу можно сжать, потянув вправо или влево. Допускается сжатие вплоть до 10% от исходного.

В левой области находятся две вкладки: *Организации*, *Группы*, в каждой вкладке находятся узлы, отсортированные по алфавиту.

Каждая вкладка имеет свой корневой узел, объединяющий все остальные. Создавать новые узлы разрешается только внутри корневого узла, создать новый корневой узел невозможно.

Чтобы раскрыть узел, необходимо нажать на стрелку слева от его названия, повторное нажатие на стрелку скрывает узел. При однократном нажатии на название, узел переходит в фокус, а в правой области справочника отображаются сотрудники.

- Во вкладке *Организации* отображаются *Организации* и *Подразделения* в виде иерархической структуры или дерева.
- Во вкладке *Группы* отображаются *Группы* контрагентов также в виде иерархической структуры или дерева.



Если в узле больше 500 дочерних элементов, они отображаются порциями. Первая порция — 500, чтобы загрузить следующую, нажмите *Ещё*. Если число дочерних элементов меньше 500, кнопка не отображается.

Правая область имеет две вкладки: *Сотрудники* и *Информация*.

- Во вкладке *Сотрудники* отображаются *Сотрудники*, относящиеся к выбранному в левой области узлу справочника: *Организации*, *Подразделению* или *Группе*.
- Во вкладке *Информация*, отображается информация об узле, открытом в левой области, например информация об *Организации*, *Подразделении* или *Группе*. Также информацию об узле справочника можно посмотреть, выбрав в контекстном меню нужного узла пункт *Информация*.

Над правой областью расположен элемент управления , позволяющий задать общие настройки дискреционной безопасности для справочника.

Работа с организациями контрагентов

Основная информация о выбранных организации или подразделении контрагента доступна в правой области справочника, во вкладке *Информация*. Полную информацию можно посмотреть, выбрав пункт *Информация* в контекстном меню.

Посмотреть информацию о контрагенте

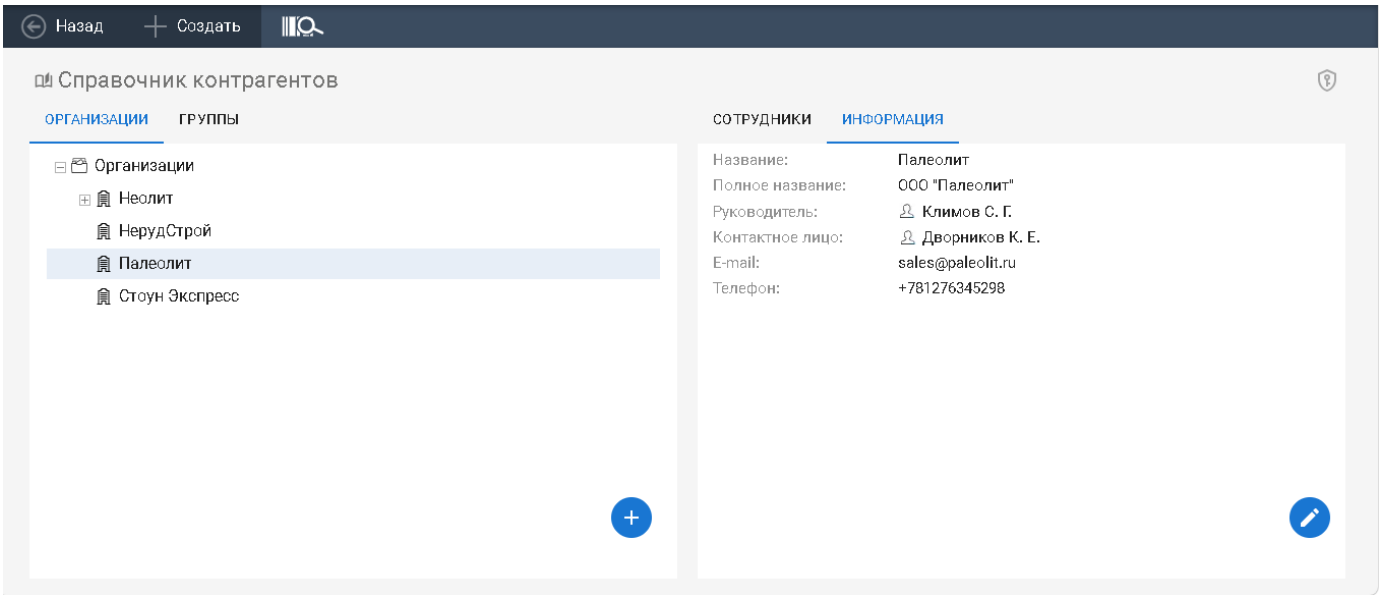


Рисунок 283. Просмотр основной информации об организации контрагента при открытой вкладке "Информация"

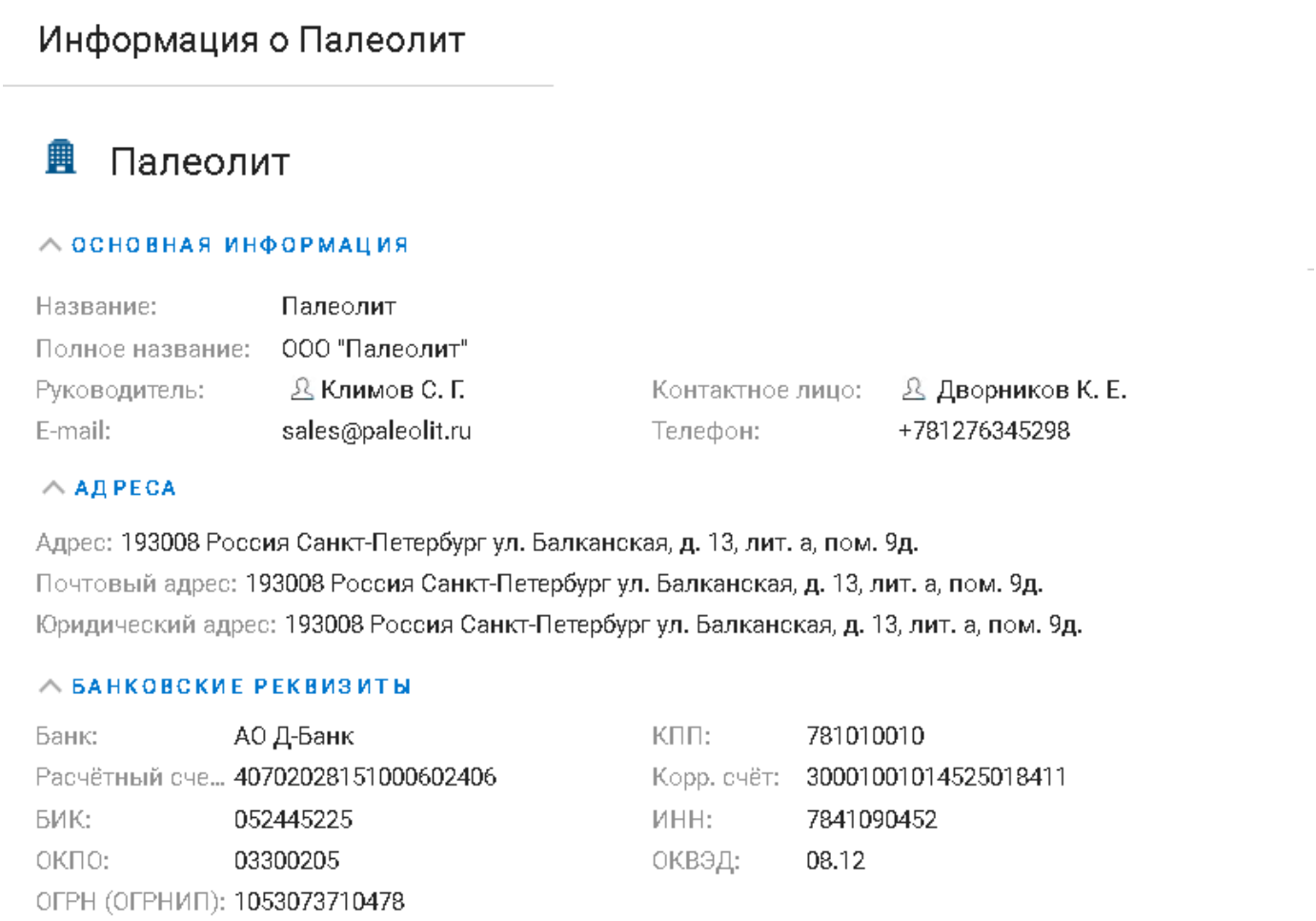


Рисунок 284. Просмотр полной информации об организации контрагента при выборе пункта контекстного меню "Информация"

В справочнике контрагентов предусмотрена возможность создания, изменения и удаления узлов организаций.



Возможность создания, изменения, удаления и просмотра узлов, а также добавления или изменения числа сотрудников в выбранном узле может быть ограничена настройками безопасности.

Создать новую организацию контрагента

Функции работы со справочником недоступны пока не создана организация контрагента.

Чтобы создать новый узел организации контрагента:

1. Перейдите в справочник контрагентов.
2. В левой области выберите вкладку **Организации**.

> Допускается создание как самостоятельных организаций, так и создание дочерних организаций.
>
> Организация не будет сохранена, пока не нажата кнопка **Создать**. Это значит, что в процессе создания организации в ней нельзя создавать сотрудников контрагентов, включая руководителя организации.

3. Выберите необходимый узел.
4. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего узла.
5. Заполните все необходимые поля и нажмите **Создать**.

   Подробнее о полях карточки организации контрагента см. "Перечень полей карточки".

Изменить организацию контрагента

Чтобы изменить созданный ранее узел организации:

1. На вкладке *Организации* в левой области справочника выберите узел из иерархического списка.
2. Выберите один из доступных способов изменения:
   - При помощи элемента управления  на вкладке *Информация* в правой области справочника.
   - Из соответствующего пункта контекстного меню текущего узла.
3. Измените необходимые поля и нажмите **Сохранить**.

Удалить организацию контрагента

Чтобы удалить созданный ранее узел организации:

1. На вкладке *Организации* в левой области справочника выберите узел из иерархического списка.
2. Вызовите контекстное меню на выбранном узле.
3. Выберите соответствующий пункт из контекстного меню, чтобы удалить узел.

4. Появится предупреждение, подтвердите действие кнопкой **ОК**.



Организации и подразделения удаляются со всеми дочерними узлами, включая сотрудников.

Перечень полей карточки организации контрагента

Основная информация об организации контрагента

Основная информация об организации контрагента может быть заполнена при создании новой организации или изменена после создания.

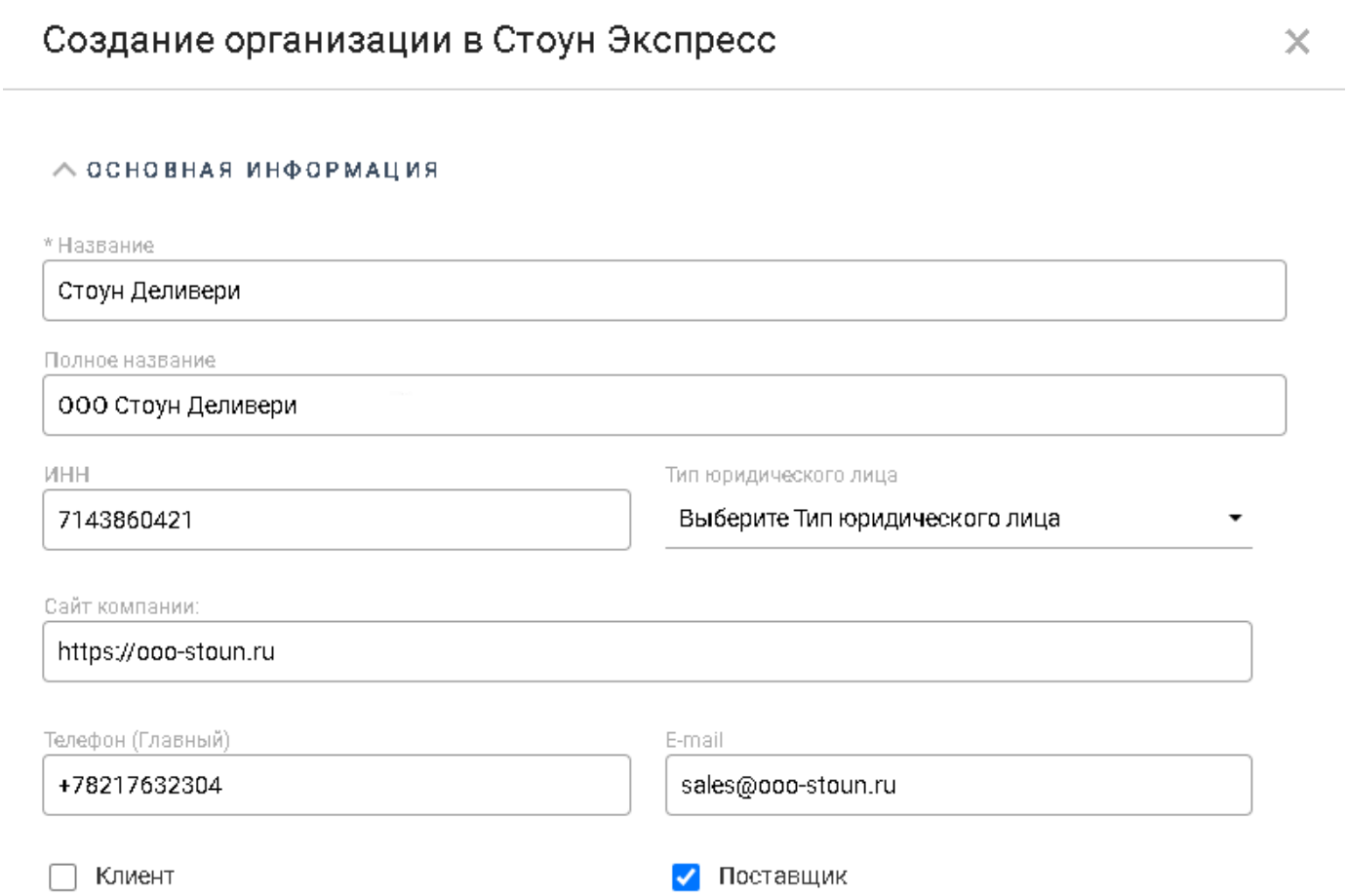


Рисунок 285. Основная информация об организации контрагента

Название	В поле указывается сокращенное название, которое будет отображаться в справочнике Docsvision.
Полное название	В поле указывается полное официальное название.
ИНН	ИНН организации контрагента
Тип юридического лица	Раскрывающийся список доступных типов юридических лиц

Сайт компании	Сайт компании в сети интернет
Телефон (главный)	В поле указывается номер телефона для связи.
E-mail	В поле указывается адрес для отправки электронных писем.
Флаги "Клиент" и "Поставщик"	Если организация-контрагент является заказчиком или поставщиком товаров или услуг, установите один из флагов. Допускается установка обоих флагов.
Адрес	В поле указывается фактический адрес. При изменении адреса в поле Адрес в разметке создания или изменения организации, информация будет скопирована в поля Почтовый адрес и Юридический адрес.
Почтовый адрес	В поле указывается адрес, который будет использован для отправки почтовой корреспонденции.
Юридический адрес	В поле указывается адрес регистрации организации.

Адрес

При выборе поля Адрес, Почтовый адрес или Юридический адрес из карточки организации контрагента открывается новое окно для добавления адреса:

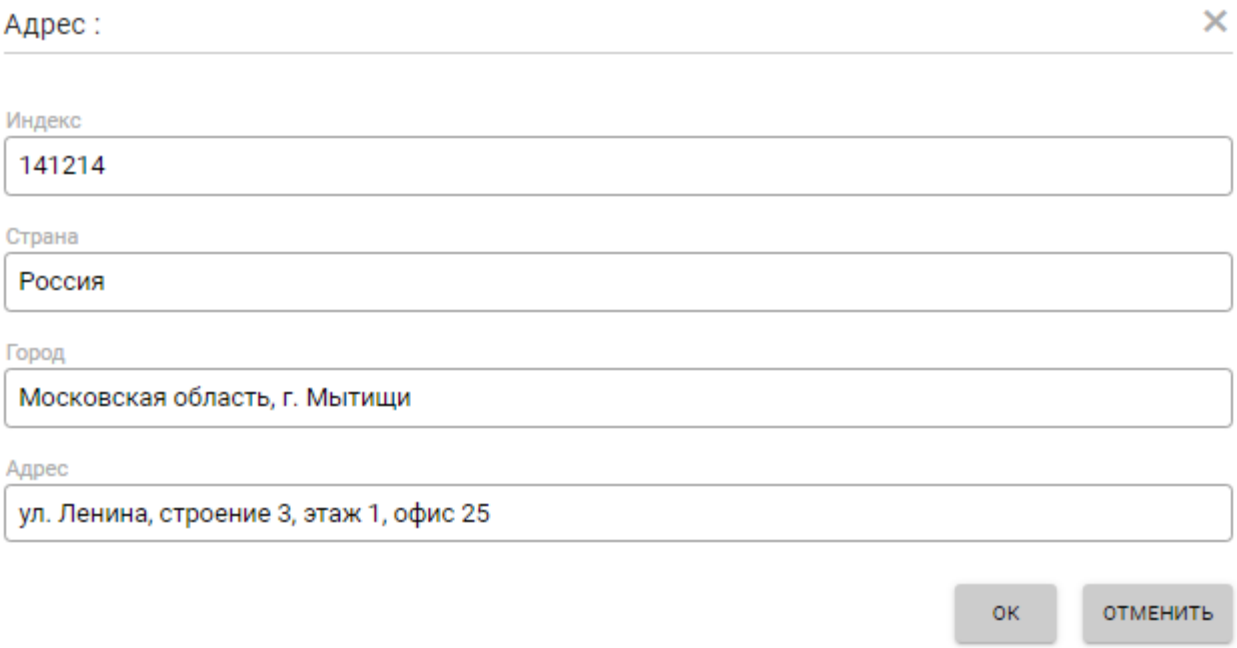


Рисунок 286. Окно добавления адреса организации

При изменении адреса в поле Адрес в разметке создания или изменения организации, информация будет скопирована в поля Почтовый адрес и Юридический адрес.

При добавлении адреса для заполнения доступны следующие поля:

Индекс Почтовый индекс организации.

Город Город, в котором организация находится.

Адрес Улица, дом, офис.

Страна Страна нахождения организации.

Банковские реквизиты

Банковские реквизиты для организации или подразделения можно заполнить при создании новой организации или изменить после создания:

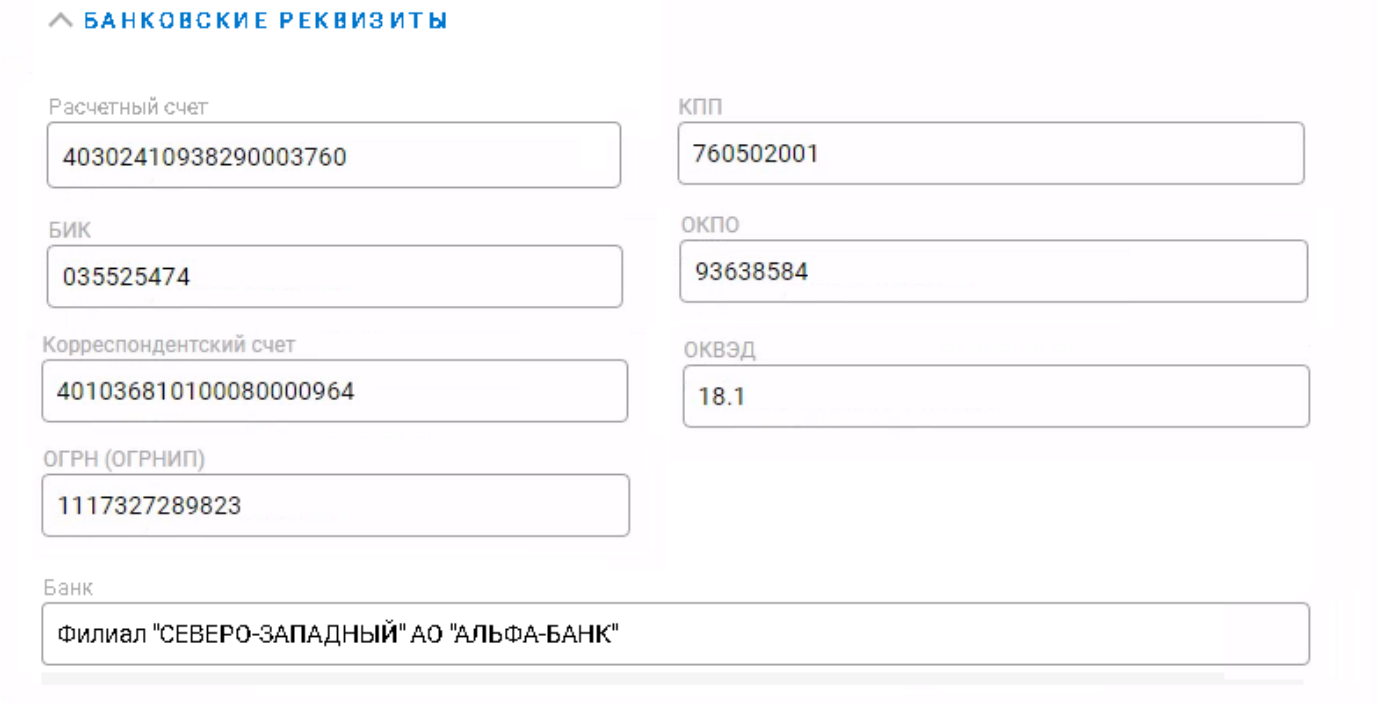


Рисунок 287. Банковские реквизиты организации контрагента

В банковских реквизитах организации контрагента можно указать следующие данные:

Банк

Название банка.

Расчетный счет
БИК
ОКПО
КПП
Корреспондентский счет
ИНН
ОКВЭД
ОГРН (ОГРНИП)

Контакты

1. В области *Контакты* нажмите на кнопку **Добавить** +.
2. Заполните поля:



Рисунок 288. Контакты организации контрагента

- В столбце *Тип* из раскрывающегося списка выберите тип контакта: *телефон*, *факс*, *E-mail*, *адрес URL*, *прочее*.
- В столбце *Название* укажите, соответственно, *номер телефона*, *факса*, *E-mail*, *URL-адрес* и прочее.
- В столбце *Комментарий* укажите дополнительную информацию.

3. Удалить контакты можно при помощи кнопки .

Коды

В секции *Коды* вкладки *Дополнительно* нажмите кнопку **Добавить** , чтобы ввести государственные коды и указать их значения.



Рисунок 289. Коды организации контрагента

Удалить коды можно при помощи кнопки .

Дополнительные банковские реквизиты

1. В секции *Дополнительные банковские реквизиты* нажмите кнопку **Добавить** , чтобы ввести дополнительные банковские реквизиты организации.
2. Заполните поля в стобцах:

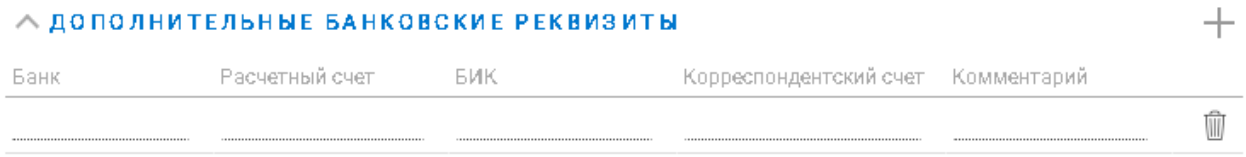


Рисунок 290. Дополнительные банковские реквизиты организации контрагента

- Банк
- Расчетный счет
- БИК
- Корреспондентский счет
- ИНН
- В столбце *Комментарий* укажите дополнительную информацию

3. Удалить дополнительные банковские реквизиты можно при помощи кнопки .

Комментарии

Текстовое поле для указания дополнительной информации об организации контрагента.



Рисунок 291. Поле "Комментарии" в организации контрагента

Работа с подразделениями контрагента

В справочнике контрагентов реализована возможность создания подразделений организаций. Каждая организация может иметь неограниченное количество подразделений. Подразделения являются составной частью организаций и не могут быть созданы вне организации.

Посмотреть информацию о подразделении

Основная информация о выбранной организации или подразделении контрагента доступна в правой области справочника, во вкладке *Информация*.

Полную информацию можно посмотреть, выбрав пункт *Информация* в контекстном меню.

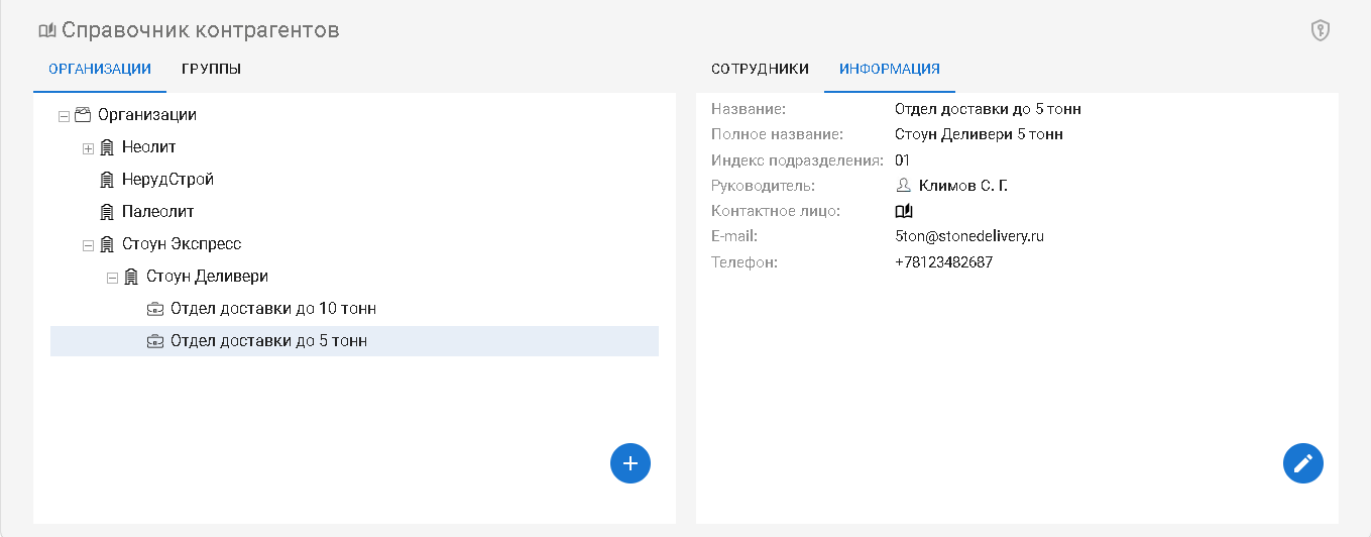


Рисунок 292. Просмотр основной информации о подразделении контрагента при открытой вкладке "Информация"

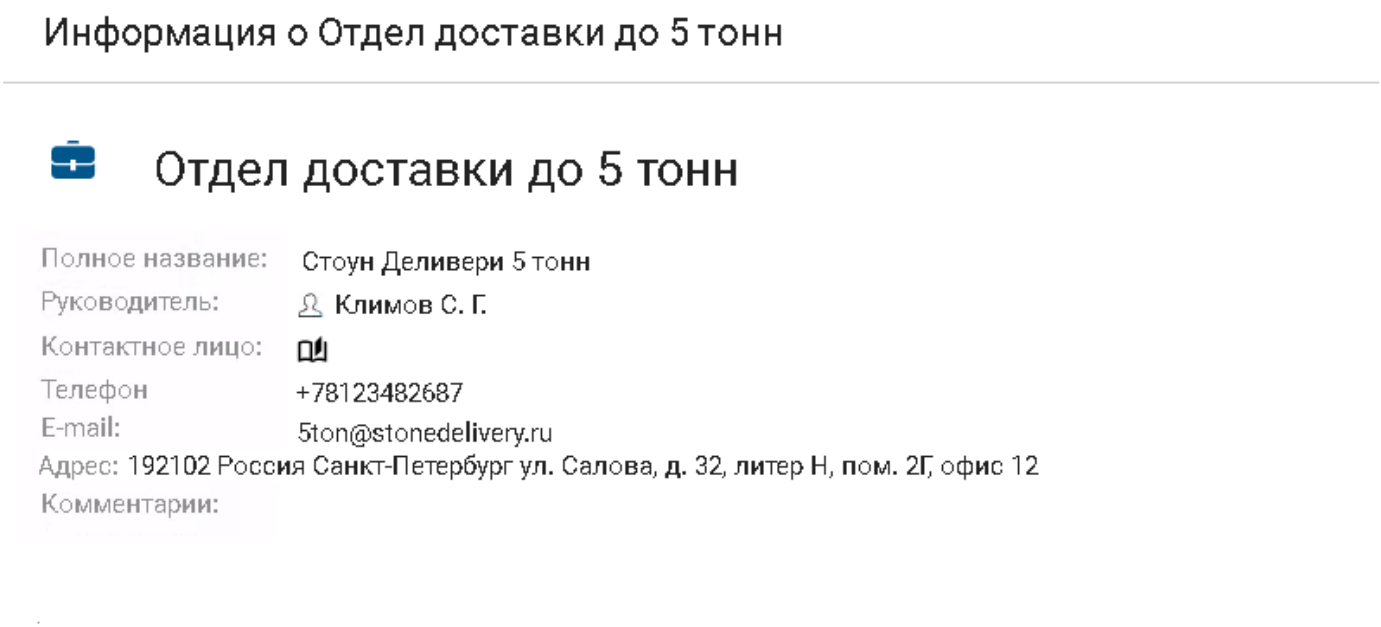


Рисунок 293. Просмотр полной информации о подразделении контрагента при выборе пункта контекстного меню "Информация"

В справочнике контрагентов предусмотрена возможность создания, изменения и удаления узлов подразделений.

Возможность создания, изменения, удаления и просмотра узлов, а также добавления или изменения числа сотрудников в выбранном узле может быть ограничена настройками безопасности.

Создать новое подразделение контрагента

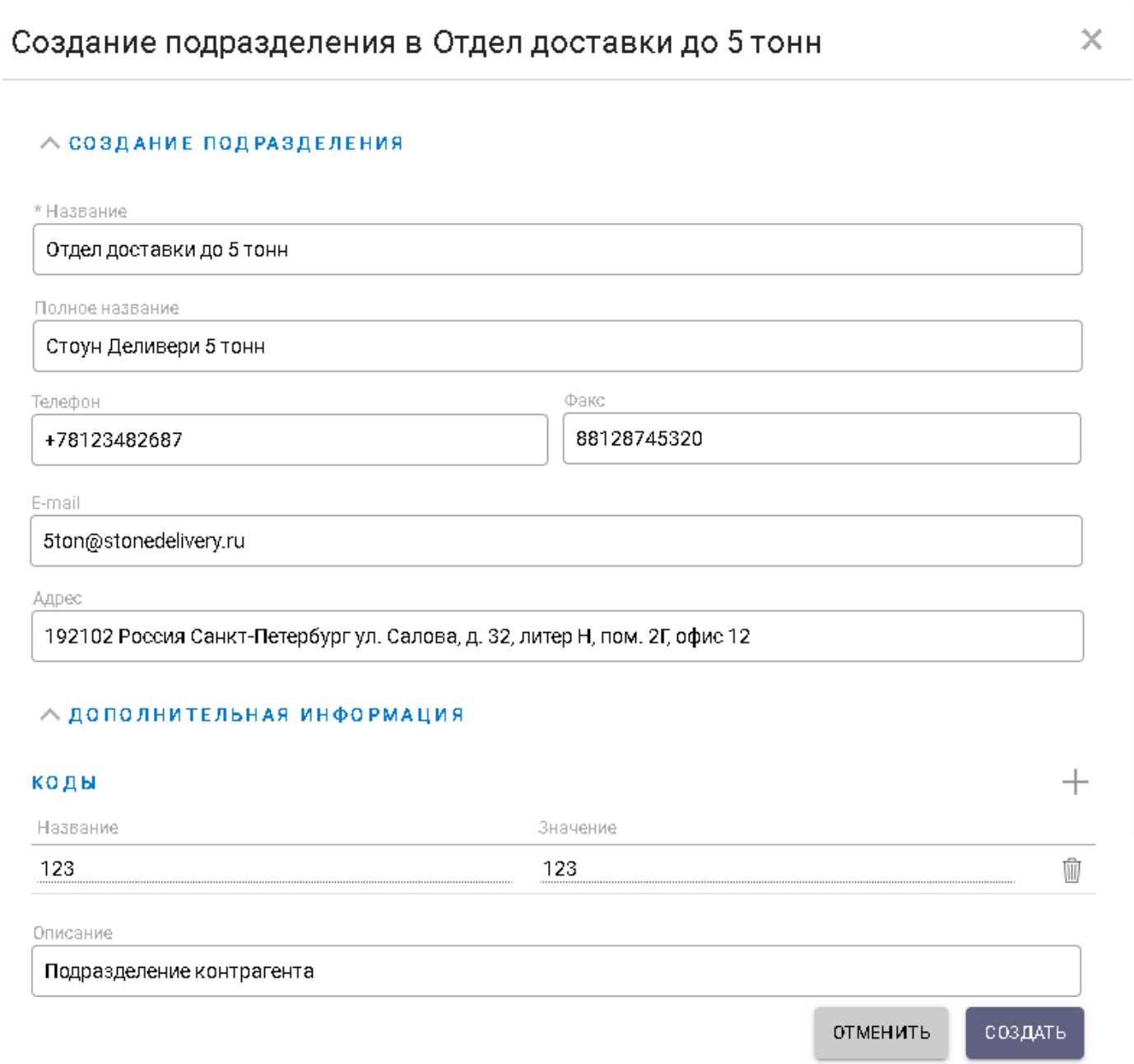


Рисунок 294. Создание нового подразделения контрагента

Чтобы создать новое подразделение контрагента:

1. Выберите необходимый узел.
2. Выберите один из доступных способов создания:

- При помощи элемента управления .
- Из соответствующего пункта контекстного меню текущего узла.

3. Заполните все необходимые поля и нажмите **Создать**.

> Внутри подразделений допускается создание дочерних подразделений.

Изменить подразделение

Чтобы отредактировать ранее созданный узел подразделения:

1. На вкладке **Организации** в левой области справочника выберите узел из иерархического списка.
2. Выберите Подразделение.
3. Выберите один из доступных способов изменения:
   - При помощи элемента управления на вкладке *Информация* в правой области справочника.
   - Из соответствующего пункта контекстного меню текущего узла.
4. Измените необходимые поля и нажмите **Сохранить**.

Удалить подразделение

Чтобы удалить ранее созданный узел подразделения:

1. На вкладке *Организации* в левой области справочника выберите узел из иерархического списка.
2. Вызовите контекстное меню на выбранном узле.
3. Выберите соответствующий пункт из контекстного меню, чтобы удалить узел.
4. Появится предупреждение, подтвердите действие кнопкой **ОК**.

> Организации и подразделения удаляются со всеми дочерними узлами, включая сотрудников.

Перечень полей карточки подразделения контрагента

Основная информация о подразделении может быть заполнена при создании нового подразделения или изменена после создания.

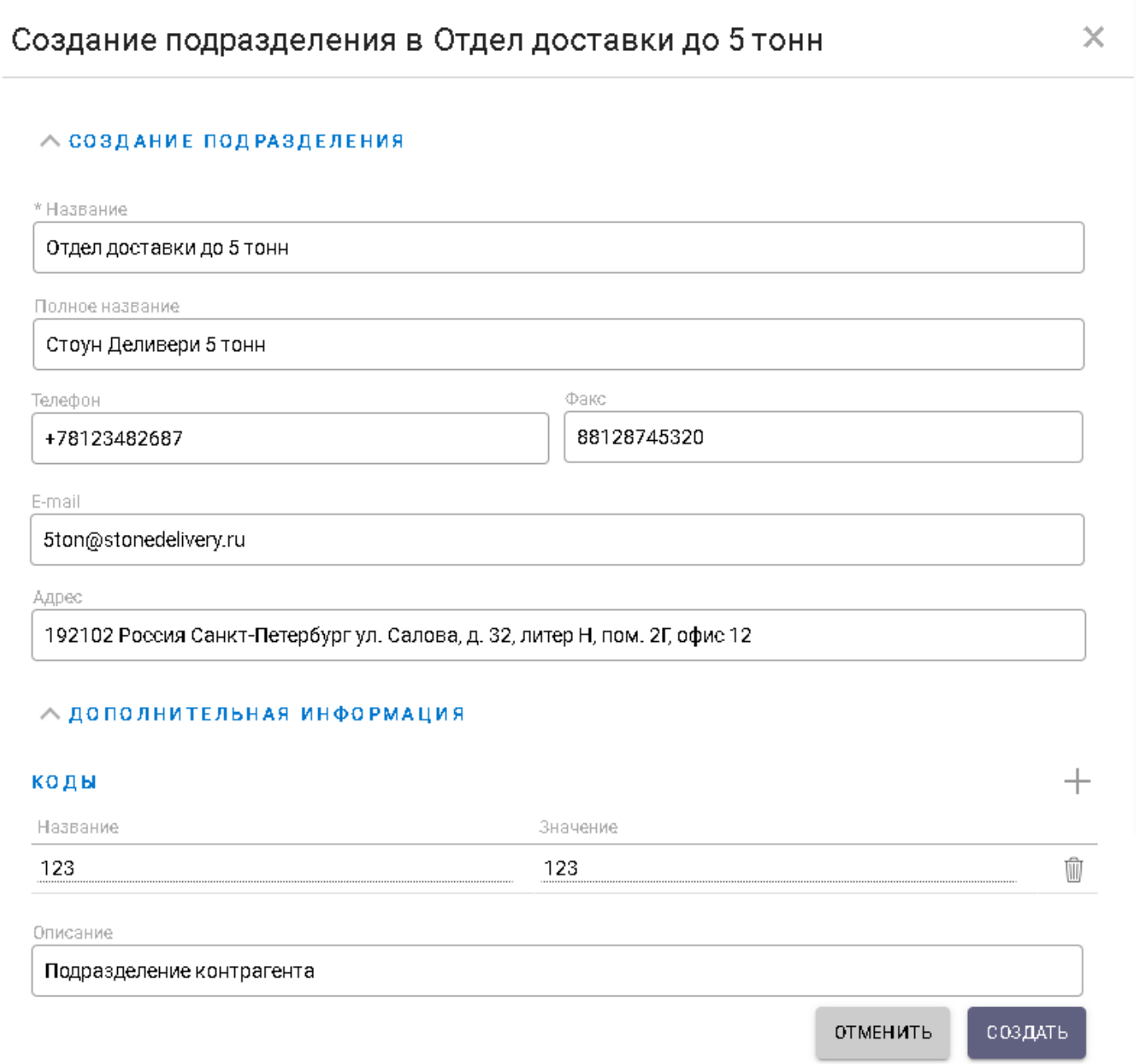


Рисунок 295. Основная информация о подразделении контрагента

Информация о подразделении

Название

В поле указывается сокращенное название, которое будет отображаться в справочнике Docsvision.

Полное название

В поле указывается полное официальное название.

Телефон

В поле указывается номер телефона для связи.

Факс

В поле указывается номер факса.

E-mail

В поле указывается адрес для отправки электронных писем.

Адрес

В поле указывается фактический адрес.

Поле заполняется по аналогии с полем Адрес в данных организации.

Дополнительная информация

Коды

В секции *Коды* вкладки *Дополнительно* нажмите кнопку **Добавить** +, чтобы ввести государственные коды и указать их значения.

Удалить коды можно при помощи кнопки 🗑.

Описание

Произвольное описание в текстовом формате.

Работа с группами

В справочнике контрагентов реализована возможность создания, изменения и удаления групп. После создания в группу можно добавить организации и подразделения или оставить группу пустой.

Посмотреть информацию о группе

Вся информация о выбранной группе доступна в правой области справочника, во вкладке *Информация* или из пункта *Информация* в контекстном меню.

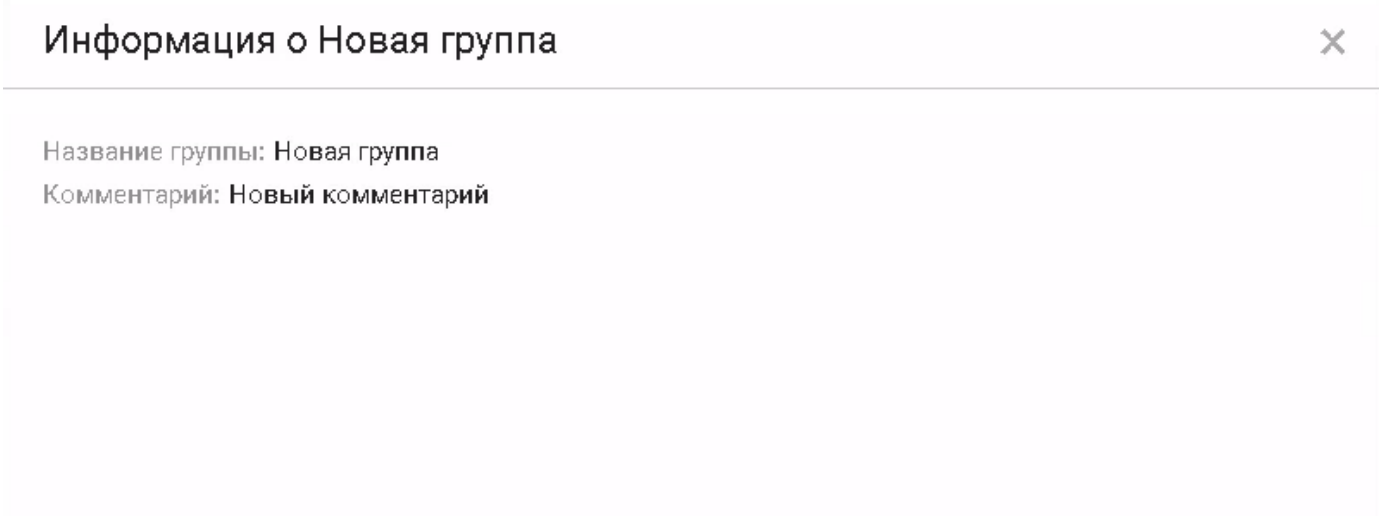


Рисунок 296. Просмотр информации о группе

Внутри одного узла группы может быть создано неограниченное количество других узлов групп с неограниченной глубиной вложенности.

> Возможность создания, изменения, удаления и просмотра узлов, а также добавления или изменения числа сотрудников в выбранном узле может быть ограничена настройками безопасности.

Создать новую группу

Чтобы создать новый узел группы:

1. В левой области справочника выберите вкладку **Группы**.
2. Выберите узел, в который необходимо добавить новую группу.
3. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего узла.
4. Заполните все необходимые поля и нажмите **Создать**.

Изменить группу

Чтобы изменить ранее созданный узел группы:

1. На вкладке *Группы* выберите группу, которую хотите изменить.
2. Выберите один из доступных способов изменения:
   - При помощи элемента управления на вкладке *Информация* в правой области справочника.
   - Из соответствующего пункта контекстного меню текущего узла.

3. Измените необходимые поля и нажмите **Сохранить**.

Удалить группу

Чтобы удалить ранее созданный узел группы:

1. На вкладке *Группы* в левой области справочника выберите из иерархического списка узел группы, который хотите удалить.
2. Вызовите контекстное меню на выбранном узле.
3. Выберите соответствующий пункт из контекстного меню, чтобы удалить узел.
4. Появится предупреждение, подтвердите действие кнопкой **ОК**.



При удалении группы контрагенты, входящие в группу, не удаляются, но перестают относиться к удалённой группе.

Скопировать и переместить группы контрагентов

Группы контрагентов могут быть скопированы, вырезаны и удалены при помощи команд контекстного меню.

Чтобы скопировать или переместить группы контрагентов:

1. Выберите в левой области дерева узел групп, который требуется скопировать или переместить.
2. Вызовите контекстное меню узла нажатием правой кнопки мыши.
3. В контекстном меню выберите соответствующую команду:
   - Вырезать — отмечает выбранную группу для последующего перемещения вместе со всеми подгруппами. Принадлежность организаций и подразделений к этим группам сохраняется: после вставки они отображаются в составе групп на новом месте, а их положение в дереве организаций не изменяется.
   - Копировать — узел копируется вместе с подчинёнными группами. Организации и подразделения не копируются.
4. Выберите в дереве целевой узел.
5. Вызовите контекстное меню на целевом узле и выберите команду *Вставить*.

Перечень полей карточки группы

Карточка группы содержит два поля:

Название	Имя группы, которое будет отображаться в справочнике Docsvision.
Комментарий	Любая дополнительная информация, относящаяся к данной группе.



При работе с полем Комментарий поддерживается HTML синтаксис.

Добавить контрагентов в группу и удалить из группы

В справочнике контрагентов реализована возможность группировать контрагентов. В группы могут быть добавлены только организации и подразделения контрагентов. Сотрудники контрагентов не могут быть добавлены в группы.

Добавление контрагентов в группу

Чтобы добавить подразделение или организаци контрагента в группу:

1. В левой области справочника откройте вкладку *Организации*.
2. Выберите узел, который необходимо добавить в группу. Можно выбрать организацию или подразделение. Выбор сотрудников контрагентов не поддерживается.
3. Вызовите контекстное меню выбранного узла.
4. Выберите команду *Добавить в группу*.
5. В появившемся окне *Добавление в группы* выберите одну или несколько групп, в которые необходимо добавить организации или подразделения контрагентов.

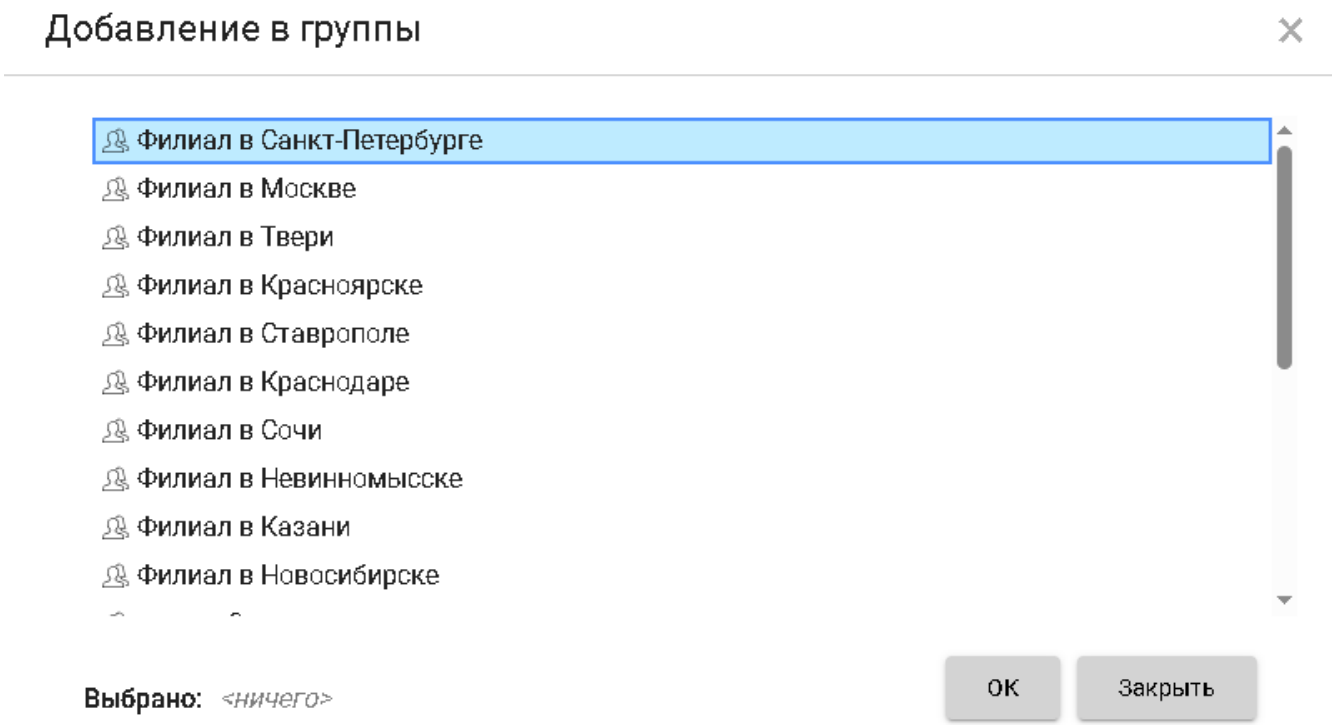


Рисунок 297. Добавление контрагентов в группы

6. Нажмите кнопку **ОК**, чтобы добавить контрагентов в группы или **Закрыть**, чтобы отменить добавление.
7. Контрагент отобразится в группе (группах).

   Если в группу отдельно были добавлены организация и её подразделение, каждый из этих элементов отображается как самостоятельный участник группы.

Создание сотрудника контрагента из группы

Внутри организаций и подразделений, входящих в группы, можно создать сотрудников.

Чтобы создать сотрудников внутри организации и подразделения контрагента:

1. В левой области справочника откройте вкладку *Группы*.
2. Выберите группу, найдите организацию или подразделение, внутри которых требуется создать сотрудника.
3. Вызовите контекстное меню организации или подразделения.
4. Выберите пункт *Создать сотрудника*.
5. Заполните поля карточки сотрудника и нажмите кнопку **Создать**.

Удаление контрагентов из группы

Чтобы удалить организацию или подразделение контрагента из группы:

1. В левой области справочника откройте вкладку *Группы*.
2. Выберите группу, в которой находится организация или подразделение.
3. Вызовите контекстное меню организации или подразделения.
4. Выберите пункт *Удалить из группы*.

   Контрагент будет удалён из группы.

Также контрагентов можно удалить из групп при повторном открытии окна *Добавление в группы*. В окне будут отображены все группы, в которыеранее были добавлены организации или подразделения контрагентов. Чтобы удалить контрагента из группы, нажмите её значок в дереве или кнопку удаления рядом с ее названием в области *Выбрано*.

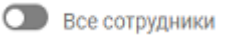
Работа с сотрудниками контрагентов

Содержимое вкладки *Сотрудники*, отображаемой в правой области справочника, зависит от узла, выбранного в левой области. На вкладке *Сотрудники* отображаются не все сотрудники контрагентов, доступные в справочнике, а только те, которые относятся к выбранной организации, подразделению и т.д.

Подробно о том, как заполнить поля на вкладках карточки сотрудника и информацию о них, см. в пункте "Поля и вкладки карточки сотрудника контрагента".

Посмотреть информацию о сотруднике контрагента

Когда узел из левой области находится в фокусе, во вкладке *Сотрудники* отображаются колонки по умолчанию: *ФИО*, *Должность*, *Рабочий телефон*, *E-mail*. Все сотрудники отсортированы в алфавитном порядке по первому столбцу.

При активации переключателя 

во вкладке отображаются все сотрудники, подчинённые текущему выбранному узлу, например, сотрудники организации и всех входящих в организацию подразделений.

Вся информация о сотруднике контрагента доступна из пункта *Информация* в контекстном меню или по клику на строку сотрудника в правой области справочника.

В поле *Руководитель подразделения* отображается руководитель ближайшего узла организации или подразделения, к которому относится сотрудник.

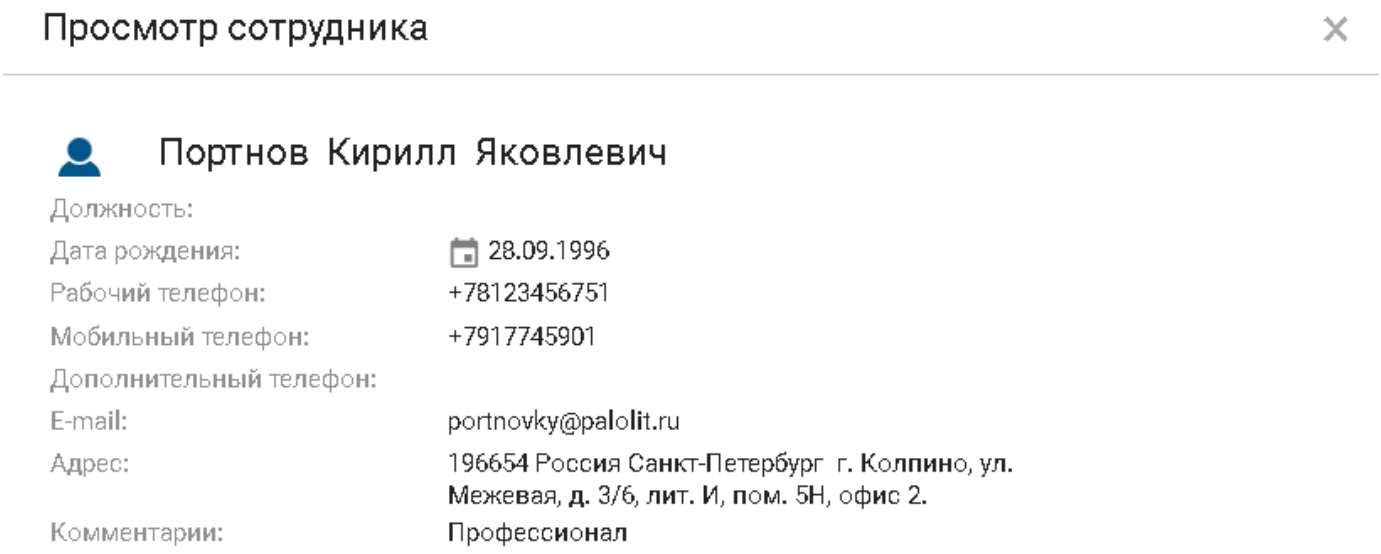


Рисунок 298. Просмотр информации о сотруднике контрагента

Возможность создания, изменения, удаления и просмотра сотрудников может быть ограничена общими настройками безопасности справочника.

Краткая информация о сотруднике контрагента

Для карточки сотрудника контрагента может использоваться разметка в режиме Справка. Такая разметка позволяет отображать краткую информацию о текущем состоянии сотрудника.

Настроить отображение краткой информации о сотруднике можно в программе Конструктор Web-разметок, настроив разметку **BackOffice** › **Справочники** › **Справочник контрагентов** › **Подразделения** › **Сотрудники** с режимом *Справка*.

Краткая информация о сотруднике контрагента выводится при наведении на строку сотрудника во всех разметках, где используется один из следующих элементов управления:

- Партнер.
- Подразделение контрагента.
- Дата/время.
- Строка конструктора справочников.
- Нумератор.

- Дело.

Краткая информация о сотруднике отображает основную информацию:

- ФИО.
- Дату рождения.
- Рабочий, мобильный и дополнительный телефон.
- E-mail.
- Адрес.
- Комментарий.

Создать нового сотрудника контрагента

Возможность создания новых сотрудников доступна, только если в левой области открыта вкладка *Организации* и выбран узел организации или подразделения. Для каждой учётной записи в ActiveDirectory может быть создана только одна карточка сотрудника в *Справочнике контрагентов*.

Чтобы создать нового сотрудника:

1. Откройте справочник контрагентов.
2. На вкладке *Организации* в левой области справочника выберите узел из иерархического списка.
3. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего узла.

> Название организации или подразделения, к которым будет добавлен сотрудник, указано в заголовке окна.

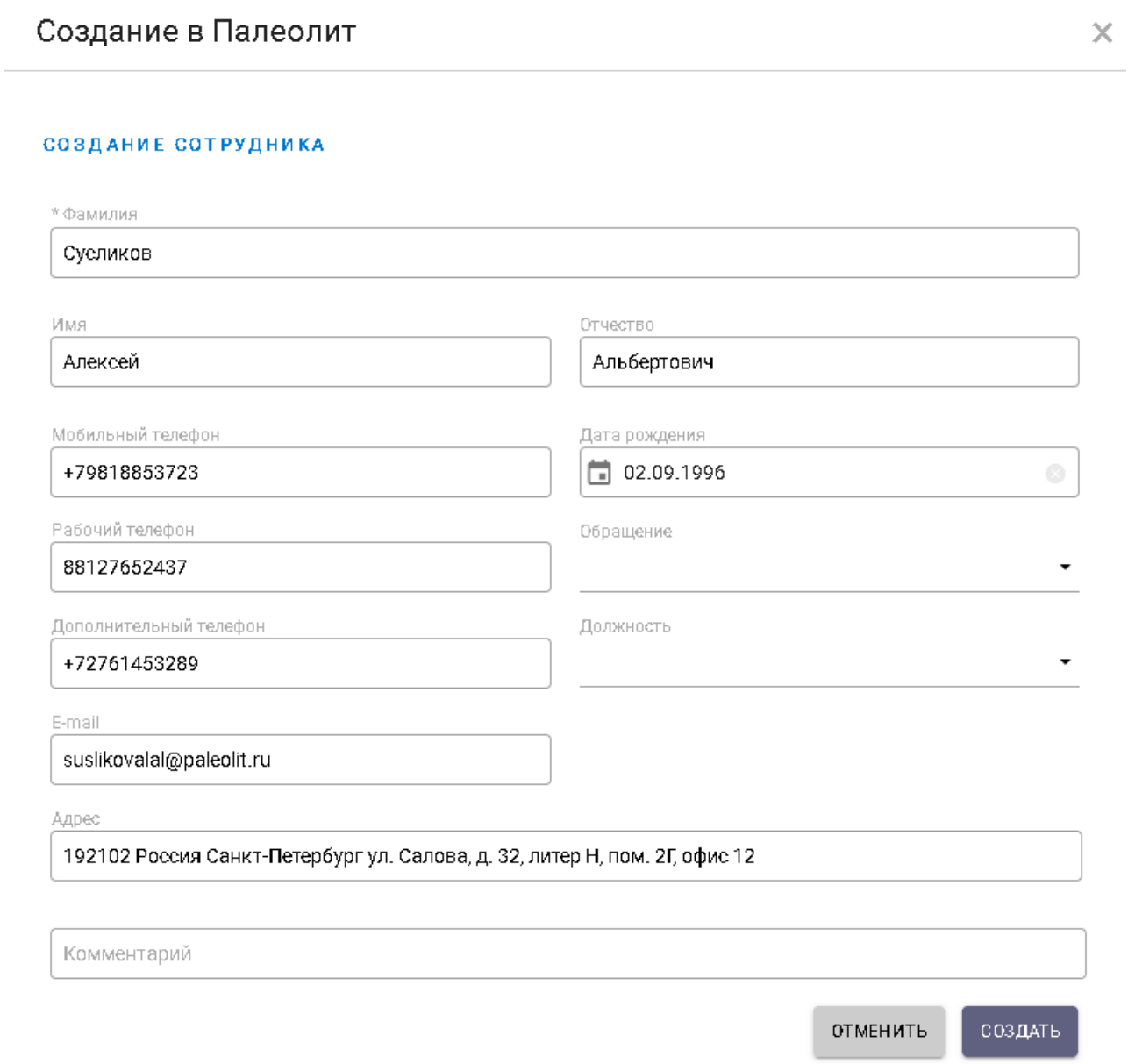


Рисунок 299. Создание нового сотрудника

4. Последовательно заполните доступные поля карточки сотрудника и нажмите **Создать**.

Изменить сведения о сотруднике контрагента

Чтобы изменить сведения о сотруднике:

1. Откройте справочник контрагентов на нужной вкладке.
2. Выберите нужного сотрудника контрагента из списка в правой области.
3. Вызовите контекстное меню на нужном сотруднике и выберите *Редактировать*.

4. Внесите необходимые изменения в доступные поля карточки сотрудника.
5. Выйдите из режима редактирования, нажав **Сохранить**.

Удалить сотрудника контрагента

Чтобы удалить ранее созданного сотрудника контрагента:

1. В правой области справочника выберите сотрудника, которого необходимо удалить.
2. Вызовите контекстное меню на выбранном сотруднике.
3. Выберите в контекстном меню пункт *Удалить*.
4. Появится предупреждение, подтвердите действие кнопкой **ОК**.



Сотрудник будет полностью удалён из справочника без возможности восстановления.

Поля и вкладки карточки сотрудника контрагента

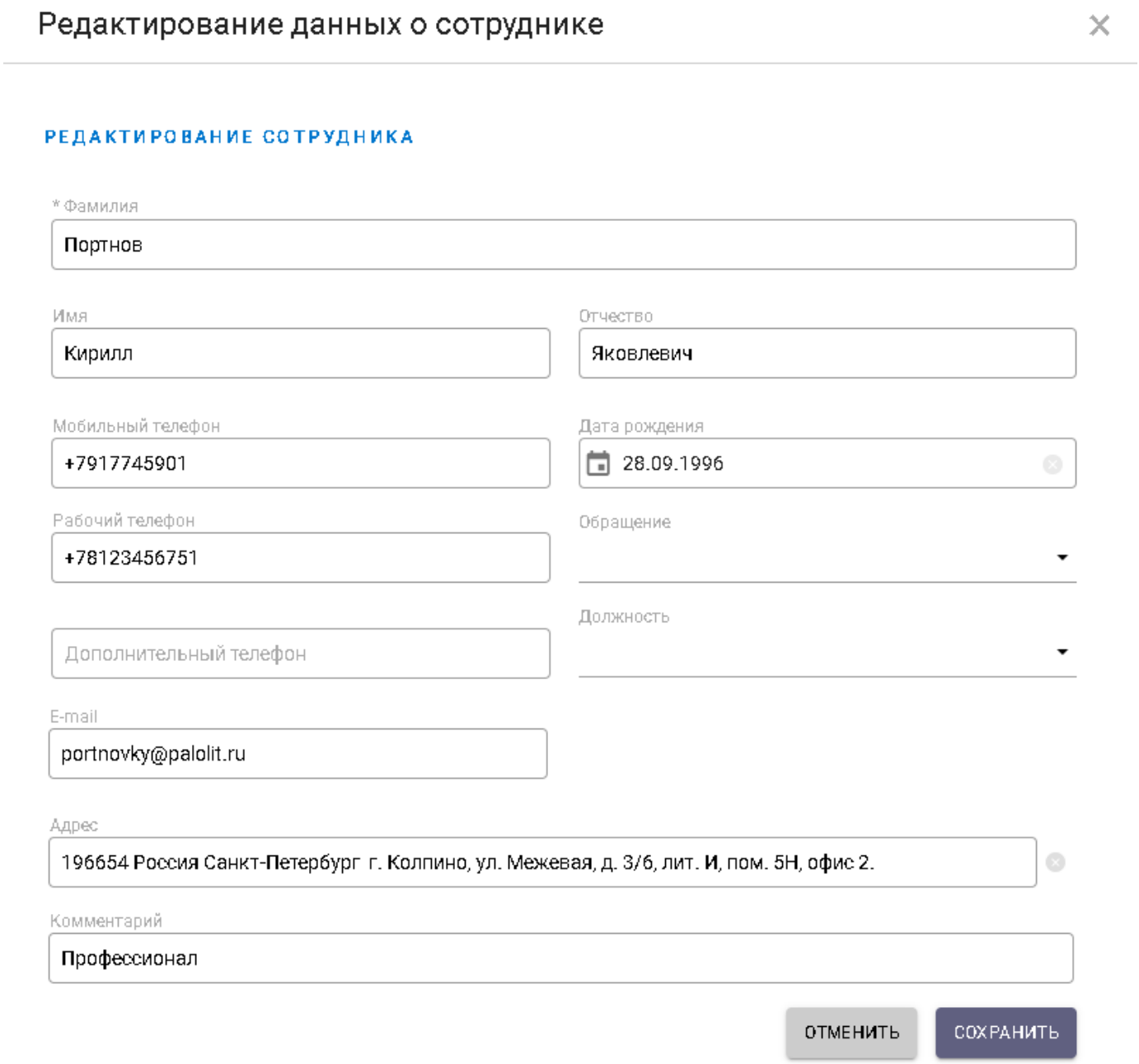


Рисунок 300. Общая информация и вкладки карточки сотрудника контрагента в режиме редактирования

Фамилия, Имя и Отчество	Поля для указания ФИО сотрудника. Обязательным для заполнения является только поле Фамилия.
Мобильный, рабочий, дополнительный телефон	В поле указываются номера телефонов для связи с сотрудником.
Дата рождения	Можно выбрать дату рождения сотрудника из календаря.

Обращение	Форма официального обращения к сотруднику выбирается из *Справочника обращений*.
Должность	Должность контрагента из собственного справочника доложностей для *Справочника контрагентов*.
E-mail	Адрес электронной почты сотрудника контрагента. Можно указать несколько адресов без пробелов, разделив их символом `;`.
Адрес	В поле указывается фактический адрес. Поле заполняется по аналогии с полем Адрес в данных организации.
Комментарий	Поле предназначено для добавления дополнительной информации к карточке сотрудника. Поле поддерживает HTML синтаксис.

Поиск по справочнику контрагентов

В справочнике реализована функция поиска. Поиск активируется нажатием на в верхней части пользовательского интерфейса Web-клиента. Результата поиска можно показать или скрыть нажатием на кнопку . Если список найденного слишком большой, можно развернуть и свернуть результаты поиска кнопками .



Если количество результатов поиска превышает 500, они загружаются порциями. Первая порция — 500, следующая будет загружена при прокрутке. Поиск показывает только найденные узлы и их родителей, дерево возвращается к исходному виду после сброса поиска.

В выпадающем списке слева от поисковой строки можно выбрать область поиска только по *Организациям и Подразделениям*, *Сотрудникам* и *Группам*:

- Поиск по *Организациям и Подразделениям* осуществляется по полям *Наименование*, *Полное наименование*, *ИНН*.
- Поиск по сотрудникам осуществляется по полям *Фамилия*, *Имя*, *Отчество*.
- Поиск по *Группам* осуществляется по полю *Название* группы.

В результатах поиска отображаются все найденные результаты, а также общее

количество найденных результатов. Цветом выделяются совпадения с поисковым запросом. Нажатие на результат поиска переводит на соответствующий узел справочника.

Если поиск осуществлялся по *Организациям и Подразделениям* или *Группам*, структура справочника раскрывается до выбранного раздела и фокус смещается на найденный узел.

Если поиск осуществлялся по сотрудникам, то структура справочника раскрывается до родительского узла, и в правой области выбирается найденный сотрудник.

Настройки безопасности справочника контрагентов

В зависимости от текущих прав доступа пользователя, справочник контрагентов:

- Откроется в режиме просмотра, если пользователю запрещено редактирование справочника.
- Откроется в режиме редактирования, если пользователю разрешено редактирование справочника.
- Не откроется, и появится предупреждение об отсутствии прав доступа, если пользователь не имеет прав на просмотр справочника.

Настройки дискреционной безопасности для всего справочника контрагентов

Администратор может задать права доступа для всего справочника при помощи элемента управления , расположенного над правой областью.

Если администратор Web-клиента установил ограничение для работы со справочником контрагентов, возможности работы с узлами справочника, включая *Организации*, *Подразделения*, *Группы*, а также работа с *Сотрудниками* могут быть ограничены. Данное ограничение также распространяется на возможность поиска по справочнику.

Чтобы настроить дискреционную безопасность для всего справочника:

1. Перейдите в справочник по ссылке из основного меню.
2. Нажмите .
3. В появившемся окне задайте параметры доступа для пользователей согласно инструкции, данной в пункте Настройка безопасности объектов.
4. Подтвердите настройки, нажав **Применить** или **ОК**.



Настройки дискреционной безопасности наследуются дочерними элементами. Это означает, что, если наследование не отключено, настройки дискреционной безопасности для подразделений и организаций будут унаследованы от родительских узлов.

При одновременном редактировании прав доступа к справочнику с разных окон или вкладок браузера изменения сохранятся только из последнего закрытого окна.

Настройки дискреционной безопасности для узлов справочника контрагентов

Администратор имеет возможность задать права доступа к следующим узлам справочника: *Организации*, *Подразделения*, *Группы*.

Чтобы задать права доступа к узлу справочника:

1. Выберите необходимый узел.
2. Вызовите контекстное меню на выбранном узле.
3. В контекстном меню выберите пункт *Права доступа*.
4. В появившемся окне задайте параметры доступа для пользователей согласно инструкции, данной в пункте Настройка безопасности объектов.
5. Подтвердите настройки, нажав **Применить** или **ОК**.

Копирование узлов справочника контрагентов

В справочнике реализована возможность *Копировать* и *Вырезать* узлы контрагентов. Операции копирования или вырезания могут быть вызваны из контекстного меню выбранного узла.



Поддерживается копирование всех узлов, кроме сотрудников. Сотрудников можно только вырезать и переместить в организацию или подразделение.

Вокруг скопированных или вырезанных узлов и сотрудников появляется пунктирная рамка. Вырезанный узел дополнительно обозначается серым цветом. Цвет скопированных узлов и сотрудников не меняется.

Копировать

Операция копирует только иерархию узла, но не копирует сотрудников. Чтобы переместить узлы вместе с сотрудниками, воспользуйтесь операцией *Вырезать*.

Вырезать

Операция перемещает узел со всеми дочерними узлами, включая сотрудников. При выполнении операции исходный узел удаляется в старом месте и перемещается в новое. Вырезать можно также отдельных сотрудников.

Вставить

Операция завершает копирование или вырезание. Операция недоступна, пока раздел не был скопирован или вырезан.

Операция **Вставить** отправляет запрос на сервер. Это значит, что узел будет скопирован или вырезан только после выбора *Вставить*.

Чтобы скопировать или переместить узел:

1. Выберите узел-источник, который желаете скопировать или вырезать.
2. Вызовите контекстное меню узла-источника.
3. Из контекстного меню узла выберите *Копировать* или *Вырезать*.



Скопированный или вырезанный узел или сотрудник контрагента помещаются в буфер обмена. После этого справа от узла или сотрудника появляется кнопка **Отменить**. Вместо пунктов *Копировать* или *Вырезать* контекстного меню появляется пункт *Отменить*. Нажатие на эту кнопку или пункт в контекстном меню очищает буфер обмена.

4. Вызовите контекстное меню на целевом узле, в который необходимо добавить или переместить текущий.
5. В контекстном меню выберите пункт *Вставить*.



Перемещать и копировать организации и подразделения контрагентов можно только в организации или подразделения, группы — только в группы.

Незавершённые операции будут отменены в следующих случаях:

- При повторном открытии справочника или обновлении страницы.
- При переходе на новую вкладку или при закрытии текущей.
- При нажатии кнопки `Esc` на клавиатуре.
- При выборе пункта *Отменить* из контекстного меню.

- При нажатии кнопки **Отменить** справа от узла.
- При копировании или вырезании другого узла.

При выполнении операций проверяются права доступа на редактирование узлов справочника.

Чтобы вырезать узлы и сотрудников контрагентов, пользователь должен иметь права на редактирование исходного и родительского узлов. Чтобы вставить узлы и сотрудников, пользователь должен иметь права на редактирование целевого узла.

После выполнения операций, настройки прав доступа исходного узла будут заменены настройками, унаследованными от нового родительского узла.

Справочник номенклатуры дел

Справочник номенклатуры дел предназначен для упорядочивания документального фонда в виде систематизированного перечня заголовков дел, с указанием сроков их хранения, оформленный в установленном порядке.



Справочник номенклатура дел становится доступным только при наличии следующих установленных компонентов:

1. Модуль *Управление архивом* и расширения Управление архивом Web-клиента.
2. Web-клиент и серверное расширение Web-клиента.

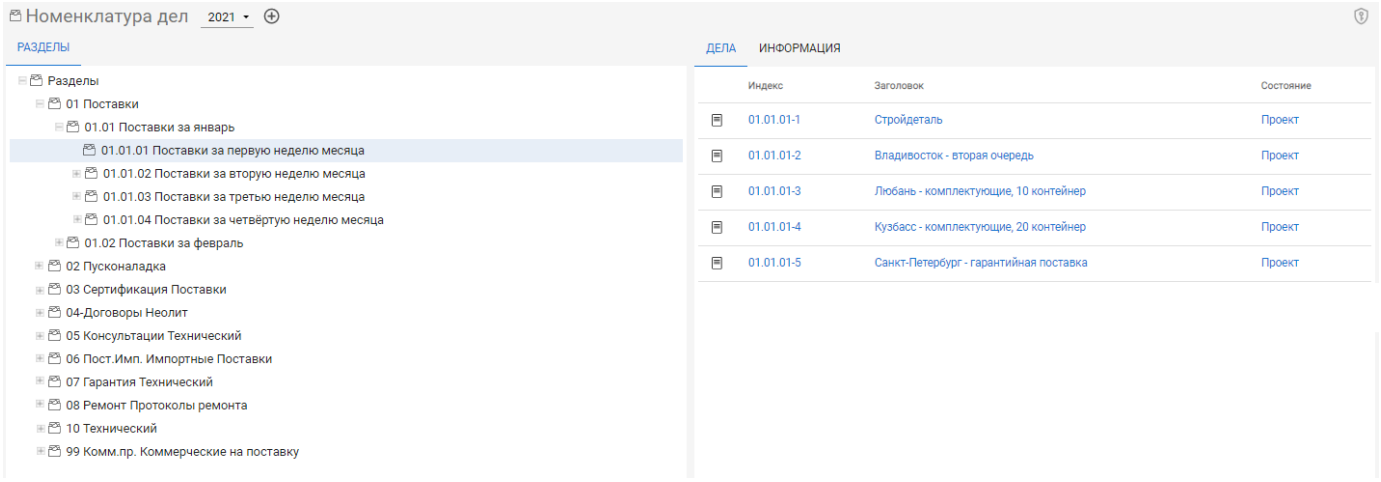


Рисунок 301. Справочник номенклатура дел

Справочник номенклатуры дел визуально поделён на две области. В левой области располагается иерархическая структура разделов, дел и томов

выбранного года в рабочем месте сотрудника, не являющегося архивистом, или иерархическая структура фондов, описей, дел и томов в рабочем месте сотрудника, являющегося архивистом. При наведении на границу областей изменяется вид указателя, границу можно сжать, потянув вправо или влево. Допускается сжатие вплоть до 10% от исходного.

Над левой областью находится выпадающий список для выбора года и элемент управления ⊕ для добавления нового года.

В правой области отображается информация о карточке, на которой установлен фокус мыши.

Над правой областью расположен элемент управления, позволяющий задать общие настройки дискреционной безопасности для справочника.

Создать год номенклатуры дел

Справочник ведётся по годам. Пока не создан год, функции работы со справочником недоступны.

Года всегда создаются в состоянии Проект.

Чтобы создать новый год номенклатуры дел:

1. Перейдите в справочник номенклатуры дел по ссылке из основного меню (может быть отключена администратором системы Docsvision).
2. Справа от выпадающего списка нажмите на кнопку ⊕, чтобы отобразить окно с диалогом создания нового года.

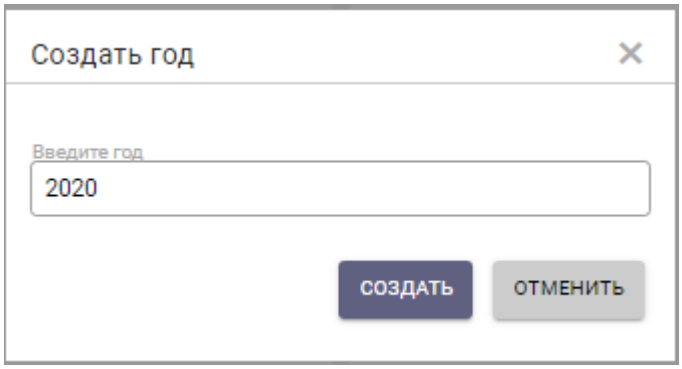


Рисунок 302. Создание нового года номенклатуры дел

3. Введите год, затем нажмите **Создать**.

После создания год нельзя удалить или изменить.

Также нельзя создать новый год внутри текущего или создать

два одинаковых года.

Работа с разделами номенклатуры дел

Для рабочего места сотрудника, не являющегося архивистом справочник номенклатуры дел представляет собой иерархическую структуру. Верхним разделом справочника является год, в каждом году по умолчанию создаётся *системный раздел Разделы*, являющийся родителем для остальных.

Системный раздел может содержать неограниченное количество *корневых разделов*. Каждый корневой раздел может иметь неограниченное число *подразделов* первого уровня.

В подразделах первого уровня может быть создано произвольное количество подразделов следующих уровней. Глубина вложенности не ограничена.

Чтобы раскрыть раздел, нажмите на ⊞ слева от его названия. Чтобы свернуть раздел, нажмите на ⊟. При нажатии левой кнопкой мыши, раздел переходит в фокус, в правой области справочника отображается информация по разделу.

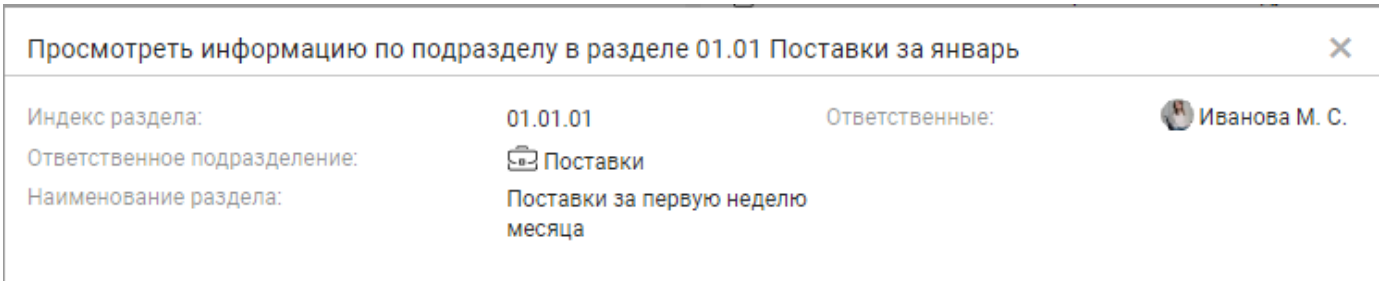


Рисунок 303. Окно просмотра информации о разделе

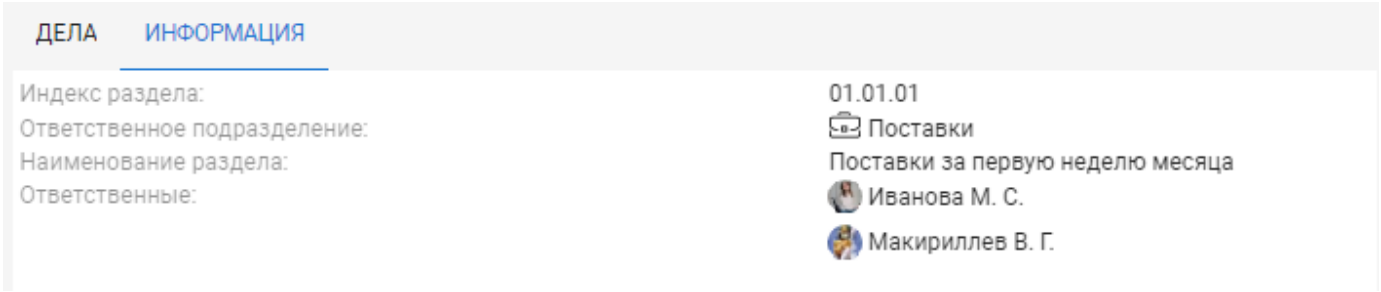


Рисунок 304. Вкладка "Информация"

Создать новый раздел номенклатуры дел

Разделы создаются в состоянии Проект.

Чтобы создать новый раздел:

1. Из выпадающего списка в левой части номенклатуры дел выберите год, в который желаете добавить раздел.

2. Установите фокус на системном разделе или разделе, к которому необходимо создать подраздел.
3. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего раздела.
4. Заполните поля нового раздела.
5. Создайте желаемое количество разделов.
6. Создавайте подразделы, повторяя шаги 1-3. Создать подраздел можно при помощи элемента управления или из пункта *Создать подраздел* контекстного меню.
7. Создавайте дочерние разделы до получения желаемой иерархической структуры.

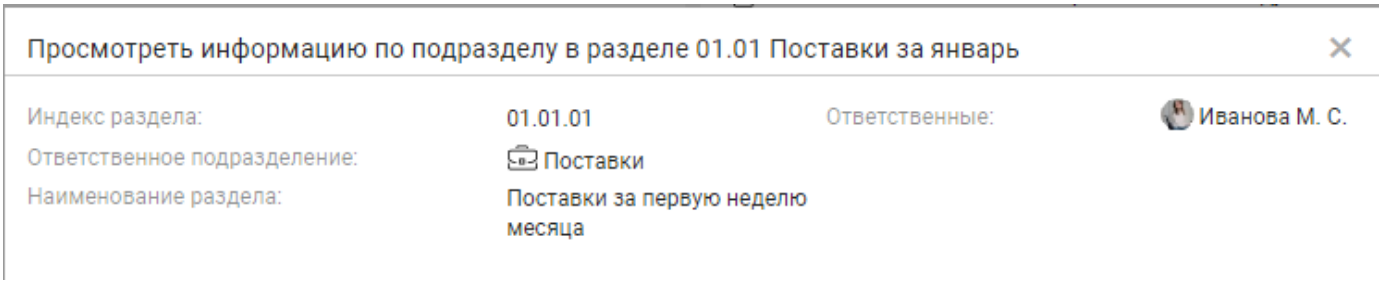


Рисунок 305. Окно просмотра информации о разделе

Изменение раздела номенклатуры дел

После создания раздел справочника может быть изменён при наличии соответствующих прав доступа.

Чтобы отредактировать раздел:

1. Выберите раздел номенклатуры дел, который нужно изменить.
2. Измените раздел одним из доступных способов:
   - Выберите в контекстном меню раздела пункт *Редактировать*.
   - При помощи элемента управления в правой области справочника.
3. Измените поля раздела.
4. Сохраните изменения, нажав кнопку **Сохранить**.

Копирование разделов

Разделы могут быть скопированы в разрезе разделов, подразделов, дел и томов.

Чтобы копировать раздел:

1. Выберите раздел номенклатуры дел, который нужно скопировать, включая входящие в его состав подразделы, дела и тома.
2. Вызовите контекстное меню раздела и выберите команду:
   - *Копировать* — будет скопирована иерархия раздела.
3. Выберите раздел, в который планируется поместить копию. Дерево разделов, включая подразделы, дела и тома, будет помещено в копию.
4. При копировании и помещении скопированного дерева разделов, включая подразделы, дела и тома:
   - Карточки разделов, подразделов будут скопированы в состоянии Проект
   - Карточки дел будут скопированы в состоянии Проект, если в карточке дела не установлен признак *Переходящее дело*
   - Если в карточке дела не установлен признак *Переходящее дело*, по карточке дела также будет скопирована информация кроме:
     - Значений из ЭУ Контрол табличных данных, который расположен на вкладке *Документы, списанные в дело*
     - Значения из поля *Дата закрытия дела*
     - Значения из поля *Место хранения*
     - Значения из поля *Опись №*
     - Значения из поля *Протокол ЭПК №*
     - Значения из поля *Фонд*
     - Значения из поля *Фонд*
     - Значения из поля *Дата начала хранения*
     - Значений в разделе *Журнал передач*
   - Значение следующих полей карточки дела будет пересчитано:
     - Значение поля *Дата начала хранения* заполнится датой 01.01. год номенклатуры дел, в которую копируется дерево разделов, включая подразделы, дела и тома
     - Значение поля *Дата окончания хранения* заполнится как: 01.01. год, следующий за годом, к которому относится родительский для дела раздел + значение поля *Срок (лет)* строки справочника, которая указана в поле *Срок хранения*

- Значение поля *Срок хранения в оперативном архиве* заполнится как: 31.12.год, следующий за годом, к которому относится родительский для дела раздел
- Значение поля *Дата заведения* дела заполнится текущей датой

◦ Если в карточке дела установлен признак *Переходящее дело* и указано состояние, отличное от состояния *Закрыто*, то копируется:

- В копируемую карточку дела с установленным признаком *Переходящее дело* в скрытое поле *Родительский раздел* добавляется также системный идентификатор текущей карточки раздела, в которую входит карточка копируемого дела, в копируемом дереве разделов.

  Таким образом карточка дела входит сразу в несколько карточек разделов в разрезе номенклатур разных лет.

◦ Если в карточке дела установлен признак *Переходящее дело* и указано состояние Закрыто, то карточка дела не копируется. Таким образом карточка переходящего дела остается в номенклатуре дел тех лет, которые содержат карточки разделов, системный идентификатор которых указаны в скрытом поле *Родительский раздел* карточки переходящего дела. После копирования дерева разделов карточку переходящего дела необходимо создать

◦ Если у карточки дела не установлен признак *Переходящее дело* и у карточки дела есть дочерняя карточка тома, то карточка тома копируется в состоянии Проект

◦ Если у карточки дела не установлен признак *Переходящее дело* и у карточки дела есть дочерняя карточка тома, то по карточке тома будет скопирована вся информация кроме Значений из ЭУ Контрол табличных данных, который расположен на вкладке *Документы, списанные в дело*

◦ Если у карточки дела установлен признак *Переходящее дело* и указано состояние, отличное от состояния Закрыто, и у карточки дела есть дочерняя карточка тома, то в копируемую карточку тома в скрытое поле *Ссылка на родительское Дело* добавляется также системный идентификатор текущей карточки дела, в которую входит карточка копируемого тома, в копируемом дереве разделов

◦ Если у карточки дела установлен признак *Переходящее дело* и указано состояние Закрыто, и у карточки дела есть дочерняя карточка тома, то карточки томов не копируются

Удаление раздела номенклатуры дел

- Разделы удаляются со всеми дочерними элементами, включая дела и тома. При удалении дела или тома, списанные в него документы не удаляются. Если в документе была ссылка на дело или том, такая ссылка будет удалена.
- Удаление возможно только при наличии соответствующих прав доступа.

Чтобы удалить раздел справочника:

1. Выберите раздел номенклатуры, который нужно удалить.
2. Выберите в контекстном меню раздела пункт *Удалить*.
3. Появится предупреждение, подтвердите действие кнопкой **ОК**.

Перечень полей раздела номенклатуры дел

Каждый раздел номенклатуры дел содержит поля для заполнения:

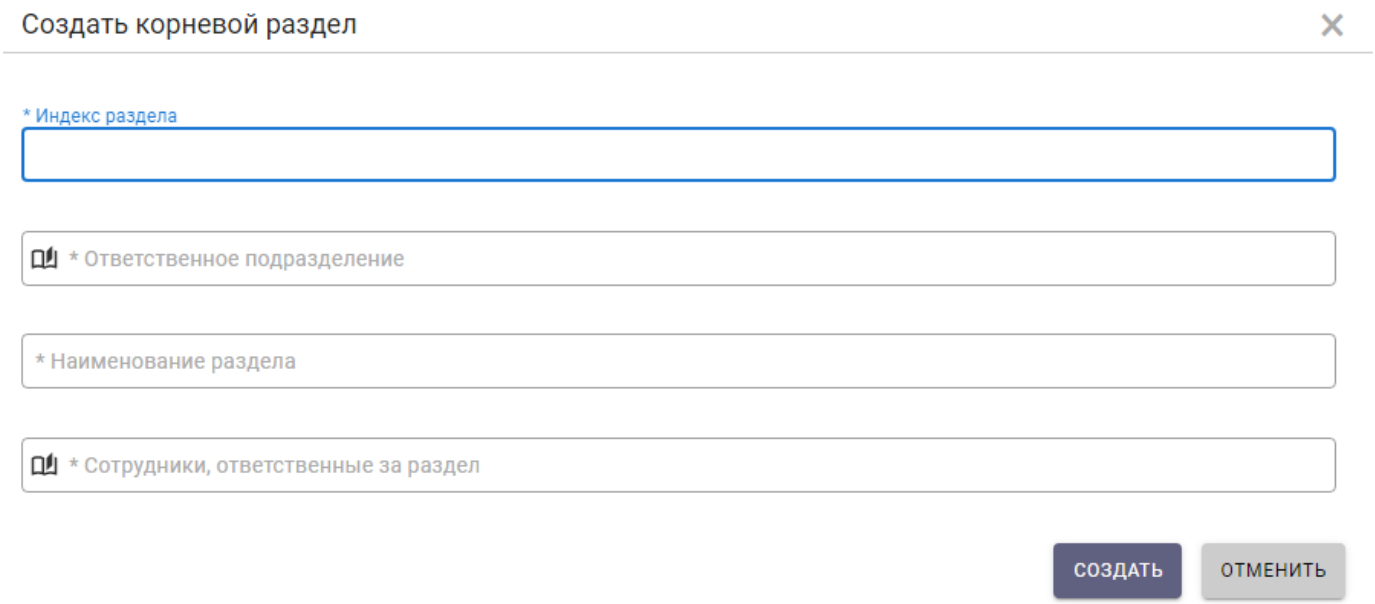


Рисунок 306. Стандартные поля раздела номенклатуры дел

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Состояние	Состояние	Да	Нет	Текущее состояние карточки раздела. При создании состояние Проект. *Необходимо для отображения состояния раздела номенклатуры дел:* • Если состояние Проект, номенклатура дел еще не утверждена, списывать документы в дела номенклатуры нельзя. • Если состояние Утвержден, номенклатура дел утверждена. С 1 января года номенклатуры дел в дела номенклатуры дел можно списывать документы. • Если состояние Закрыт, делопроизводство по документам дел раздела завершено. Списывать документы в дела текущего раздела нельзя. *Состояние карточки раздела изменяется:* • При утверждении раздела по кнопке

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Блок *Основные реквизиты*				
Родительский раздел	Карточка	Да	Нет	Поле в схеме данных. Системный идентификатор системного раздела или родительского раздела, в состав которого входит текущая карточка раздела. Необходимо для построения дерева номенклатуры дел в рабочем месте сотрудника, не являющегося архивистом.
Подразделение	Подразделение	Да	Да	Выбор подразделения из справочника сотрудников. Необходимо для формирования номенклатуры дел структурного типа, то есть номенклатуры дел на основе структуры организации.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Индекс раздела	Строка	Да	Нет	Заполняется автоматически после заполнения поля *Подразделение* выбором значения из справочника сотрудников: значение из поля *Индекс организации* или *Индекс подразделения*. При очищении поля *Подразделение* значение поля *Индекс раздела* также очищается. Необходимо для систематизации и структурирования дел в справочнике номенклатуры дел.
Заголовок раздела	Строка	Да	Да	Вручную. Необходимо для систематизации и структурирования дел в справочнике номенклатуры дел.
Таблица *Ответственные*				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Ответственный	Сотрудник	Да	Да	Выбор значения из справочника сотрудников. .Необходимо для указания сотрудников, которые является ответственными за: * Формирование номенклатуры дел своего структурного подразделения (или всей организации); * Передачу дел долговременного срока хранения (свыше 10 лет срока хранения или постоянного срока хранения) из структурных подразделений в архив организации; * Уничтожение дел оперативного срока хранения (менее 10 лет срока хранения) в структурных подразделениях.
Вкладка *Дела*				
Таблица *Дела* (Контрол табличных данных)				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Индекс дела	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки дела, у которой в скрытом поле *Ссылка на родительский раздел* указан системный идентификатор текущей карточки раздела • Если по результатам поиска карточка дела будет найдена, то необходимо получить значение поля *Индекс дела* найденной карточки дела и вывести его в текущую колонку • Если по результатам поиска подходящая карточка дел не будет найдена, то в текущую колонку ничего не выводится

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Заголовок дела	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки дела, у которой в скрытом поле *Ссылка на родительский раздел* указан системный идентификатор текущей карточки раздела • Если по результатам поиска карточка дела будет найдена, то необходимо получить значение поля *Заголовок дела* найденной карточки дела и вывести его в текущую колонку • Если по результатам поиска подходящая карточка дел не будет найдена, то в текущую колонку ничего не выводится Отображается в виде кликабельного дайджеста

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Состояние	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки дела, у которой в скрытом поле Ссылка на родительский раздел указан системный идентификатор текущей карточки раздела • Если по результатам поиска карточка дела будет найдена, то необходимо получить значение системного поля *Состояние* найденной карточки дела и вывести его в текущую колонку • Если по результатам поиска подходящая карточка дел не будет найдена, то в текущую колонку ничего не выводится

Жизненный цикл карточки "Раздел"

Жизненный цикл карточек в процессе работы определяется автоматом состояний:

a. Проект — начальное состояние Раздела
b. Утвержден
c. Закрыт

Изменение состояний выполняется по приведенной схеме.



Рисунок 307. Жизненный цикл карточки "Раздел"

Дайджест раздела

При сохранении карточки раздела ей присваивается дайджест. Дайджест формируется по следующему правилу

```
<индекс раздела> <название подразделения>
```

- <индекс раздела> — значение поля *Индекс раздела* карточки раздела
- <название подразделения> — значение поля *Полное название* карточки подразделения, которое указано в поле *Подразделения* карточки раздела

Кнопки раздела

Для карточки раздела настроены кнопки **Утвердить** и **Закрыть**, позволяющие перевести раздел в другое состояние. В зависимости от текущего состояния раздела и настроек безопасности, кнопки перевода раздела в другое состояние могут быть недоступны пользователю. Так же настроена кнопка **Утвердить приказом**, при нажатии на которую открывается карточка вида указанного в настройках ЭУ Номенклатуры дел.

Закрытие раздела информирует о невозможности списания документов в дела раздела.

Работа с делами номенклатуры дел



Дела не могут быть созданы в *системном* разделе *Разделы*. Дела могут быть созданы в корневых разделах, подразделах, а также в описях в неограниченном количестве.

В окне информации о деле отображаются *Сведения о родительском разделе*, *Сведения о деле* и *Документы, списанные в дело*.

Чтобы раскрыть дело, нажмите на ⊞ слева от его названия. Чтобы свернуть дело, нажмите на ⊟. При нажатии левой кнопкой мыши, дело переходит в фокус, в правой области справочника отображается информация по делу, включая вкладки карточки дела: "Тома" и "Документы, списанные в дело".

В правой области справочника отображается информация по делу, включая вкладки карточки дела: *Тома* и *Документы, списанные в дело*.

Во вкладке *Тома* отображаются все дочерние карточки томов выбранной карточки дела. На вкладке *Документы, списанные в дело* отображаются карточки архивных документов.

Создание нового дела

Дела создаются в состоянии Проект.

Чтобы создать новое дело:

1. Выберите раздел, подраздел или опись для создания дела.
2. Выберите один из доступных способов создания:
   - При помощи элемента управления ⊕.
   - Из соответствующего пункта контекстного меню текущего раздела.
3. Заполните поля карточки дела, затем нажмите **Создать**.

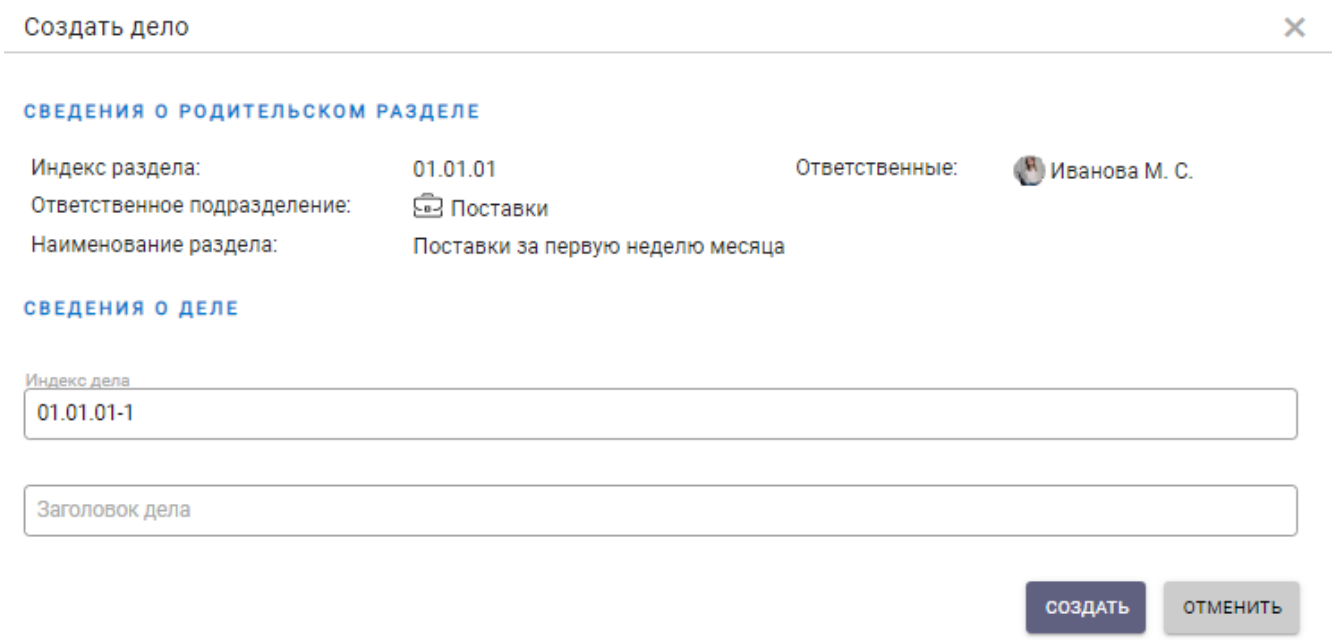


Рисунок 308. Окно создания нового дела

Изменение дела

После создания дела, его можно редактировать при наличии соответствующих прав доступа.

Чтобы изменить дело:

1. Выберите раздел, в котором хранится дело.
2. Установите фокус на необходимом деле, вызовите контекстное меню.
3. Выберите *Редактировать*.
4. Измените поля карточки дела.
5. Сохраните изменения, нажав кнопку **Сохранить**.

Удаление дела

Удаление возможно только при наличии соответствующих прав доступа.

Чтобы удалить дело:

1. Выберите раздел, в котором хранится дело.
2. Перейдите в область дел, вызовите контекстное меню.
3. Выберите *Удалить*.



Дела удаляются со всеми дочерними элементами, включая тома. Если дело связано с несколькими разделами (в случае переходящего дела), оно удалится с удалением последнего раздела, к которому оно относится.

При удалении дела, списанные в него документы, не удаляются. Если в документе была ссылка на дело, такая ссылка будет удалена.

Перечень полей карточки дела

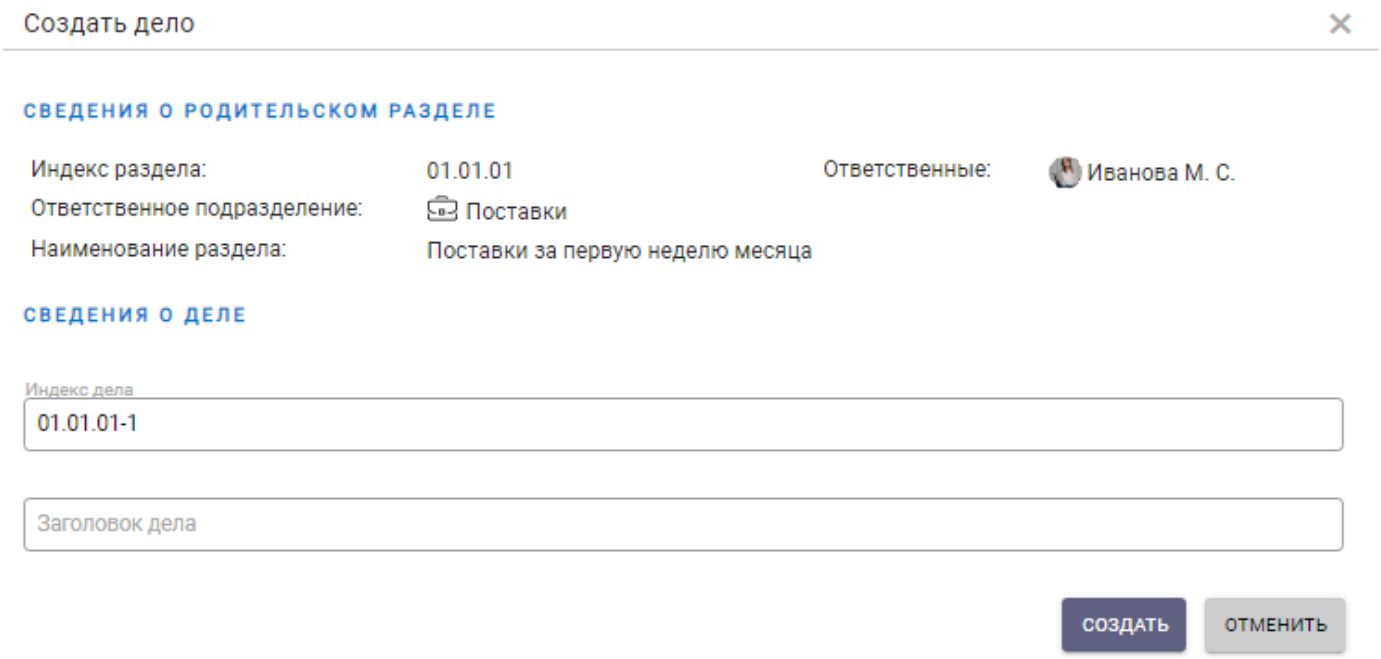


Рисунок 309. Стандартные поля дела номенклатуры дел

Таблица 12. Каждое дело номенклатуры дел содержит:

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Состояние	Состояние	Да	Нет	Текущее состояние карточки дела. При создании состояние `Проект`. .Необходимо для отображения состояния дела номенклатуры дел. * Если состояние `Проект`, номенклатура дел еще не утверждена или не открыта, списывать документы в дела номенклатуры дел нельзя. * Если состояние `Открыто`, номенклатура дел утверждена и открыта. В дела номенклатуры можно списывать документы. * Если состояние `Закрыто`, делопроизводство по документам дела завершено. Списывать документы в дела нельзя. * Если состояние `В архиве`, дела с документами переданы в архив организации и находятся на архивном хранении в архиве организации.
Блок *Основные реквизиты*				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Родительский раздел	Карточка	Да	Нет	Поле в схеме данных. Системный идентификатор родительского раздела, в состав которого входит текущая карточка дела. Необходимо для построения дерева номенклатуры дел в рабочем месте сотрудника, не являющегося архивистом.
Родительская опись	Карточка	Да	Нет	Поле в схеме данных. Системный идентификатор родительской описи, в состав которого входит текущая карточка дела. Необходимо для построения дерева номенклатуры дел в рабочем месте сотрудника, являющегося архивистом.
Заголовок дела	Строка	Да	Да	Заполняется вручную. Необходимо для структурирования и систематизации документов в справочнике номенклатуры дел.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Индекс дела	Строка	Да	Нет	Формируется и заполняется автоматически по следующему правилу: `<Индекс раздела> -- <Порядковый номер>` • `<Индекс раздела>` — значение поля *Индекс раздела* карточки родительского раздела; • `<Порядковый номер>` — порядковый номер в разрезе родительского раздела, ссылка на который указана в скрытом поле *Родительский раздел* карточки дела. В порядковые номера с 1 по 9 добавить ведущий ноль. Необходимо для структурирования и систематизации документов в справочнике номенклатуры дел.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Тип дела	Выпадающий список	Да	Да	Выбор значения из списка: • Электронное • Бумажное • Смешанное Необходимо для учета и систематизации документов в справочнике номенклатуры дел.
Подразделение	Подразделение	Да	Нет	Заполняется автоматически значением из поля *Подразделение*, которое указано в поле *Подразделение* родительской карточки раздела. Необходимо для формирования номенклатуры дел структурного типа, то есть номенклатуры дел на основе структуры организации.
Дата заведения дела	Дата	Да	Нет	Заполняется автоматически текущей датой. Необходимо для фиксации даты заведения карточки дела в справочник номенклатуры дел.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Дата закрытия дела	Дата	Нет	Нет	Заполняется автоматически: • После смены состояния карточки дела на Закрыто БП на изменение состояние дела и тома • После нажатия на кнопку **Закрыть** в карточке дела Необходимо для фиксации даты завершения делопроизводства по документам дела.
Ответственный за принятие в архив	Сотрудник	Нет	Да	Заполняется вручную. Выбор из группы *Архивисты* справочника сотрудников. Необходимо для фиксации ответственного за принятие в архив организации.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Дата начала хранения	Дата	Нет	Нет	Заполняется автоматически значением даты начала года номенклатуры дел (01.01. год номенклатуры дел). Необходимо для фиксации даты, начиная с которой можно списывать документы в дела.
Срок хранения в оперативном архиве	Дата	Да	Да	Заполняется автоматически значением даты окончания года номенклатуры дел, следующего за годом номенклатуры дел (`31.12.год номенклатуры+1`). Необходимо для фиксации срока хранения в оперативном архиве (хранение в структурном подразделении) до передачи дел и документов в архив организации.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Автоматическая конвертация в формат долгосрочного хранения	Да/Нет	Нет	Да	Заполняется вручную. Информационный признак. Необходимо для конвертации электронных документов дел в формат долгосрочного хранения при передаче дел и документов из структурного подразделения организации в архив организации.
Переходящее дела	Да/Нет	Нет	Да	Заполняется вручную. Информационный и технологический признак. Необходимо для фиксации того, что дело является переходящим, то есть по делу не заканчивается делопроизводство в конце текущего делопроизводственного документа. При копировании номенклатуры дел переходящее дело не копируется, если оно не закрыто.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Разрешить списание в закрытое дело	Да/Нет	Нет	Да	Заполняется вручную. Информационный признак. Необходимо для фиксации того, что списание документов допустимо даже в закрытое дело, то есть в дело, по которому завершено делопроизводство.
Примечания	Текст	Нет	Да	Заполняется вручную. Необходимо для фиксации дополнительной необходимой информации по делу.
Место хранения	Строка справочника	Нет	Да	Заполняется вручную. Выбор из справочника *Места хранения*. Необходимо для фиксации места хранения дела в структурном подразделении организации или в архиве организации.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Отключить формирование томов	Да/Нет	Нет	Да	Заполняется вручную. Информационный и технологический признак. Необходимо для фиксации того, что по делу не должны формироваться тома дела при превышении допустимого количества листов в деле. При списании документа в дело через поля *Дело* превышение количества листов в деле для формирования тома не учитывается.
Выдан	Да/Нет	Нет	Да	Заполняется вручную. Информационный признак. Необходимо для фиксации того, что дело выдано из архива организации.
Утерян	Да/Нет	Нет	Да	Заполняется вручную. Информационный признак. Необходимо для фиксации того, что дело утеряно в архиве организации.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Выделен к уничтожению	Да/Нет	Нет	Да	Заполняется вручную. Информационный признак. Необходимо для фиксации того, что по делу истек срок хранения, то есть дело выделено к уничтожению.
Уничтожен	Да/Нет	Нет	Да	Заполняется вручную. Информационный признак. Необходимо для фиксации того, что дело уничтожено в структурном подразделении организации или архиве организации.
Блок *Хранение*				
Кабинет/Помещение	Строка	Нет	Да	Заполняется вручную. Необходимо для фиксации информации о физическом хранении дела в структурном подразделении организации или архиве организации.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
№ шкафа	Строка	Нет	Да	Заполняется вручную. Необходимо для фиксации информации о физическом хранении дела в структурном подразделении организации или архиве организации.
№ полки	Строка	Нет	Да	Заполняется вручную. Необходимо для фиксации информации о физическом хранении дела в структурном подразделении организации или архиве организации.
Блок *Журнал передач*				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Выдана ед. хранения	Карточка	Нет	Да	Заполняется вручную. Для получения карточек документов обязательно необходимо указать хотя бы один из параметров запроса. В качестве параметров запроса используются стандартные параметры: "Автор", "Дата изменения", "Дата создания", "Дайджест". Необходимо для фиксации того, какой документ дела был выдан из архива организации. Если из архива выдается дело, то поле не заполняется.
Выдал	Сотрудник	Нет	Да	Заполняется вручную. Необходимо для фиксации того, кому выданы документ или дело из архива организации.
Кому выдано	Сотрудник	Нет	Да	Заполняется вручную. Необходимо для фиксации того, кому выданы документ или дело из архива организации.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Дата выдачи	Дата	Нет	Да	Заполняется вручную. Необходимо для фиксации даты выдачи документа или дела из архива организации.
Дата возврата (план)	Дата	Нет	Да	Заполняется вручную. Необходимо для фиксации плановой даты возврата документа или дела в архив организации.
Принял	Сотрудник	Нет	Да	Заполняется вручную. Необходимо для фиксации того, какой сотрудник принял документ или дело в архив организации.
Дата возврата (факт)	Дата	Нет	Да	Заполняется вручную. Необходимо для фиксации фактической даты возврата документа или дела в архив организации.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Комментарии	Текст	Нет	Да	Заполняется вручную. Необходимо для фиксации дополнительного комментария по выдаче документа или дела из архива организации.
Блок *Уничтожение*				
Опись №	Строка	Нет	Да	Заполняется вручную. Необходимо для фиксации номера сводной описи, в состав которой входит дело.
Протокол ЭПК №	Строка	Нет	Да	Заполняется вручную. Необходимо для фиксации номера протокола заседания экспертно-проверочной комиссии при согласовании дел, выделенных к уничтожению, то есть дел, у которых истек срок хранения.
Фонд	Строка	Нет	Да	Заполняется вручную. Необходимо для указания архивного фонда, к которому относится дело.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Срок хранения	Строка справочника	Да	Да	Заполняется вручную. Выбор из справочника *Сроки хранения*. Необходимо для фиксации срока хранения документов дела в структурном подразделении организации или в архиве организации.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Дата окончания хранения	Дата	Нет	Да	Заполняется автоматически после заполнения поля *Срок хранения*. • Если в поле *Срок хранения* выбрана строка справочника *Сроки хранения*, у которой заполнено поле *Срок (лет)*, то дата окончания хранения рассчитывается как 1 января года, следующего за годом, к которому относится родительский для дела раздел + Срок (лет) из соответствующей карточки строки конструктора справочника Сроки хранения. • Если в поле *Срок хранения* выбрана строка справочника *Сроки хранения*, у которой заполнено поле *Срок (список)*, то дата окончания хранения не рассчитывается и не записывается. • При очищении поля *Срок хранения*

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Фактическая дата уничтожения	Дата	Нет	Да	Заполняется вручную. Необходимо для фиксации даты фактического уничтожения документов дела в структурном подразделении организации или архиве организации.
Заявка на уничтожение	Карточка	Нет	Да	Заполняется вручную. Для получения карточек документов обязательно необходимо указать хотя бы один из параметров запроса. В качестве параметров запроса используются стандартные параметры: *Автор*, *Дата изменения*, *Дата создания*, *Дайджест*. Необходимо для фиксации архивной учетной формы, в рамках которой уничтожены дела и документы.
Вкладка *Тома*				
Таблица *Тома* (Контрол табличных данных)				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
№	—	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки тома, у которой в скрытом поле *Родительское дело* указан системный идентификатор текущей карточки дела. • Если по результатам поиска карточка тома будет найдена, то необходимо получить значение поля *№* найденной карточки тома и вывести его в текущую колонку. • Если по результатам поиска подходящая карточка тома не будет найдена, то в текущую колонку ничего не выводится.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Наименование	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки тома, у которой в скрытом поле *Родительское дело* указан системный идентификатор текущей карточки тома. • Если по результатам поиска карточка тома будет найдена, то необходимо получить значение поля *Наименование* найденной карточки тома и вывести его в текущую колонку. • Если по результатам поиска подходящая карточка тома не будет найдена, в текущую колонку ничего не выводится. Отображается в виде кликабельного дайджеста

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Количество листов	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки тома, у которой в скрытом поле *Родительское дело* указан системный идентификатор текущей карточки тома. • Если по результатам поиска карточка тома будет найдена, то необходимо получить значение поля *Количество листов* найденной карточки тома и вывести его в текущую колонку. • Если по результатам поиска подходящая карточка тома не будет найдена, то в текущую колонку ничего не выводится.
Вкладка *Документы, списанные в дело*				
Таблица *Документы, списанные в дело* (Контрол табличных данных)				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Том	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки тома, у которой в скрытом поле *Родительское дело* указан системный идентификатор текущей карточки дела. • Если по результатам поиска карточка тома будет найдена, то необходимо получить значение поля *№* найденной карточки тома и вывести его в текущую колонку. • Если по результатам поиска подходящая карточка тома не будет найдена, то в текущую колонку ничего не выводится. • Ищем карточки вида *Том*, у которых в поле *Ссылка на родительское дело* ссылка на текущую карточку дела.

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Наименование документа	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки архивного документа, у которой в поле *Дело* указан системный идентификатор текущей карточки дела. • Если по результатам поиска карточка архивного документа будет найдена, то необходимо получить значение поля *Тема* найденной карточки архивного документа и вывести его в текущую колонку. • Если по результатам поиска подходящая карточка архивного документа не будет найдена, то в текущую колонку ничего не выводится. Отображается в виде кликабельного дайджеста

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Кол-во листов	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки архивного документа, у которой в поле *Дело* указан системный идентификатор текущей карточки дела. • Если по результатам поиска карточка архивного документа будет найдена, то необходимо получить значения полей из схемы данных *Кол-во листов* в карточке документа и значение поля *Кол-во листов в приложении* в карточке архивного документа (Стандартные поля в схеме данных карточки документа), просуммировать их и вывести его в текущую колонку. • Если по результатам поиска подходящая

Жизненный цикл карточки вида "Архивное дело"

Жизненный цикл карточек в процессе работы определяется автоматом состояний:

1. Проект — начальное состояние карточки *Архивное дело*
2. Открыто
3. Закрыто
4. В архиве

Изменение состояний выполняется по приведенной схеме.



Рисунок 310. Автомат состояний карточки "Архивное дело"

Дайджест дела

При сохранении карточки дела ей присваивается дайджест. Дайджест

формируется по следующему правилу

```
<индекс дела> <заголовок дела>
```

- <индекс дела> - значение поля *Индекс дела* карточки дела
- <заголовок дела> - значение поля *Заголовок дела* карточки дела

Кнопки дела

Для карточки дела настроены кнопки **Открыть**, **Закрыть** и **В архиве**, позволяющие перевести дело в другое состояние. В зависимости от текущего состояния дела и настроек безопасности, кнопки перевода дела в другое состояние могут быть недоступны пользователю.

Закрытие дела информирует о невозможности списания в него документов, если не указано соответствующее разрешение.

Работа с томами номенклатуры дел

Тома могут быть созданы только в делах. Тома могут быть созданы в неограниченных количествах.

Создание доступно в рабочем месте сотрудника, не являющегося архивистом.

Посмотреть информацию о томе можно установив фокус на нем: в правой части экрана отобразится информация, указанная в выбранном томе, включая вкладку *Документы, списанные в дело*. На вкладке *Документы, списанные в дело*, отображаются карточки архивных документов.

Создать новый том номенклатуры дел

Тома создаются в состоянии Проект.

Чтобы создать новый том:

1. Выберите дело для создания тома;
2. Выберите один из доступных способов для создания:
3. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего фонда.
4. Заполните поля карточки тома, затем нажмите **Создать**.

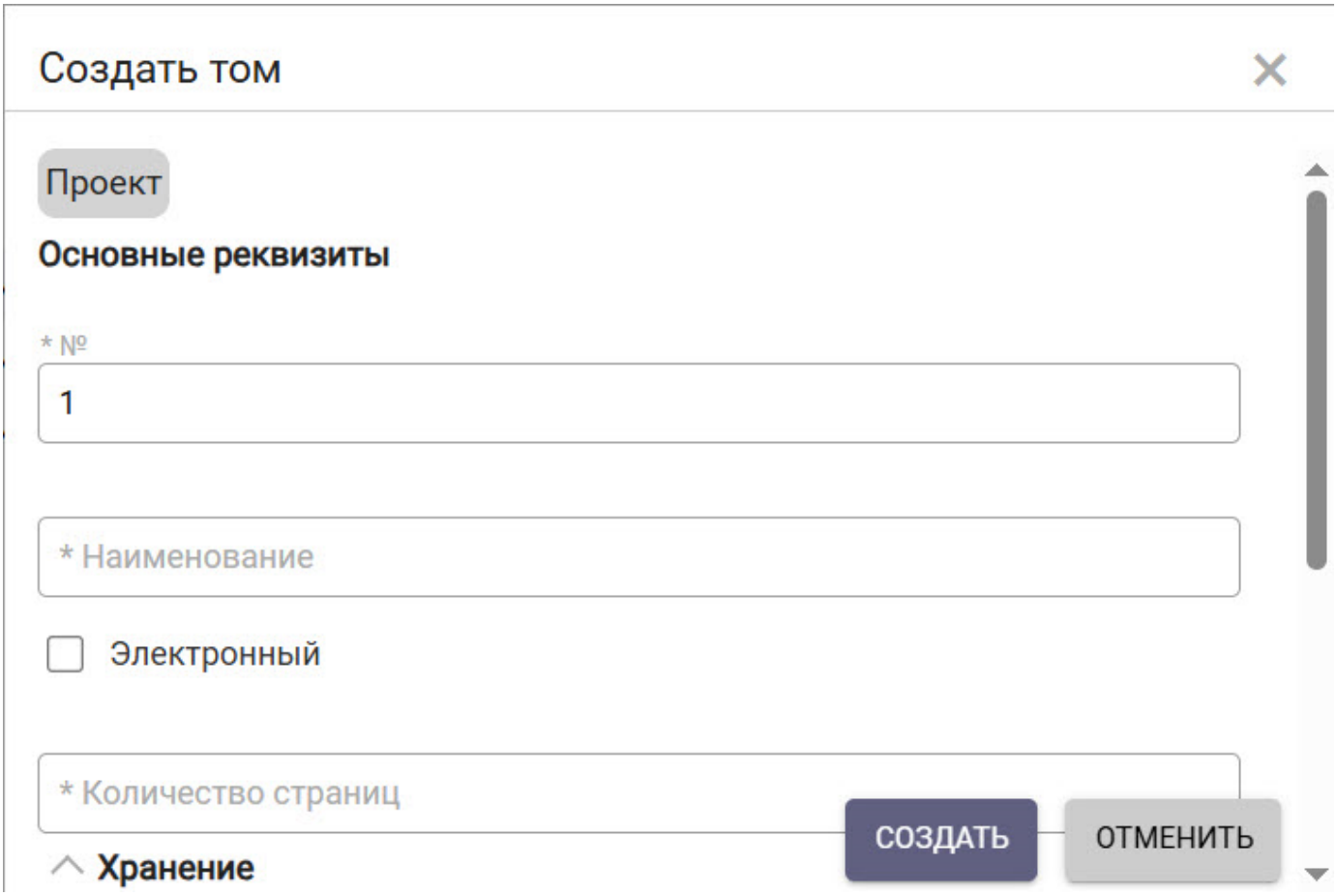


Рисунок 311. Окно создания нового тома

Изменение тома

После создания тома, его можно редактировать при наличии соответствующих прав доступа.

Чтобы изменить том:

1. Выберите раздел, в котором хранится дело
2. Выберите дело, в котором хранится том
3. Выберите том, вызовите контекстное меню
4. Выберите **Редактировать**
5. Измените поля карточки тома

Удаление тома

Удаление возможно только при наличии соответствующих прав доступа.

Чтобы удалить том:

1. Выберите раздел, в котором хранится дело

2. Выберите дело, в котором хранится том
3. Выберите том, вызовите контекстное меню
4. Выберите **Удалить**

При удалении тома, списанные в него документы, не удаляются. Если в документе была ссылка на том, такая ссылка будет удалена.

Перечень полей карточки тома

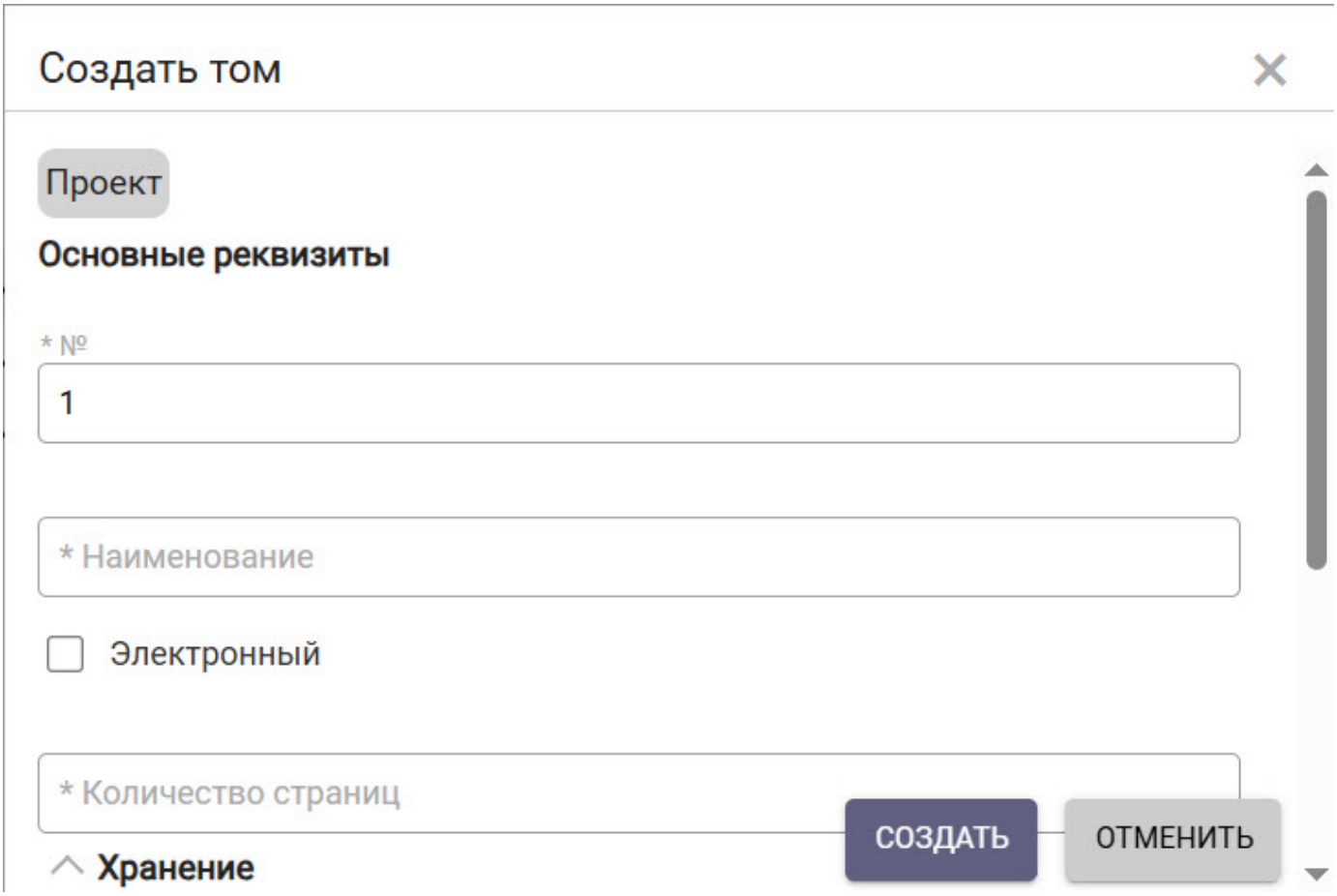


Рисунок 312. Стандартные поля тома номенклатуры дел

Таблица 13. Каждый том номенклатуры дел содержит:

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Состояние	Состояние	Да	Нет	Текущее состояние карточки тома. При создании состояние `Проект`. .Необходимо для отображения состояния тома дела номенклатуры дел: * Если состояние `Проект`, то номенклатура дел еще не утверждена или не открыта, списывать документы в том дела номенклатуры дел нельзя. * Если состояние `Открыто`, то номенклатура дел утверждена и открыта. В том дела номенклатуры можно списывать документы. * Если состояние `Закрыто`, то делопроизводство по документам тома дела завершено. Списывать документы в том дела нельзя. * Если состояние `В архиве`, то том дела с документами переданы в архив организации и находятся на архивном хранении в архиве организации.
Блок *Основные реквизиты*				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Ссылка на родительское дело	Карточка	Да	Нет	Поле в схеме данных. Системный идентификатор родительского дела, в состав которого входит текущая карточка тома. Необходимо для построения дерева номенклатуры дел в рабочем месте сотрудника, не являющегося архивистом.
№	Число	Да	Нет	Заполняется автоматически порядковым номером в разрезе родительского дела, начиная с 1. Необходимо для структурирования и систематизации документов в справочнике номенклатуры дел.
Наименование	Строка	Да	Да	Заполняется вручную. Необходимо для структурирования и систематизации документов в справочнике номенклатуры дел.
Количество страниц	Число	Да	Да	Заполняется вручную.
Электронный	Да/Нет	Нет	Да	Заполняется вручную.

Название поля	**Тип поля**	**Обязательность заполнения**	**Доступность редактирования**	**Комментарий**
Блок *Хранение*				
Кабинет/Помещение	Строка	Нет	Да	Заполняется вручную
№ шкафа	Строка	Нет	Да	Заполняется вручную
№ полки	Строка	Нет	Да	Заполняется вручную
Вкладка *Документы, списанные в дело*				
Таблица *Документы, списанные в дело* (Контрол табличных данных)				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Наименование документа	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки архивного документа, у которой в поле *Дело* указан системный идентификатор на текущую карточку тома • Если по результатам поиска карточка архивного документа будет найдена, то необходимо получить значение поля *Тема* найденной карточки архивного документа и вывести его в текущую колонку • Если по результатам поиска подходящая карточка архивного документа не будет найдена, то в текущую колонку ничего не выводится Отображается в виде кликабельного дайджеста

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Кол-во листов	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки архивного документа, у которой в поле *Дело* указан системный идентификатор текущей карточки тома • Если по результатам поиска карточка архивного документа будет найдена, то необходимо получить значения полей из схемы данных *Кол-во листов* в карточке документа и значение поля *Кол-во листов в приложении* в карточке архивного документа (Стандартные поля в схеме данных карточки документа), просуммировать их и вывести его в текущую колонку • Если по результатам поиска подходящая

Жизненный цикл карточки вида "Том"

Жизненный цикл карточек в процессе работы определяется автоматом состояний:

1. Проект — начальное состояние карточки *Том*
2. Открыто
3. Закрыто
4. В архиве

Изменение состояний выполняется по приведенной схеме.



Рисунок 313. Автомат состояний карточки "Том"

Дайджест тома

При сохранении карточки тома ей присваивается дайджест. Дайджест

формируется по следующему правилу

```
<№> <заголовок тома>
```

- <№> — значение поля *№* карточки тома
- <заголовок тома> — значение поля *Заголовок тома* карточки тома

Кнопки тома

Для карточки тома настроены кнопки **Открыть**, **Закрыть** и **В архиве**, позволяющие перевести дело в другое состояние. В зависимости от текущего состояния тома и настроек безопасности, кнопки перевода дела в другое состояние могут быть недоступны пользователю.

Закрытие тома информирует о невозможности списания в него документов.

Работа с архивными документами

Создать архивный документ

Создание архивных документов вручную не предусмотрено.

Архивный документ должен являться частью процесса передачи-приема документального фонда организации с временными сроками хранения свыше 10 лет или с постоянными сроками хранения в архив организации. В карточку архивного документа должны конвертировать карточки документов, которые были приняты в архив организации.

Процесс передачи-приема документального фонда в архив организации настраивается самостоятельно, либо установкой модуля *Долговременный архив*.

Изменение архивного документа

Изменение карточки архивного документа не предусмотрено.

Удаление архивного документа

Удаление архивного документа возможно при наличии соответствующих прав с помощью стандартной функции удаления карточек.

Перечень полей карточки архивного документа

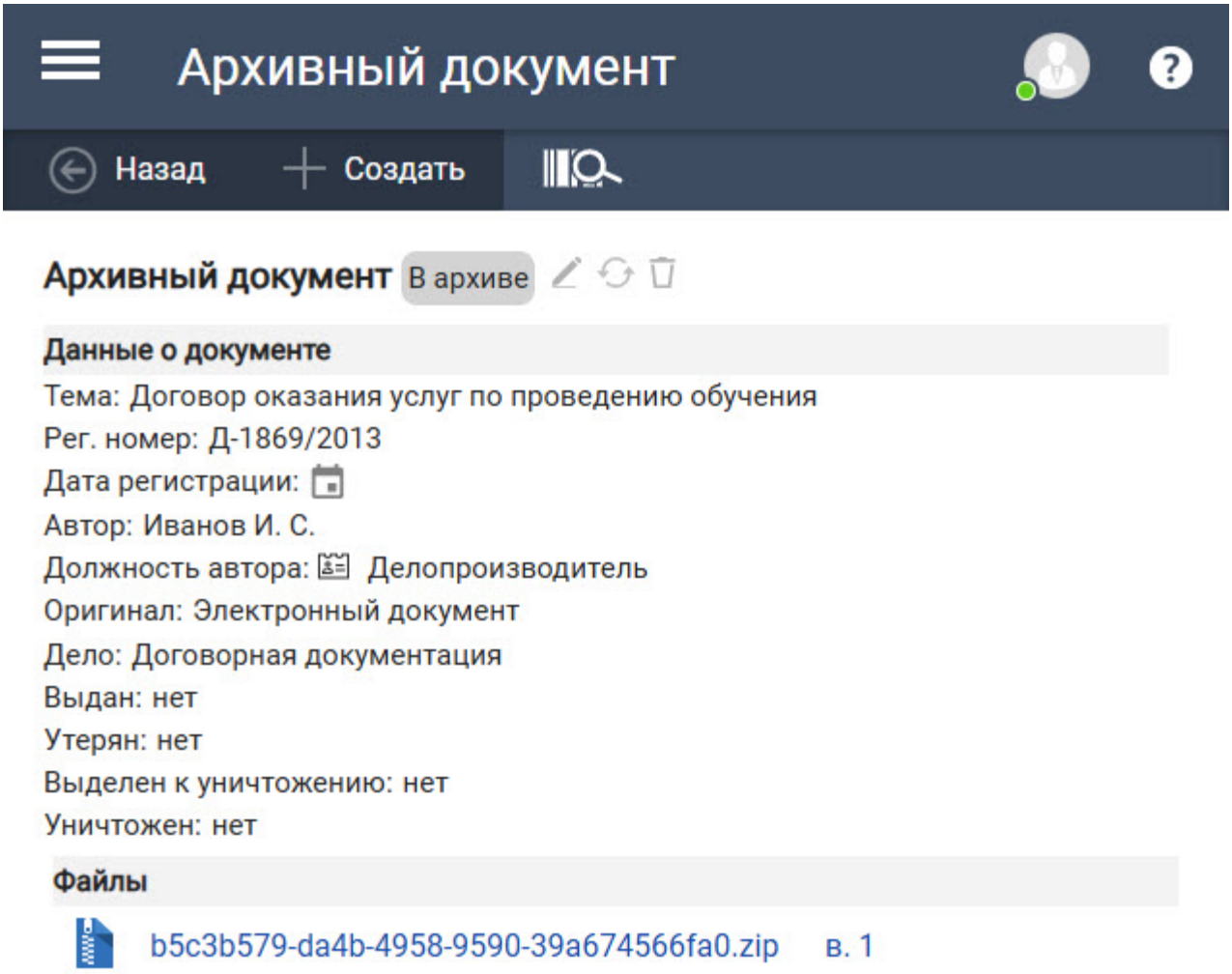


Рисунок 314. Информация об архивном документе

Таблица 14. Каждый архивный документ содержит.

Название поля	**Тип поля**	**Обязательность заполнения**	**Доступность редактирования**	**Комментарий**
Состояние	Состояние	Да	Нет	Текущее состояние карточки архивного документа. При создании состояние `В архиве`
Блок *Данные о документе*				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Тема	Строка	Да	Нет	Заполняется автоматически в процессе передачи-приема в архив организации
Рег. номер	Строка	Да	Нет	
Дата регистрации	Строка	Да	Нет	
Автор	Строка	Да	Нет	
Должность автора	Записи справочника сотрудников	Да	Нет	
Выдан	Да/Нет	Нет	Нет	
Утерян	Да/Нет	Нет	Нет	
Выделен к уничтожению	Да/Нет	Нет	Нет	
Дело	Дело	Да	Нет	
Блок *Файлы*				
Файлы	Файл	Да	Нет	Заполняется автоматически в процессе передачи-приема в архив организации

Жизненный цикл карточки вида "Архивный документ"

Жизненный цикл карточек в процессе работы определяется автоматом состояний:

1. В архиве — начальное состояние карточки *Архивный документ*

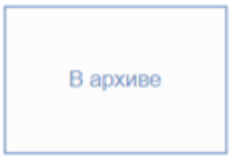


Рисунок 315. Автомат состояний карточки "Архивный документ"

Дайджест архивного документа

При сохранении карточки архивного документа ей присваивается дайджест. Дайджест формируется по следующему правилу

```
<Тема><Рег.номер><Автор>
```

- <Тема> — значение поля *Тема*
- <Рег.номер> — значение поля *Рег.номер*
- <Автор> — значение поля *Автор*

Кнопки архивного документа

Для карточки архивного документа кнопок не предусмотрено.

Утверждение номенклатуры дел

В рабочем месте сотрудника, не являющегося архивистом, доступны кнопки для утверждения номенклатуры дел:

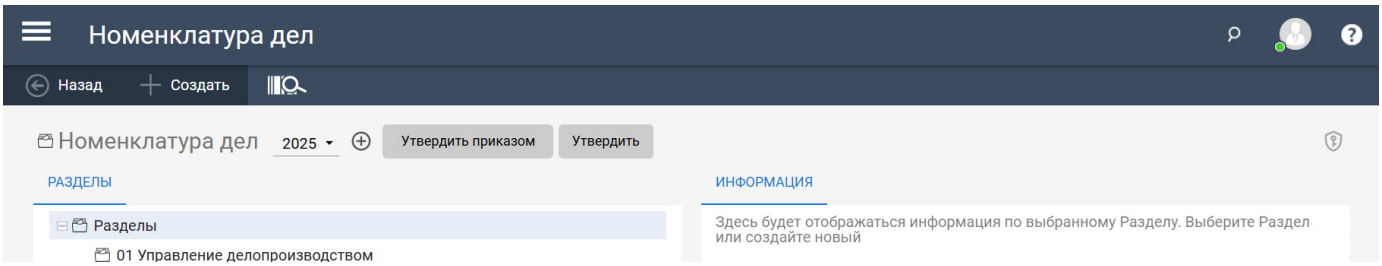


Рисунок 316. Кнопки утверждения номенклатуры дел

- Кнопка **Утвердить**.

  После нажатия кнопки:

  - Разделы номенклатуры дел выбранного года, из которой была нажата кнопка, переводятся в состояние `Утвержден`
  - Карточки дел, которые входят в состав этих разделов, переводятся в состояние `открыто`
  - Карточки томов, которые входят в состав дел, переводятся в состояние `открыто`

- Кнопка **Утвердить приказом**.

  После нажатия кнопки:

  - Открывается карточка выбранного вида из модуля управления документами (по умолчанию карточка вида ОРД).
  - В схему данных карточки передается признак `ArchiveManagment=True`
  - В схему данных карточки также передается год, в рамках которой был утвержден приказ

Работа с фондами номенклатуры дел

Для рабочего места сотрудника, являющегося архивистом, справочник представляет собой иерархическую структуру. Верхним разделом справочника является системный раздел *Фонд*, являющийся родителем для всех остальных.

Системный раздел может содержать неограниченное количество корневых фондов. Каждый корневой фонд может иметь неограниченное число подфондов первого уровня.

В подфондах первого уровня может быть создано произвольное количество подфондов следующих уровней. Глубина вложенности не ограниченная.

Чтобы раскрыть фонд, нажмите на ⊞ слева от его названия. Чтобы свернуть фонд, нажмите на ⊟. При нажатии левой кнопкой мыши, фонд переходит в фокус, в правой области справочника отображается информация по фонду.

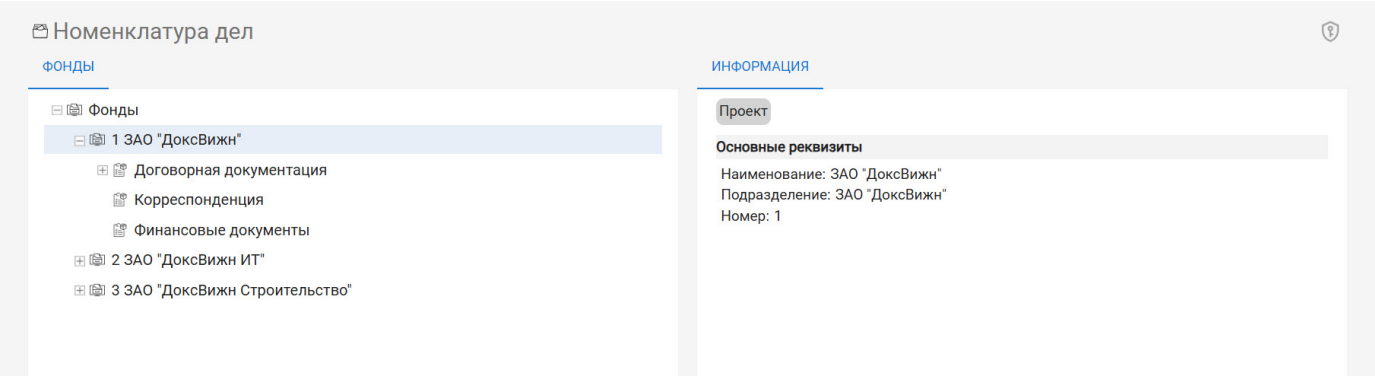


Рисунок 317. Вкладка "Информация"

Создать новый фонд номенклатуры дел

Фонды создаются в состоянии `Проект`.

Чтобы создать фонд:

1. Установите фокус на системном фонде или на фонде, к которому необходимо создать подфонд

2. Выберите один из доступных способов создания:
3. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего фонда.
4. Заполните поля нового фонда.
5. Создайте желаемое количество фондов.
6. Создавайте подфонды, повторяя шаги 1-3. Создать подфонд можно при помощи элемента управления или из пункта Создать фонд контекстного меню.
7. Создавайте дочерние фонды до получения желаемой иерархической структуры.

Изменение фонда номенклатуры дел

После создания фонд справочника может быть изменен при наличии соответствующих прав доступа.

Чтобы отредактировать фонд:

1. Выберите фонд номенклатуры дел, который нужно изменить.
2. Измените фонд одним из доступных способов:
   - Выберите в контекстном меню фонда пункт *Редактировать*.
   - При помощи элемента управления в правой области справочника.
3. Измените поля фонда.
4. Сохраните изменения, нажав кнопку **Сохранить**.

Удаление фонда номенклатуры дел

Удаление возможно только при наличии соответствующих прав доступа.



Фонды удаляются со всеми дочерними элементами, включая описи, дела и тома. При удалении дела или тома, списанные в него архивные документы, не удаляются. Если в архивном документе была ссылка на дело или том, такая ссылка будет удалена.

Чтобы удалить фонд справочника:

1. Выберите фонд номенклатуры, который нужно удалить.
2. Выберите в контекстном меню раздела пункт Удалить.

3. Появится предупреждение, подтвердите действие кнопкой **ОК**.

Перечень полей фонда номенклатуры дел

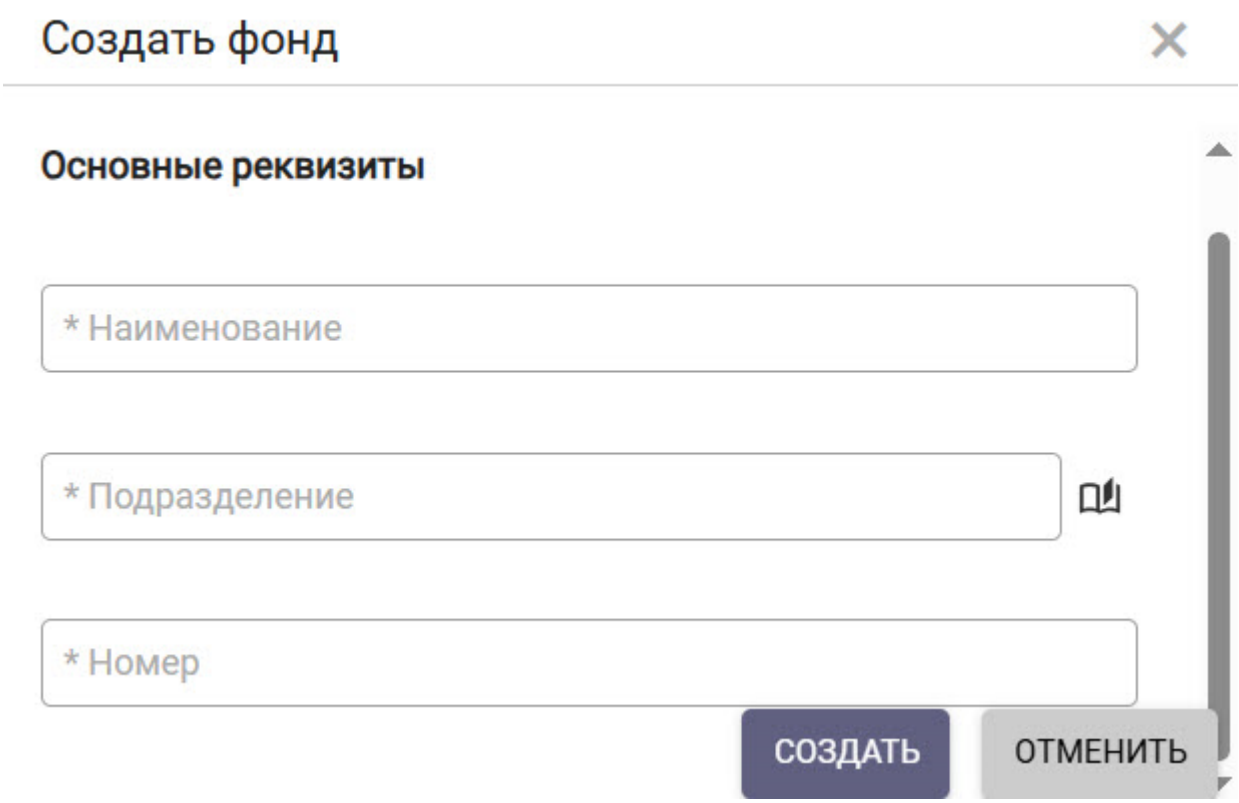


Рисунок 318. Стандартные поля фонда номенклатуры дел

Таблица 15. Каждый фонд содержит поля для заполнения.

Название поля	**Тип поля**	**Обязательность заполнения**	**Доступность редактирования**	**Комментарий**
Состояние	Состояние	Да	Нет	Текущее состояние карточки фонда. При создании состояние Проект
Блок *Основные реквизиты*				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Родительский фонд	Карточка	Да	Нет	Поле в схеме данных. Системный идентификатор системного фонда или родительского фонда, в состав которого входит текущая карточка фонда
Наименование	Строка	Да	Да	Заполняется вручную
Подразделение	Подразделение	Да	Да	Заполняется вручную
Номер	Строка	Да	Да	Заполняется вручную

Жизненный цикл карточки вида "Фонд"

Жизненный цикл карточек в процессе работы определяется автоматом состояний:

1. Проект — начальное состояние карточки *Фонд*

   Изменение состояний выполняется по приведенной схеме.



Рисунок 319. Автомат состояний карточки "Фонд"

Дайджест фонда

При сохранении карточки фонда ей присваивается дайджест. Дайджест формируется по следующему правилу

```
<номер> <наименование>
```

- <номер> - значение поля *Номер* карточки фонда
- <наименование> - значение поля *Наименование* карточки фонда

Кнопки фонда

Для карточки фонда кнопок не предусмотрено.

Работа с описями номенклатуры дел

Для рабочего места сотрудника, являющегося архивистом, возможно создание описей.

Описи не могут быть созданы в системном фонде Фонд. Описи могут быть созданы в корневых фондах и подфондах. Описи могут быть созданы в корневых фондах и подфондах в неограниченном количестве.

Создать новую опись номенклатуры дел

Описи создаются в состоянии Проект.

Чтобы создать опись:

1. Установите фокус на корневом фонде или на подфонде, к которому необходимо создать опись.
2. Выберите один из доступных способов создания:
3. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего фонда.
4. Заполните поля новой описи, затем нажмите кнопку **Создать**.

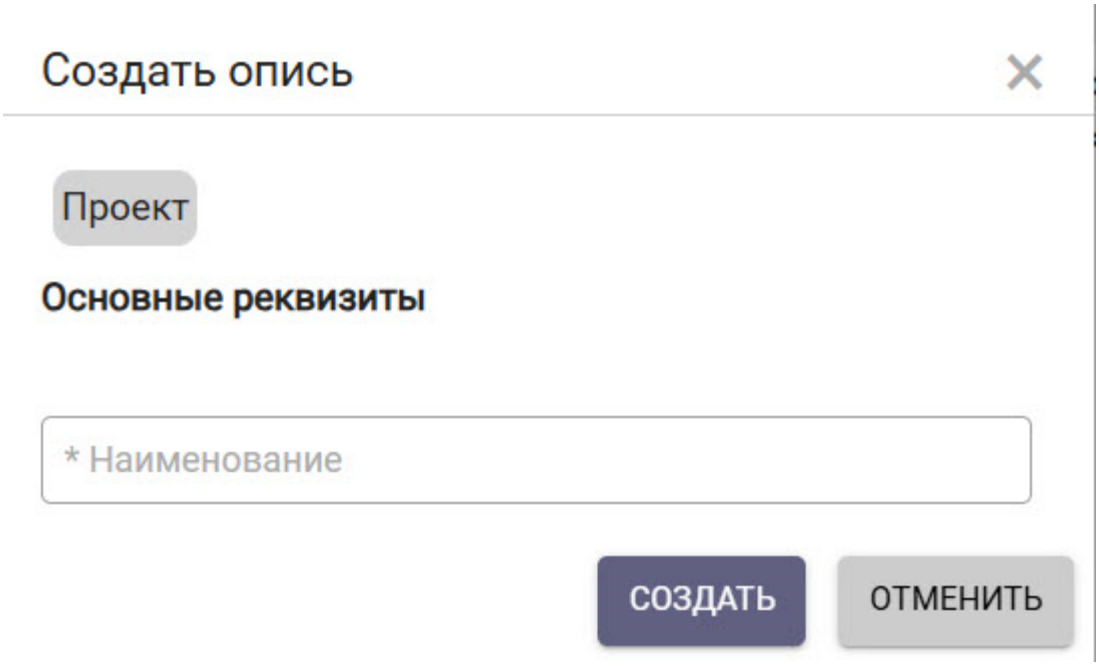


Рисунок 320. Окно создания новой описи

Изменение описи номенклатуры дел

После создания опись справочника может быть изменена при наличии соответствующих прав доступа.

Чтобы отредактировать опись:

1. Выберите опись номенклатуры дел, которую нужно изменить.
2. Измените опись одним из доступных способов:
   - Выберите в контекстном меню фонда пункт *Редактировать*.
   - При помощи элемента управления в правой области справочника.
3. Измените поля описи.
4. Сохраните изменения, нажав кнопку **Сохранить**.

Удаление описи номенклатуры дел

Удаление возможно только при наличии соответствующих прав доступа.

> Описи удаляются со всеми дочерними элементами, включая дела и тома. При удалении дела или тома, списанные в него архивные документы, не удаляются. Если в архивном документе была ссылка на дело или том, такая ссылка будет удалена.

Чтобы удалить опись справочника:

1. Выберите опись номенклатуры, которую нужно удалить.
2. Выберите в контекстном меню раздела пункт **Удалить**.
3. Появится предупреждение, подтвердите действие кнопкой **ОК**.

Перечень полей описи номенклатуры дел

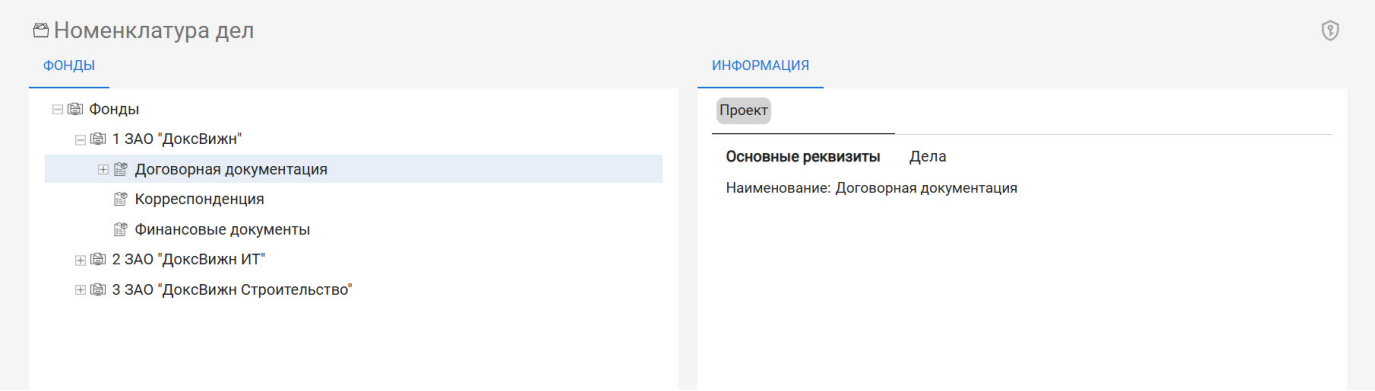


Рисунок 321. Стандартные поля описи номенклатуры дел

Таблица 16. Каждая опись содержит следующие поля для заполнения:

Название поля	**Тип поля**	**Обязательность заполнения**	**Доступность редактирования**	**Комментарий**
Состояние	Состояние	Да	Нет	Текущее состояние карточки описи. При создании состояние Проект
Блок *Основные реквизиты*				
Родительский фонд	Карточка	Да	Нет	Поле в схеме данных. GUID корневого фонда, в состав которого входит текущая карточка описи
Наименование	Строка	Да	Да	Заполняется вручную
Вкладка *Дела*				
Таблица *Дела* (Контрол табличных данных)				

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Индекс дела	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки дела, у которой в скрытом поле *Родительская опись* указан системный идентификатор текущей карточки описи • Если по результатам поиска карточка дела будет найдена, то необходимо получить значение поля *Индекс дела* найденной карточки дела и вывести его в текущую колонку • Если по результатам поиска подходящая карточка дел не будет найдена, то в текущую колонку ничего не выводится

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Заголовок дела	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки дела, у которой в скрытом поле *Родительская опись* указан системный идентификатор текущей карточки описи • Если по результатам поиска карточка дела будет найдена, то необходимо получить значение поля *Заголовок дела* найденной карточки дела и вывести его в текущую колонку • Если по результатам поиска подходящая карточка дел не будет найдена, то в текущую колонку ничего не выводится Отображается в виде кликабельного дайджеста

Название поля	Тип поля	Обязательность заполнения	Доступность редактирования	Комментарий
Состояние	-	-	-	Значение колонки заполняется в соответствии со следующим правилом: • Осуществляется поиск карточки дела, у которой в скрытом поле *Родительская опись* указан системный идентификатор текущей карточки описи • Если по результатам поиска карточка дела будет найдена, то необходимо получить значение системного поля *Состояние* найденной карточки дела и вывести его в текущую колонку • Если по результатам поиска подходящая карточка дел не будет найдена, то в текущую колонку ничего не выводится

Жизненный цикл карточки вида "Опись"

Жизненный цикл карточек в процессе работы определяется автоматом состояний:

1. Проект — начальное состояние карточки *Опись*

   Изменение состояний выполняется по приведенной схеме.



Рисунок 322. Автомат состояний карточки "Опись"

Дайджест описи

При сохранении карточки описи ей присваивается дайджест. Дайджест формируется по следующему правилу

```
<наименование>
```

- `<наименование>` — значение поля *Наименование* карточки описи

Кнопки описи

Для карточки описи кнопок не предусмотрено.

Настройки безопасности справочника номенклатуры дел



Настройки дискреционной безопасности наследуются дочерними элементами. Это означает, что, если наследование не отключено, настройки дискреционной безопасности для дел будут унаследованы от родительских разделов, настройки разделов от родительских годов и т.д.

Администратор может задать права доступа для всего справочника при помощи элемента управления , расположенного над правой областью.

Чтобы задать права доступа для всего справочника:

1. Перейдите в справочник по ссылке из основного меню.
2. Нажмите кнопку .
3. В появившемся окне задайте параметры доступа для пользователей согласно инструкции, данной в пункте Настройка безопасности объектов.
4. Подтвердите настройки, нажав **Применить** или **ОК**.

По умолчанию доступ к первому уровню справочника предоставляется только двум категориям пользователей: **Архивисты** и **Системная для WF**. В случае необходимости изменения логики доступа, настройка осуществляется через ЭУ описанный выше.

Настройки ролевой безопасности

В справочнике номенклатуры дел реализована ролевая безопасность. Настройки ролевой безопасности регулируют доступность операций для следующих карточек:

1. Раздел
2. Архивное дело
3. Том
4. Фонд
5. Опись
6. Архивный документ

При необходимости сотруднику могут быть назначены дополнительные роли. При назначении сотрудника на роль ему становится доступным ряд операций по тому или иному виду карточки

Назначение ролевой безопасности подробнее описывается в Руководстве администратора модуля *Управление архивом*, пункт "Настройка ролевой модели справочника номенклатуры дел".

Поиск по номенклатуре дел

В справочнике реализована функция поиска. Поиск активируется нажатием на в верхней части пользовательского интерфейса Web-клиента. Результата поиска можно показать или скрыть нажатием на кнопку . Если список найденного слишком большой, можно развернуть и свернуть результаты поиска кнопками .

Поиск осуществляется:

- В режиме отображения "Делопроизводитель" поиск осуществляется по всем разделам и делам справочника в выбранном году. Если был выбран другой год, то поиск запустится снова, но уже для другого года. В раскрывающемся списке слева от поисковой строки можно выбрать поиск только по делам или

только по разделам.

- В режиме отображения "Архивариус" поиск осуществляется по всем фондам, описям и делам справочника.

Поиск осуществляется по полям "Объектов номенклатуры": MainInfo\Name, MainInfo\Title, MainInfo\Index, MainInfo\Number.

В результатах поиска отображаются все найденные результаты, а также указывается общее количество найденных результатов. Цветом выделяются совпадения с поисковым запросом.

Нажатие на результат поиска переводит в иерархическую структуру справочника. Структура справочника сворачивается до выбранного элемента и фокус смещается на найденный элемент.

Копировать элементы справочника номенклатуры дел

В справочнике реализована возможность *Копировать* и *Вырезать* разделы справочника. Операции копирования или вырезания могут быть вызваны из контекстного меню выбранного раздела. Вокруг скопированного или вырезанного раздела появляется пунктирная рамка. Вырезанный раздел дополнительно обозначается серым цветом. Цвет скопированного раздела не меняется.

Операция Копировать

Копирует:

- Иерархию разделов, дел и том, без списанных документов, в рабочем месте сотрудника, не являющегося архивистом;
- Иерархию фондов и описей, без включенных документов, в рабочем месте сотрудника, являющегося архивистом.

При копировании дерева номенклатуры дел рядом с копируемым разделом отображается статус "В процессе копирования". Данный статус необходим для информационного отображения того, что происходит копирование дел и томов раздела.

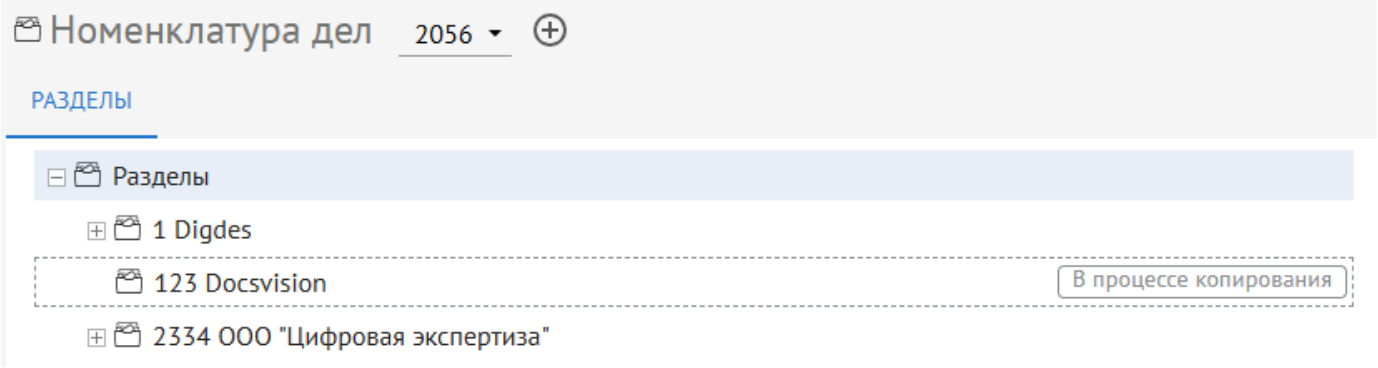


Рисунок 323. Статус "В процессе копирования"

После окончания копирования дел и томов раздела статус "В процессе копирования" не отображается.

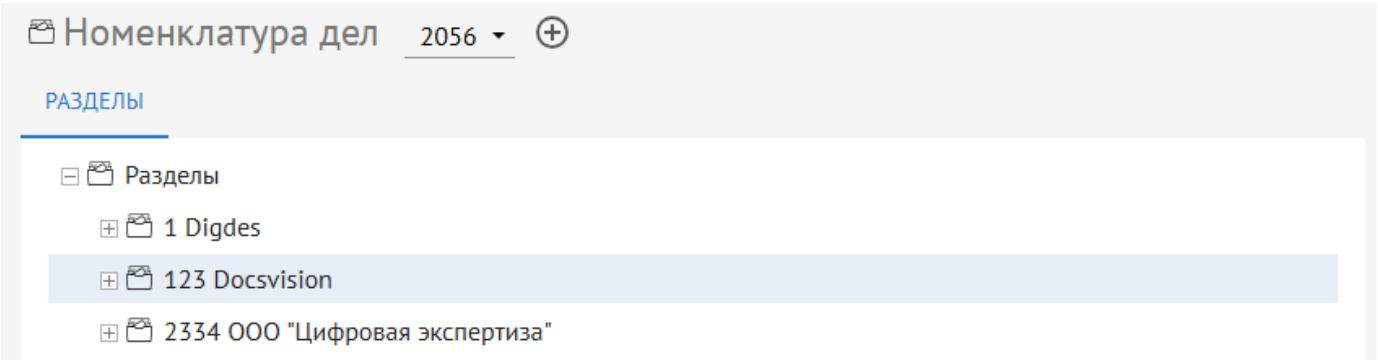


Рисунок 324. Копирование окончено

Операция Вырезать

Перемещает раздел со всеми дочерними элементами, включая дела. Раздел будет перемещён в новое место, имеющиеся внешние ссылки на раздел будут сохранены.

Скопированный или вырезанный раздел помещается в буфер обмена. Справа от раздела появляется кнопка **Отменить**, вместо пункта *Копировать* или *Вырезать* контекстного меню появляется пункт *Отменить*. Нажатие на эту кнопку или пункт в контекстном меню очищает буфер обмена.

Операция Вставить

Завершает копирование или вырезание. Для выполнения операции нужно сначала выбрать *Копировать* или *Вырезать* из контекстного меню исходного раздела. После этого необходимо выбрать операцию *Вставить* из контекстного меню целевого раздела.

Операция *Вставить* отправляет запрос на сервер. Это значит, что раздел будет скопирован или вырезан только после нажатия на *Вставить*.

Любой раздел одного года можно скопировать или вырезать и вставить в другой год. Включая корневой раздел.

Незавершённые операции будут отменены в следующих случаях:

- При повторном открытии справочника или обновлении страницы.
- При переходе на новую вкладку или при закрытии текущей.
- При нажатии кнопки `Esc` на клавиатуре.
- При выборе пункта *Отменить* из контекстного меню.
- При нажатии кнопки **Отменить** справа от элемента.
- При копировании или вырезании другого элемента.

При выполнении операций проверяются права доступа на редактирование разделов справочника. Чтобы вырезать раздел, пользователь должен иметь права на редактирование исходного раздела. Чтобы вставить раздел, пользователь должен иметь права на редактирование целевого раздела.

После выполнения операций настройки прав доступа исходного раздела перестанут действовать и будут заменены настройками, унаследованными от нового родительского раздела.

Справочник сотрудников

Справочник сотрудников позволяет создавать, редактировать и просматривать иерархическую структуру узлов организаций и подразделений, групп. Справочник позволяет создавать и объединять сотрудников в группы, назначать сотрудникам роли, а также просматривать и редактировать должности.

Чтобы иметь доступ к справочнику рекомендуется добавить ссылку на него в основное меню.

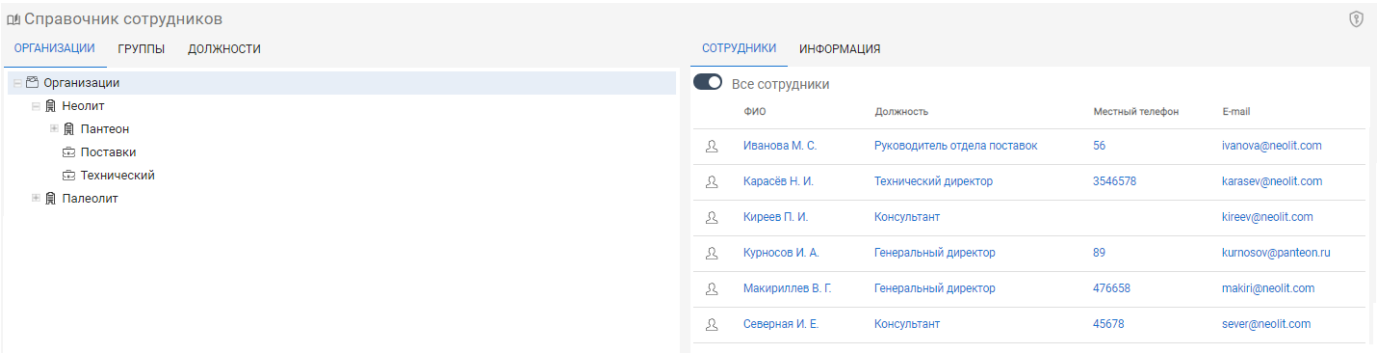


Рисунок 325. Общий вид справочника сотрудников

Справочник визуально поделён на две области. При наведении на границу областей изменяется вид указателя, границу можно сжать, потянув вправо или влево. Допускается сжатие вплоть до 10% от исходного.

В левой области находятся три вкладки: *Организации*, *Группы*, *Должности*, в каждой вкладке находятся узлы, отсортированные по алфавиту.

Каждая вкладка имеет свой корневой узел, объединяющий все остальные. Создавать новые узлы разрешается только внутри корневого узла, создать новый корневой узел невозможно.

Чтобы раскрыть узел, необходимо нажать на стрелку слева от его названия, повторное нажатие на стрелку скрывает узел. При однократном нажатии на название, узел переходит в фокус, а в правой области справочника отображаются сотрудники.

- Во вкладке *Организации* отображаются *Организации* и *Подразделения* в виде иерархической структуры или дерева.
- Во вкладке *Группы* отображаются *Группы* сотрудников также в виде иерархической структуры или дерева.
- Во вкладке *Должности* отображаются *Должности* из справочника в виде списка.

Правая область имеет две вкладки: *Сотрудники* и *Информация*.

- Во вкладке *Сотрудники* отображаются *Сотрудники*, относящиеся к выбранному в левой области узлу справочника: *Организации*, *Подразделению*, *Группе* или *Должности*.
- Во вкладке *Информация*, отображается информация об узле, открытом в левой области, например информация об *Организации*, *Подразделении*, *Группе* или *Должности*. Также информацию об узле справочника можно посмотреть, выбрав в контекстном меню нужного узла пункт *Информация*.

Над правой областью расположен элемент управления , позволяющий задать общие настройки дискреционной безопасности для справочника.

Работа с организациями

Основная информация о выбранной организации или подразделении доступна в правой области справочника, во вкладке *Информация*. Полную информацию можно посмотреть, выбрав пункт *Информация* в контекстном меню.

Посмотреть информацию об организации

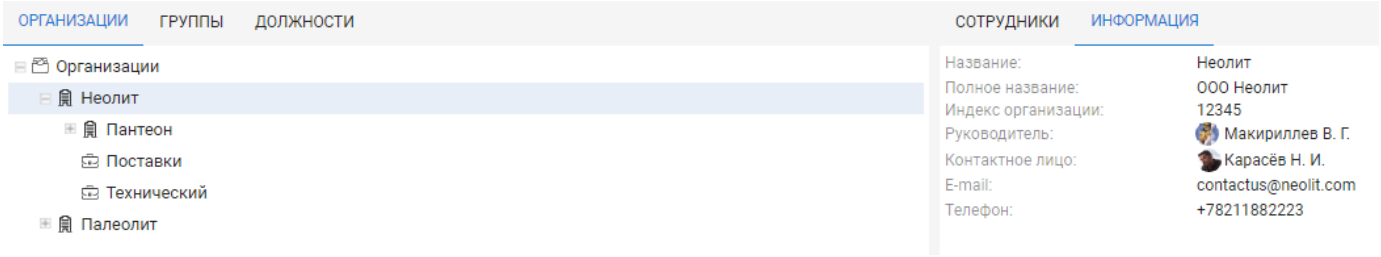


Рисунок 326. Просмотр основной информации об организации при открытой вкладке "Информация"

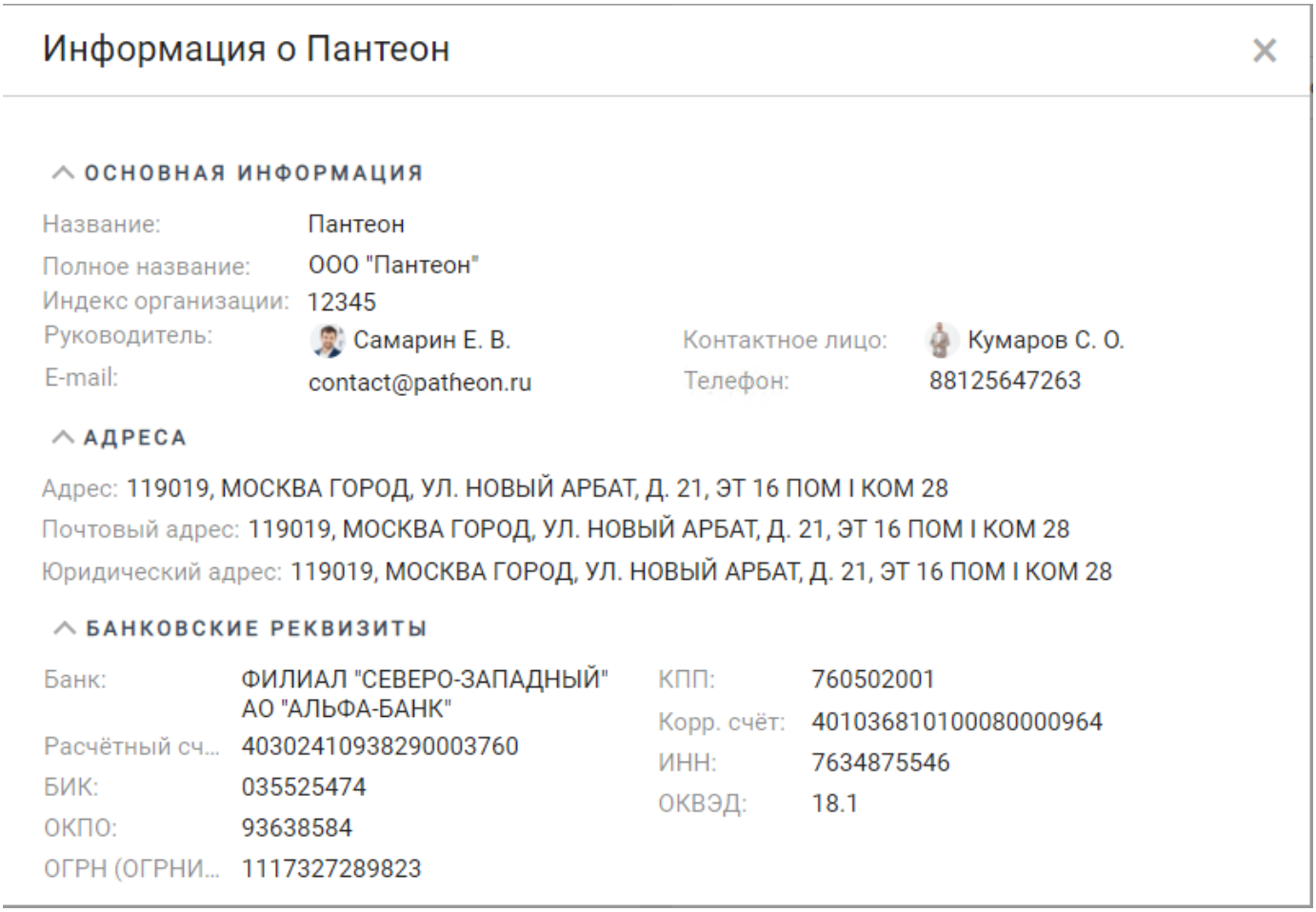


Рисунок 327. Просмотр полной информации об организации при выборе пункта контекстного меню "Информация"

В справочнике сотрудников предусмотрена возможность создания, изменения и удаления узлов организаций.



Возможность создания, изменения, удаления и просмотра узлов, а также добавления или изменения числа сотрудников в выбранном узле может быть ограничена настройками безопасности.

Создать новую организацию

Функции работы со справочником недоступны пока не создана организация.

Чтобы создать новый узел организации:

1. Перейдите в справочник сотрудников.
2. В левой области выберите вкладку **Организации**.

> Допускается создание как самостоятельных организаций, так и создание дочерних организаций.
>
> Организация не будет сохранена, пока не нажата кнопка **Создать**. Это значит, что в процессе создания организации в ней нельзя создавать сотрудников, включая руководителя организации.

3. Выберите необходимый узел.
4. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего узла.
5. Заполните все необходимые поля и нажмите **Создать**.

   Подробнее о полях карточки организации см. Перечень полей карточки организации.

Изменить организацию

Чтобы изменить созданный ранее узел организации:

1. На вкладке *Организации* в левой области справочника выберите узел из иерархического списка.
2. Выберите один из доступных способов изменения:
   - При помощи элемента управления  на вкладке *Информация* в правой области справочника.
   - Из соответствующего пункта контекстного меню текущего узла.
3. Измените необходимые поля и нажмите **Сохранить**.

Удалить организацию

Чтобы удалить созданный ранее узел организации:

1. На вкладке *Организации* в левой области справочника выберите узел из иерархического списка.
2. Вызовите контекстное меню на выбранном узле.
3. Выберите соответствующий пункт из контекстного меню, чтобы удалить узел.
4. Появится предупреждение, подтвердите действие кнопкой **ОК**.



Организации и подразделения удаляются со всеми дочерними узлами, включая сотрудников.

Перечень полей карточки организации

Основная информация об организации

Основная информация об организации может быть заполнена при создании новой организации или изменена после создания.

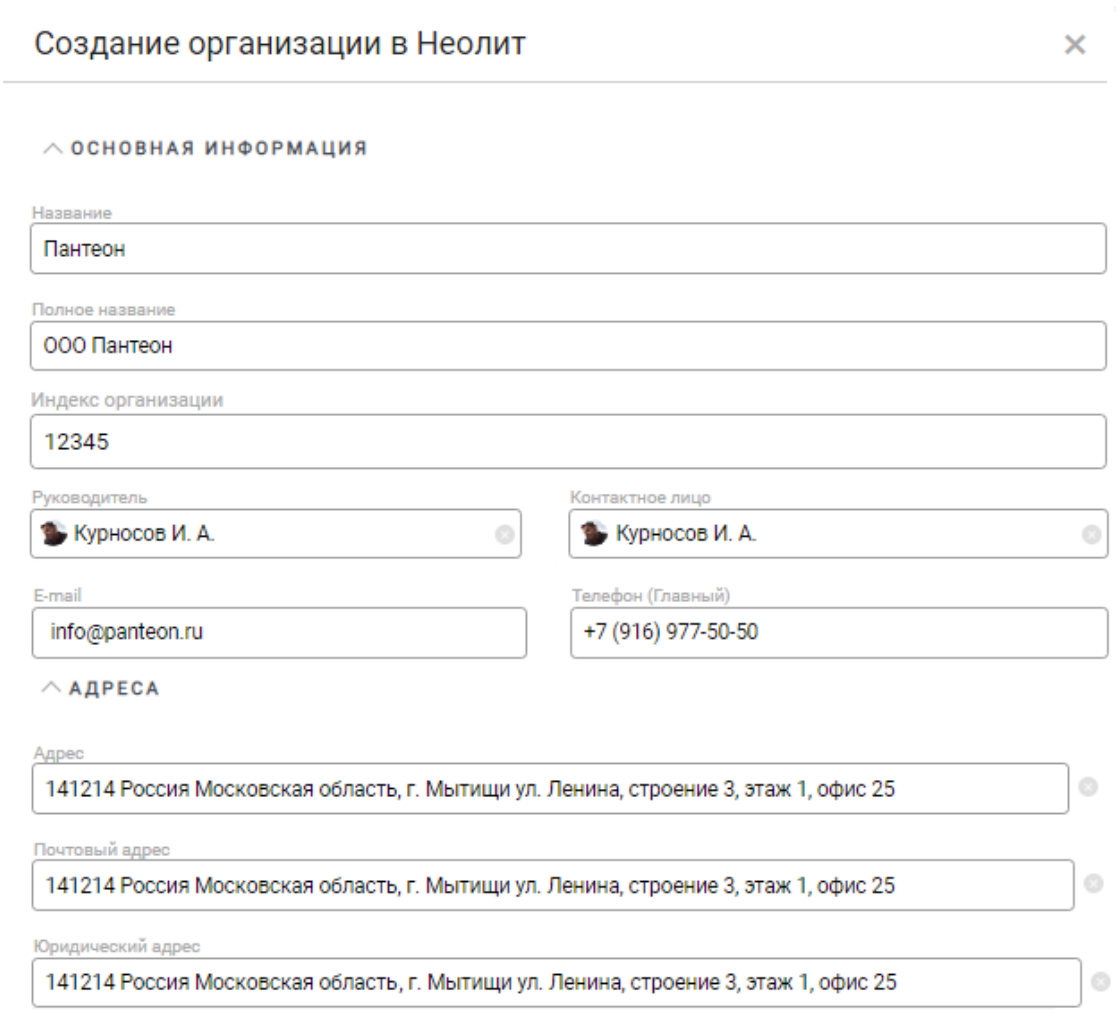


Рисунок 328. Основная информация об организации

Название	В поле указывается сокращенное название, которое будет отображаться в справочнике Docsvision.
Полное название	В поле указывается полное официальное название.
Индекс организации	В поле указывается индекс организации.
Руководитель	В поле указывается руководящий сотрудник. Нажмите на кнопку, чтобы выбрать сотрудника из справочника, или начните вводить ФИО сотрудника, чтобы активировать быстрый поиск.  Указанный руководитель не наследуется и не будет указан в соответствующих полях дочерних узлов. Руководитель не может быть указан из карточки сотрудника.
Контактное лицо	В поле указывается сотрудник, который является официальным представителем организации. Контактное лицо, как и руководителя, можно выбрать из справочника или при помощи быстрого поиска.
E-mail	В поле указывается адрес для отправки электронных писем.
Телефон	В поле указывается номер телефона для связи.
Адрес	В поле указывается фактический адрес. При изменении адреса в поле Адрес в разметке создания или изменения организации, информация будет скопирована в поля Почтовый адрес и Юридический адрес.
Почтовый адрес	В поле указывается адрес, который будет использован для отправки почтовой корреспонденции.
Юридический адрес	В поле указывается адрес регистрации организации.

Адрес

При выборе поля Адрес, Почтовый адрес или Юридический адрес из карточки организации открывается новое окно для добавления адреса:

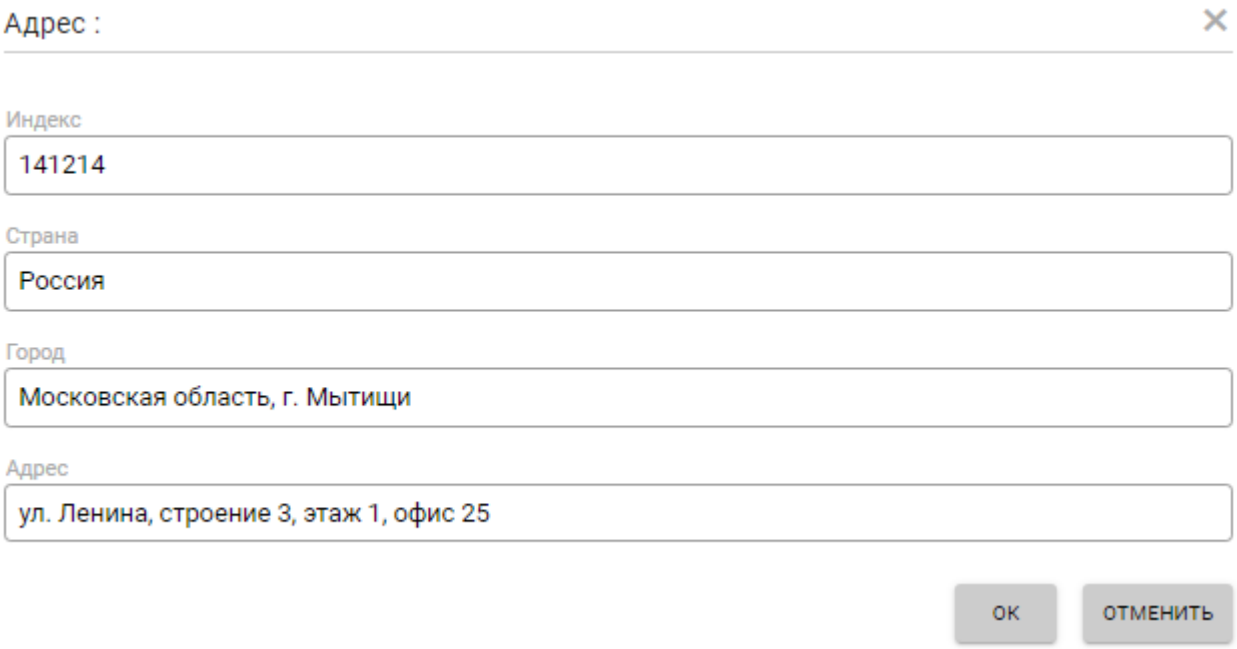


Рисунок 329. Окно добавления адреса организации

При изменении адреса в поле Адрес в разметке создания или изменения организации, информация будет скопирована в поля Почтовый адрес и Юридический адрес.

При добавлении адреса для заполнения доступны следующие поля:

Индекс Почтовый индекс организации.

Город Город, в котором организация находится.

Адрес Улица, дом, офис.

Страна Страна нахождения организации.

Банковские реквизиты

Банковские реквизиты для организации или подразделения можно заполнить при создании новой организации или изменить после создания:

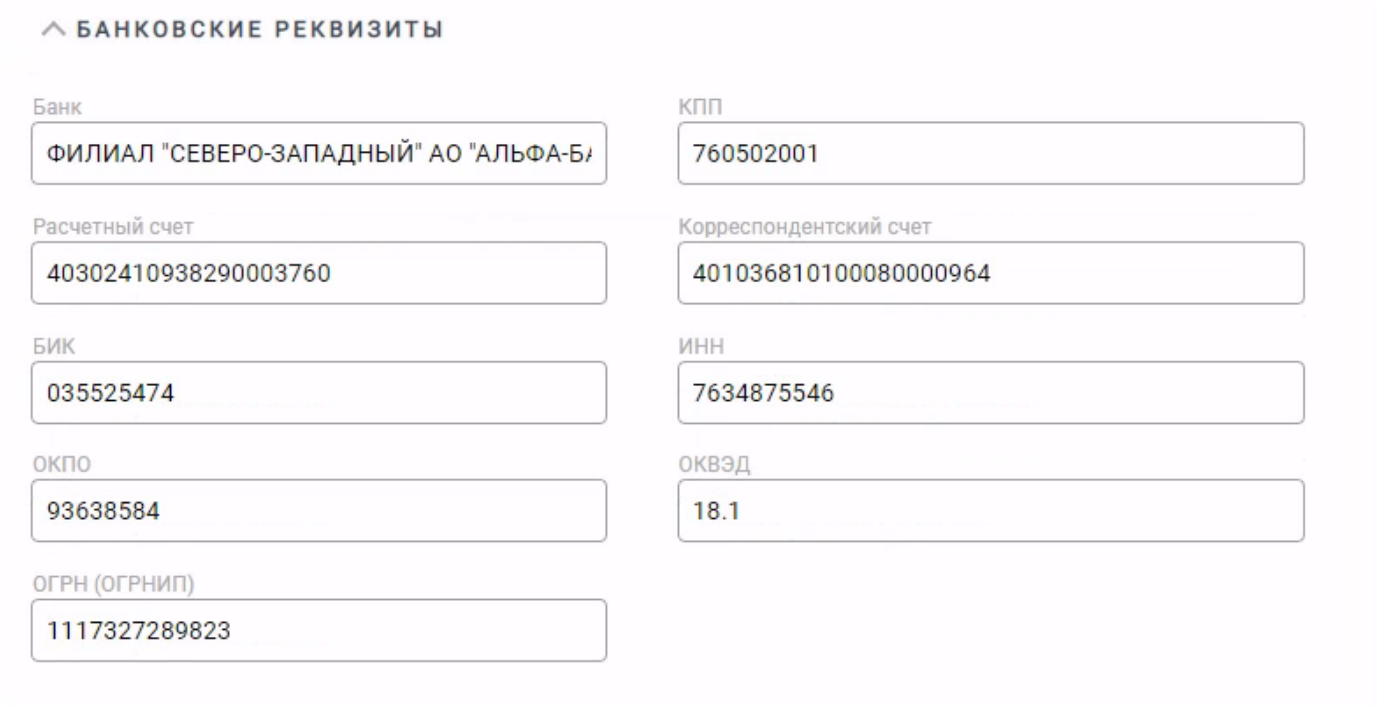


Рисунок 330. Банковские реквизиты организации

В банковских реквизитах компании можно указать следующие данные:

Банк

Название банка.

Расчетный счет

БИК

ОКПО

КПП

Корреспондентский счет

ИНН

ОКВЭД

ОГРН (ОГРНИП)

:!current:

Работа с подразделениями

В справочнике сотрудников реализована возможность создания подразделений организаций. Каждая организация может иметь неограниченное количество подразделений. Подразделения являются составной частью организаций и не могут быть созданы вне организации.

Посмотреть информацию о подразделении

Основная информация о выбранной организации или подразделении доступна в правой области справочника, во вкладке *Информация*. Полную информацию можно посмотреть, выбрав пункт *Информация* в контекстном меню.

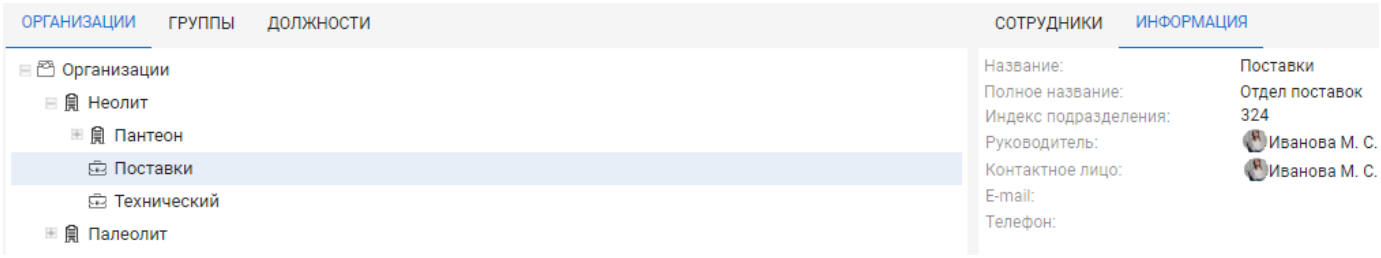


Рисунок 331. Просмотр основной информации о подразделении при открытой вкладке "Информация"

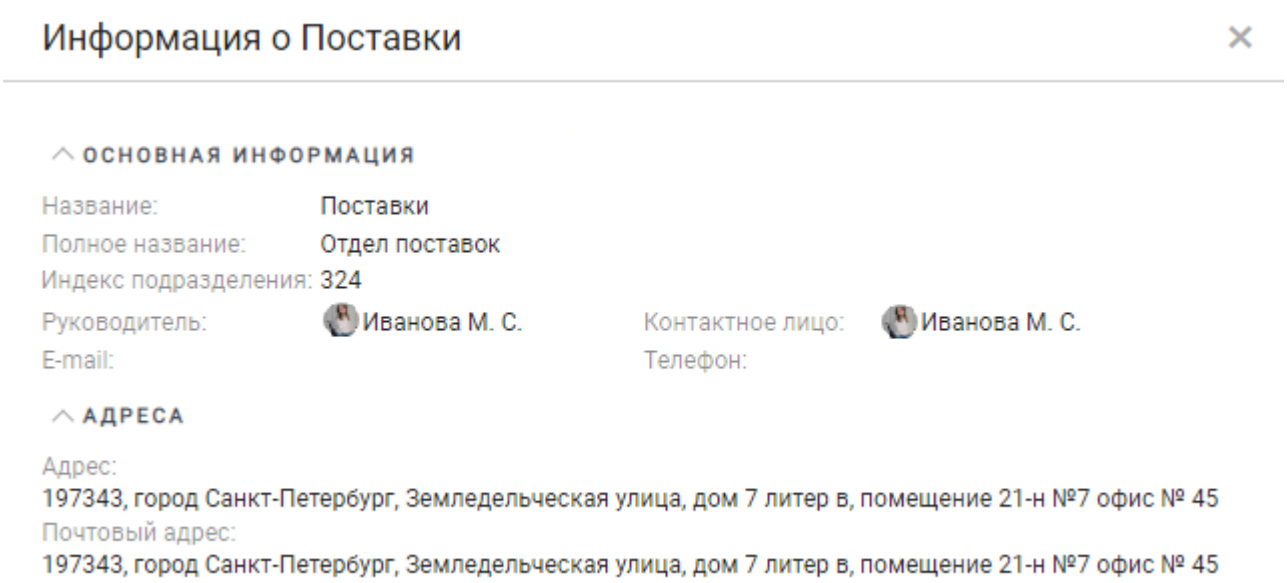


Рисунок 332. Просмотр полной информации о подразделении при выборе пункта контекстного меню "Информация"

В справочнике сотрудников предусмотрена возможность создания, изменения и удаления узлов подразделений.

Возможность создания, изменения, удаления и просмотра узлов, а также добавления или изменения числа сотрудников в выбранном узле может быть ограничена настройками безопасности.

Создать новое подразделение

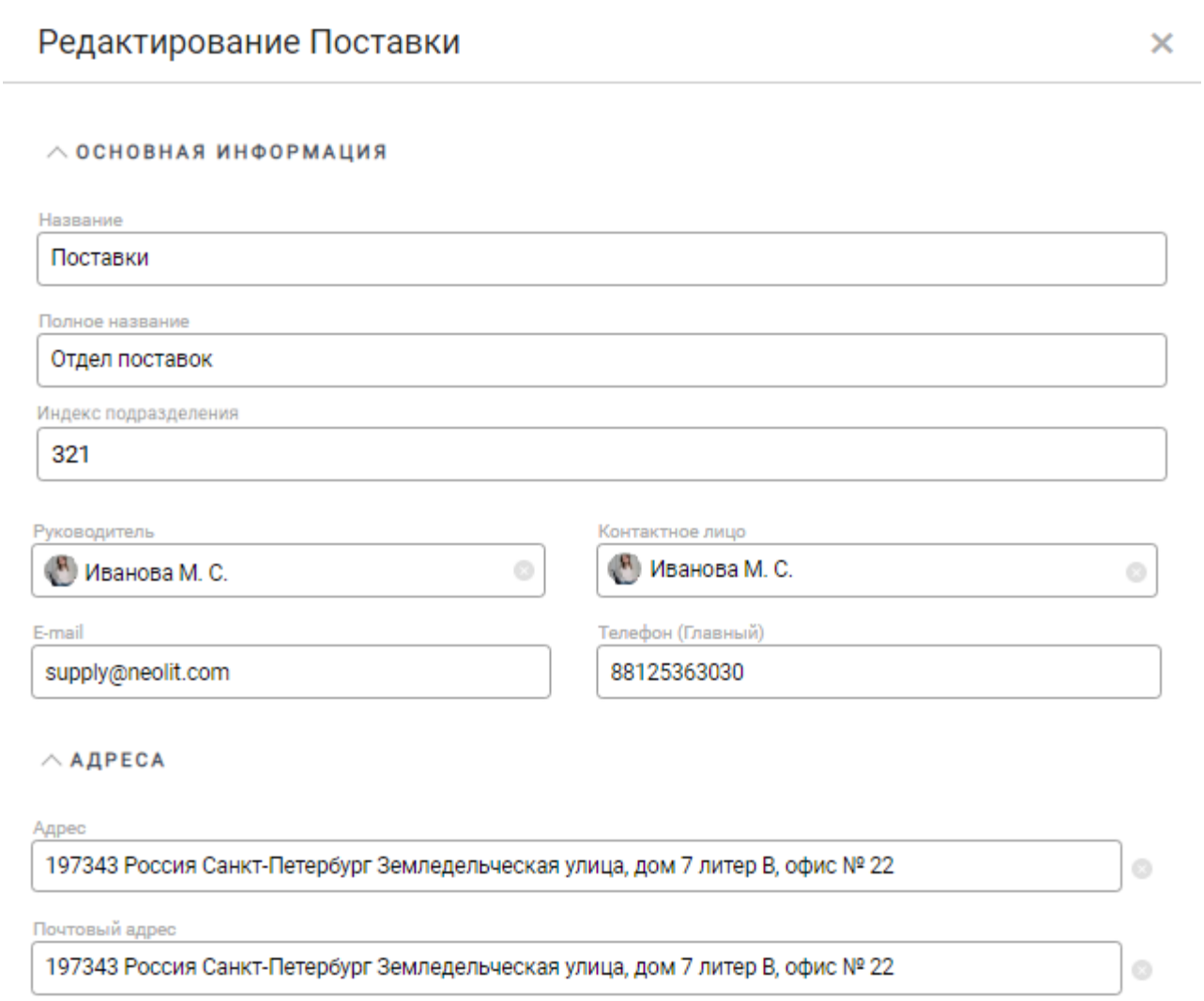


Рисунок 333. Создание нового подразделения

Чтобы создать новое подразделение:

1. Выберите необходимый узел.
2. Выберите один из доступных способов создания:
   - При помощи элемента управления ⊕.
   - Из соответствующего пункта контекстного меню текущего узла.
3. Заполните все необходимые поля и нажмите **Создать**.

> Внутри подразделений допускается создание дочерних подразделений.

Изменить подразделение

Чтобы отредактировать ранее созданный узел подразделения:

1. На вкладке **Организации** в левой области справочника выберите узел из иерархического списка.
2. Выберите Подразделение.
3. Выберите один из доступных способов изменения:
   - При помощи элемента управления на вкладке *Информация* в правой области справочника.
   - Из соответствующего пункта контекстного меню текущего узла.
4. Измените необходимые поля и нажмите **Сохранить**.

Удалить подразделение

Чтобы удалить ранее созданный узел подразделения:

1. На вкладке *Организации* в левой области справочника выберите узел из иерархического списка.
2. Вызовите контекстное меню на выбранном узле.
3. Выберите соответствующий пункт из контекстного меню, чтобы удалить узел.
4. Появится предупреждение, подтвердите действие кнопкой **ОК**.



Организации и подразделения удаляются со всеми дочерними узлами, включая сотрудников.

Перечень полей карточки подразделения

Основная информация о подразделении может быть заполнена при создании нового подразделения или изменена после создания.

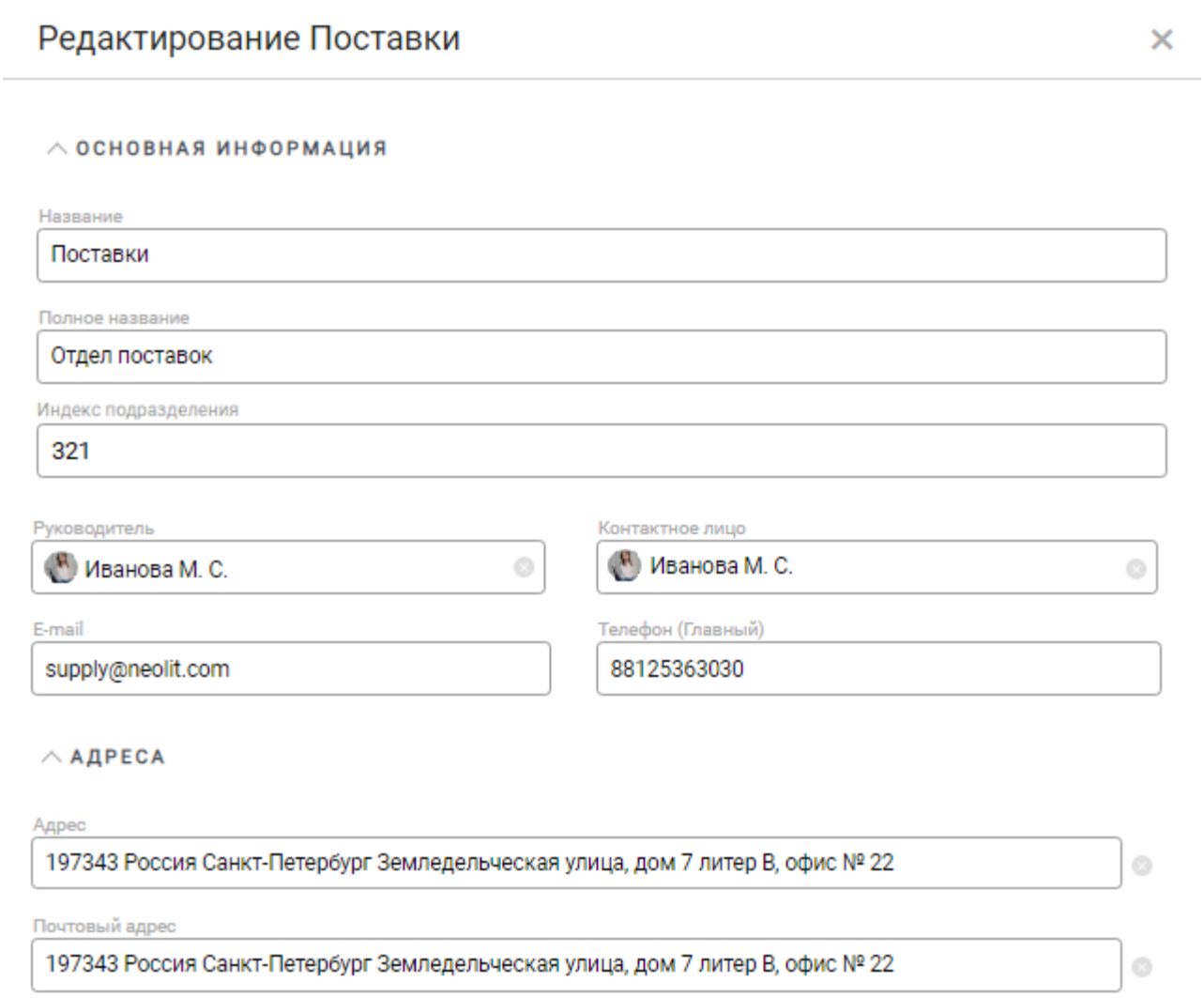


Рисунок 334. Основная информация о подразделении

Название

В поле указывается сокращенное название, которое будет отображаться в справочнике Docsvision.

Полное название

В поле указывается полное официальное название.

Индекс подразделения

В поле указывается индекс подразделения.

Руководитель

В поле указывается руководящий сотрудник. Нажмите на кнопку 🕮, чтобы выбрать сотрудника из справочника, или начните вводить ФИО сотрудника, чтобы активировать быстрый поиск.



Указанный руководитель не наследуется и не будет указан в соответствующих полях дочерних узлов.

Руководитель не может быть указан из карточки сотрудника.

Контактное лицо

В поле указывается сотрудник, который является официальным представителем подразделения. Контактное лицо можно выбрать из справочника или при помощи быстрого поиска, как и руководителя.

E-mail

В поле указывается адрес для отправки электронных писем.

Телефон

В поле указывается номер телефона для связи.

Адрес

В поле указывается фактический адрес. При изменении адреса в поле Адрес в разметке создания или изменения подразделения, информация будет скопирована в поле Почтовый адрес.

Почтовый адрес

В поле указывается адрес, который будет использован для отправки почтовой корреспонденции.



Поля Адрес и Почтовый адрес заполняются аналогично с полем Адрес в данных организации.

Работа с группами

В справочнике сотрудников реализована возможность создания, изменения и удаления групп. После создания в группу можно добавить сотрудников или оставить группу пустой.

Посмотреть информацию о группе

Вся информация о выбранной группе доступна в правой области справочника, во вкладке *Информация* или из пункта *Информация* в контекстном меню.

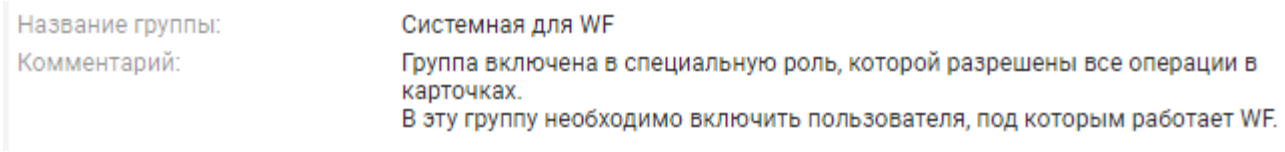


Рисунок 335. Просмотр информации о группе

Внутри одного узла группы может быть создано неограниченное количество других узлов групп с неограниченной глубиной вложенности.

> Возможность создания, изменения, удаления и просмотра узлов, а также добавления или изменения числа сотрудников в выбранном узле может быть ограничена настройками безопасности.

Создать новую группу

Чтобы создать новый узел группы:

1. В левой области справочника выберите вкладку **Группы**.
2. Выберите узел, в который необходимо добавить новую группу.
3. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего узла.
4. Заполните все необходимые поля и нажмите **Создать**.

Изменить группу

Чтобы изменить ранее созданный узел группы:

1. На вкладке *Группы* выберите группу, которую хотите изменить.
2. Выберите один из доступных способов изменения:
   - При помощи элемента управления на вкладке *Информация* в правой области справочника.
   - Из соответствующего пункта контекстного меню текущего узла.
3. Измените необходимые поля и нажмите **Сохранить**.

Удалить группу

Чтобы удалить ранее созданный узел группы:

1. На вкладке *Группы* в левой области справочника выберите из иерархического списка узел группы, который хотите удалить.
2. Вызовите контекстное меню на выбранном узле.

3. Выберите соответствующий пункт из контекстного меню, чтобы удалить узел.
4. Появится предупреждение, подтвердите действие кнопкой **ОК**.



При удалении группы сотрудники, входящие в группу, не удаляются, но перестают относиться к удалённой группе.

Перечень полей карточки группы

Карточка группы содержит два поля:

Название — Имя группы, которое будет отображаться в справочнике Docsvision.

Комментарий — Любая дополнительная информация, относящаяся к данной группе.



При работе с полем Комментарий поддерживается HTML синтаксис.

Скопировать группы сотрудников



Для копирования требуются права доступа на изменение групп и сотрудников. Если пользователь не обладает достаточными разрешениями, операция не будет выполнена.

В справочнике сотрудников реализована возможность копировать группы между сотрудниками.

При копировании происходит сравнение групп сотрудников. Скопированы будут только те группы, в которые сотрудник ещё не включён.

Чтобы скопировать группы сотрудников:

1. Перейдите в справочник сотрудников.
2. Выберите необходимый узел в левой области справочника.

   В правой области справочника будут отображены сотрудники, относящиеся к данному узлу.

3. Откройте карточку сотрудника в режиме редактирования, выбрав пункт *Редактировать* из контекстного меню узла.
4. В карточке сотрудника перейдите на вкладку *Дополнительно*.

- При помощи кнопки  выберите одного или нескольких сотрудников, группы которых требуется скопировать текущему сотруднику.

> Группы будут скопированы только после нажатия кнопки **Сохранить** в карточке текущего сотрудника.

- При помощи кнопки  выберите одного или нескольких сотрудников, которым необходимо скопировать группы текущего сотрудника.

> Группы копируются сразу в карточки выбранных сотрудников, без нажатия кнопки **Сохранить**, но после подтверждения изменений.

5. После выбора сотрудников нажмите **ОК**.

Добавить сотрудников в группу и удалить из группы

В справочнике сотрудников реализована возможность группировать сотрудников. Добавить сотрудника в группу можно двумя способами:

1. Из вкладки *Сотрудники* в правой области, когда в левой области открыта вкладка *Группы*.

> Данный способ позволяет добавить в одну группу несколько сотрудников.

2. Из карточки сотрудника.

> Данный способ позволяет добавить одного сотрудника в несколько групп.

Удаление выполняется также двумя способами, аналогично добавлению:

1. Первый способ позволяет удалить несколько сотрудников из одной группы.
2. Второй способ позволяет удалить одного сотрудника из нескольких групп.

Добавление сотрудников в группу

Добавить несколько сотрудников в группу из вкладки "Сотрудники"

Чтобы добавить несколько сотрудников в группу из вкладки Сотрудники:

1. В левой области справочника откройте вкладку *Группы*.

2. Выберите узел, в который необходимо добавить сотрудника.
3. В правой области выберите вкладку *Сотрудники*.
4. Выберите одного или несколько сотрудников:
   - При помощи быстрого поиска, начав вводить ФИО сотрудника в строке.
   - Из справочника, нажав на 🕮.

> Один сотрудник может относиться к нескольким группам.
>
> В группу можно добавить всех сотрудников определённой должности или узла организации, подразделения или другой группы. При этом сам выбранный узел в группу не добавляется.

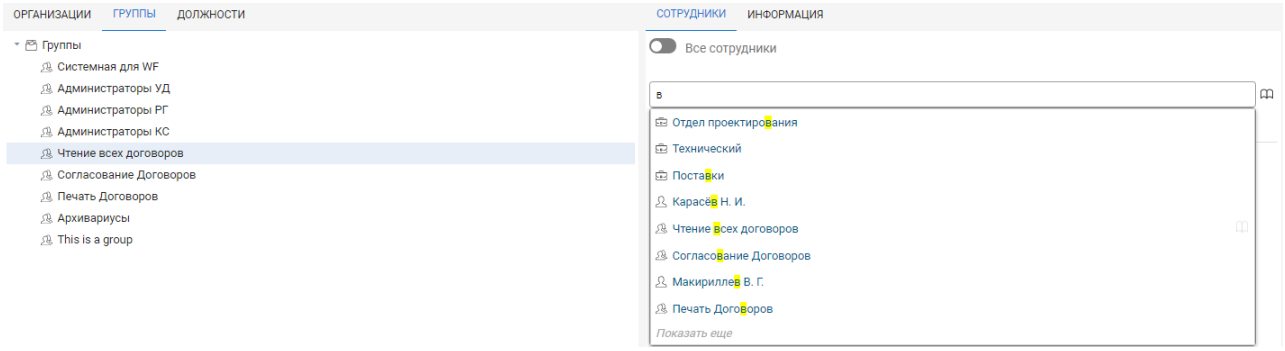


Рисунок 336. Добавление нескольких сотрудников в группу из вкладки "Сотрудники"

Добавить сотрудника в несколько групп из карточки сотрудника

Чтобы добавить сотрудника в несколько групп из карточки сотрудника:

1. Перейдите в справочник сотрудников.
2. Выберите необходимый узел.
3. Выберите:
   - Изменение добавленного ранее сотрудника.
   - Создание нового сотрудника.
4. Перейдите на вкладку *Дополнительно* карточки сотрудника.
5. В поле `Сотрудник включён в группу` выберите несколько групп:
   - Из быстрого поиска, начав вводить название группы в строке.
   - Из справочника сотрудников, нажав на 🕮.

> Один сотрудник может относиться к нескольким группам.

В группу можно добавить всех сотрудников определённой должности или узла организации, подразделения или другой группы. При этом сам выбранный узел в группу не добавляется.

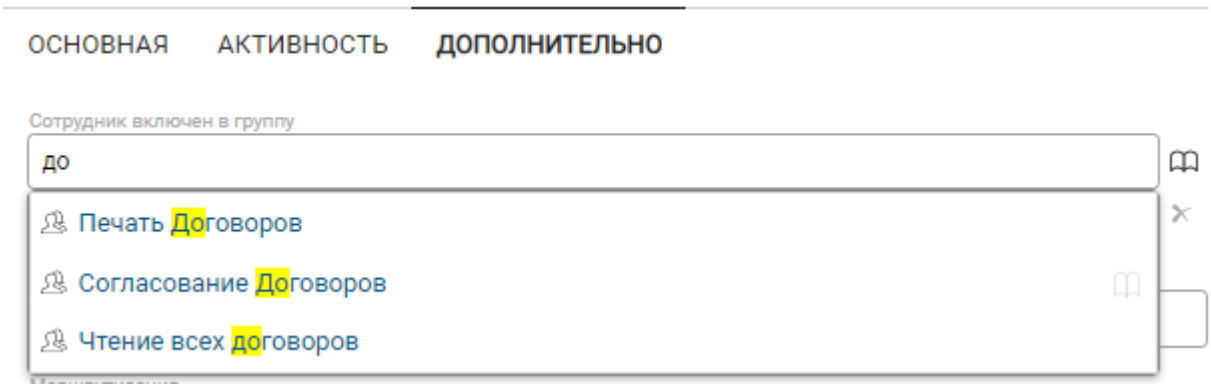


Рисунок 337. Добавление сотрудника в несколько групп из карточки сотрудника

Удаление сотрудников из группы

Удалить несколько сотрудников из группы на вкладке "Сотрудники"

Чтобы удалить нескольких сотрудников из группы на вкладке "Сотрудники":

1. В левой области справочника откройте вкладку *Группы*.
2. Выберите группу, в которой находится сотрудник.
3. В правой области найдите сотрудника в списке.
4. Удалите нескольких сотрудников из списка, нажав на ⊗.

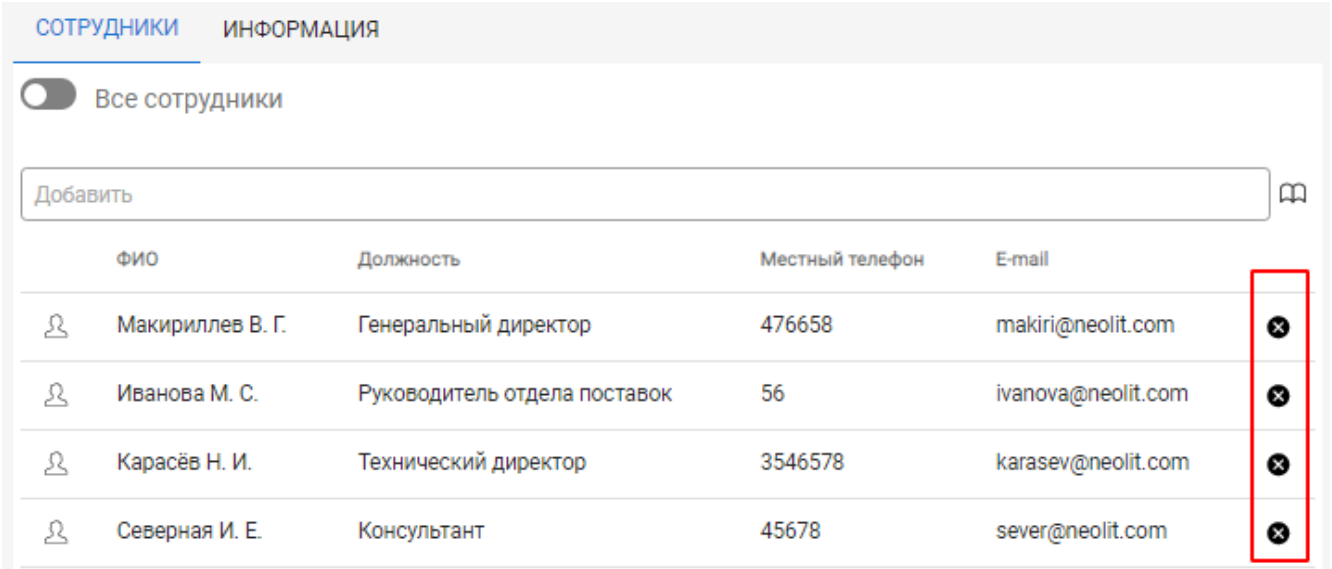


Рисунок 338. Удаление нескольких сотрудников из группы на вкладке "Сотрудники"

Удаление сотрудника из нескольких групп в карточке сотрудника

Чтобы удалить сотрудника из нескольких групп в карточке сотрудника:

1. Перейдите в справочник сотрудников.
2. Выберите необходимый узел.
3. Найдите сотрудника в правой области справочника.
4. Вызовите контекстное меню на сотруднике и выберите *Редактировать.*
5. На вкладке *Дополнительно* удалите сотрудника из нескольких групп при помощи ✕.

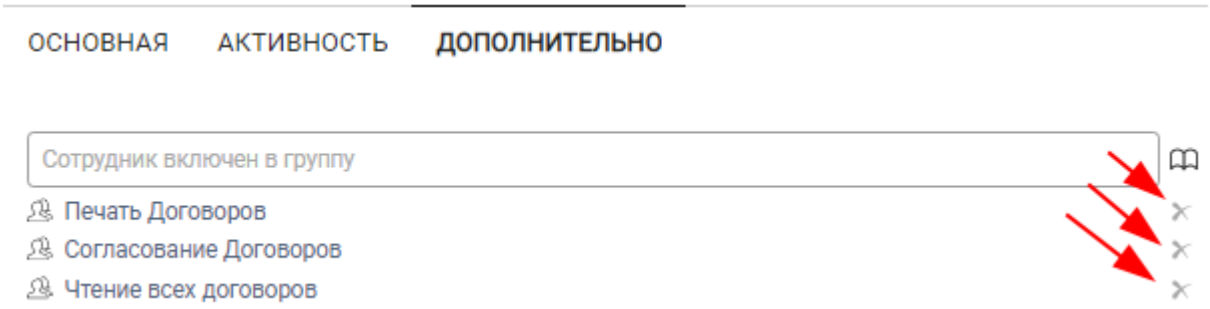


Рисунок 339. Удаление сотрудника из нескольких групп в карточке сотрудника

Работа с должностями

В справочнике сотрудников реализована возможность создания, изменения и удаления должностей. После создания, должность может быть назначена одному или нескольким сотрудникам. Должности отображаются в виде списка в левой области справочника, на вкладке *Должности*. Сотрудники, относящиеся к выбранной должности, отображаются в правой области справочника.

Посмотреть информацию о должности

Вся информация о выбранной должности доступна в правой области справочника, во вкладке *Информация* или из пункта *Информация* в контекстном меню.

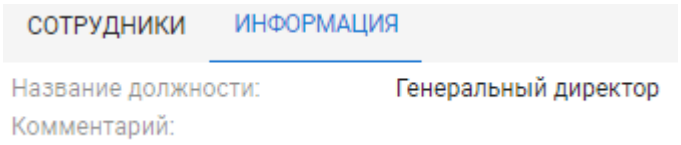


Рисунок 340. Просмотр информации о должности



Возможность создания, изменения, удаления и просмотра должностей может быть ограничена общими настройками безопасности справочника.

Создать новую должность

Чтобы создать новый узел должности:

1. Перейдите на вкладку *Должности* в левой области справочника.
2. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего узла.
3. Заполните все необходимые поля и нажмите **Создать**.

Назначить должность сотруднику можно из карточки сотрудника.

Изменить должность

Чтобы отредактировать созданный ранее узел должности:

1. На вкладке *Должности* выберите узел должности, который хотите изменить.
2. Выберите один из доступных способов изменения:
   - При помощи элемента управления  на вкладке *Информация* в правой области справочника.
   - Из соответствующего пункта контекстного меню текущего узла.
3. Измените необходимые поля и нажмите **Сохранить**.

Удалить должность

Чтобы удалить созданный ранее узел должности:

1. На вкладке *Должности* выберите узел должности, который хотите удалить.
2. Вызовите контекстное меню на выбранном узле.
3. Выберите соответствующий пункт из контекстного меню, чтобы удалить узел.
4. Появится предупреждение, подтвердите действие кнопкой **ОК**.



При удалении должности сотрудники, у которых данная должность указана в карточке, удалены не будут, но поле `Должность` в карточке сотрудника окажется пустым.

Перечень полей карточки должности

Карточка должности содержит два поля:

Название должности — Имя должности, которое будет отображаться в справочнике Docsvision.

Комментарий	Любая дополнительная информация, относящаяся к данной должности.



При работе с полем Комментарий поддерживается HTML синтаксис.

Работа с сотрудниками

Содержимое вкладки *Сотрудники*, отображаемой в правой области справочника, зависит от узла, выбранного в левой области. На вкладке *Сотрудники* отображаются не все сотрудники, доступные в справочнике, а только те, которые относятся к выбранной организации, подразделению и т.д.

Подробно о том, как заполнить поля на вкладках карточки сотрудника и информацию о них, см. в пункте "Поля и вкладки карточки сотрудника".

Посмотреть информацию о сотруднике

Когда узел из левой области находится в фокусе, во вкладке *Сотрудники* отображаются колонки по умолчанию: *ФИО*, *Должность*, *Местный телефон*, *E-mail*. Все сотрудники отсортированы в алфавитном порядке по первому столбцу.

При активации переключателя Все сотрудники во вкладке отображаются все сотрудники, подчинённые текущему выбранному узлу, например, сотрудники организации и всех входящих в организацию подразделений. Кнопка отображается, только если в левой области открыта вкладка *Организации* или *Группы*.

Вся информация о сотруднике доступна из пункта *Информация* в контекстном меню или по клику на строку сотрудника в правой области справочника.

В поле *Руководитель подразделения* отображается руководитель ближайшего узла организации или подразделения, к которому относится сотрудник.



Рисунок 341. Просмотр информации о сотруднике

Возможность создания, изменения, удаления и просмотра сотрудников может быть ограничена общими настройками безопасности справочника.

Краткая информация о сотруднике

Для карточки сотрудника может использоваться разметка в режиме Справка. Такая разметка позволяет отображать краткую информацию о текущем состоянии сотрудника.

В Web-клиент по умолчанию существует два вида разметок: *Просмотр информации о …* и *Справка информации о …*. В стандартной поставке разметки отличаются лишь способом вызова, однако формально являются разными разметками, которые вызываются разным способом.

При выборе значения из справочника нажатие кнопки ⋮ **> Информация** или пункта *Информация* в контекстном меню узла вызывает разметку *Просмотр информации о …*.

При нажатии кнопки ℹ в результатах быстрого поиска или наведение курсора на выбранный элемент в разметке (например 🏢) документа

вызывает разметку *Справка информации о …*, содержащую краткую информацию об узле.



Настроить отображение краткой информации о сотруднике можно в программе Конструктор Web-разметок, настроив разметку **BackOffice** › **Справочники** › **Справочник сотрудников** › **Подразделения** › **Сотрудники** с режимом *Справка*.

Краткая информация о сотруднике выводится при наведении на строку сотрудника во всех разметках, где используется один из следующих элементов управления:

- Записи справочника сотрудников.
- Партнер.
- Подразделение контрагента.
- Дата/время.
- Строка конструктора справочников.
- Нумератор.
- Дело.

Краткая информация о сотруднике отображает основную информацию:

- ФИО.
- Фото.
- Должность.
- E-mail.
- Телефон.

В краткой информации может отображаться также текущий статус сотрудника, например, если сотрудник болен или в отпуске:

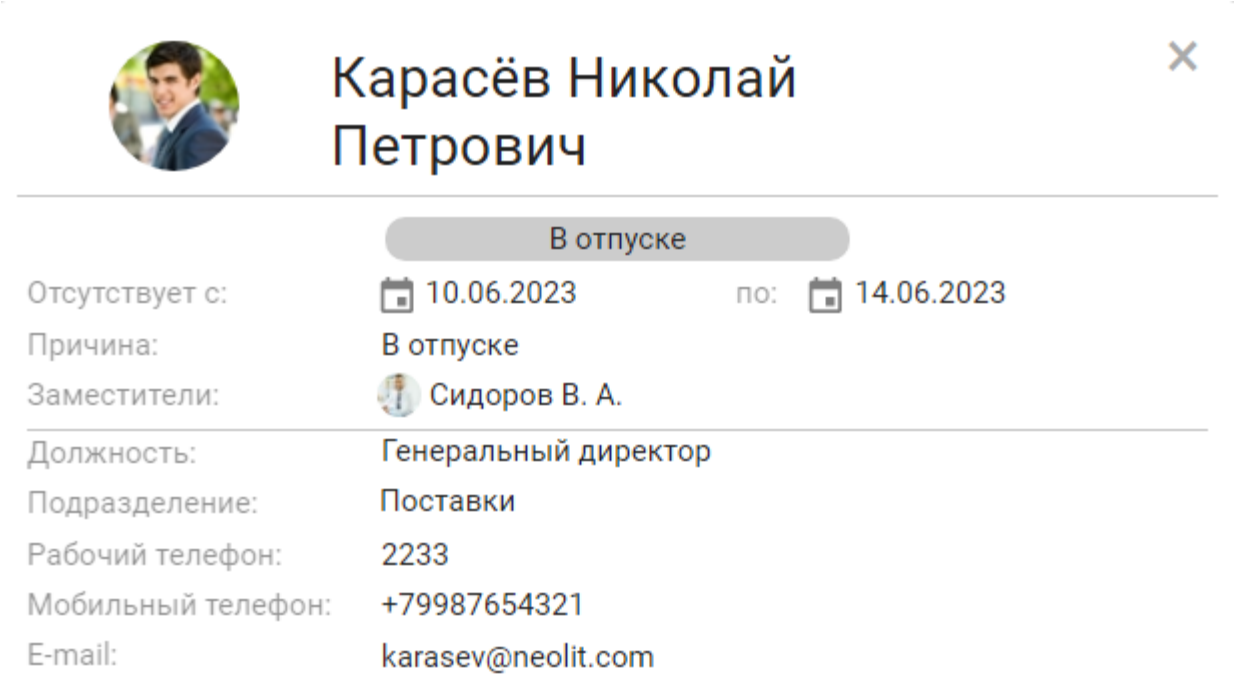


Рисунок 342. Отображение статуса сотрудника в режиме справки

Информация о периоде и причине отсутствия, заместителях сотрудника отображается, только если в профиле или справочнике сотрудников задан период недоступности.

Создать нового сотрудника

Возможность создания новых сотрудников доступна, только если в левой области открыта вкладка *Организации* и выбран узел организации или подразделения. Для каждой учётной записи в ActiveDirectory может быть создана только одна карточка сотрудника в *Справочнике сотрудников*.

Чтобы создать нового сотрудника:

1. Откройте справочник сотрудников.
2. На вкладке *Организации* в левой области справочника выберите узел из иерархического списка.
3. Выберите один из доступных способов создания:
   - При помощи элемента управления .
   - Из соответствующего пункта контекстного меню текущего узла.

Название организации или подразделения, к которым будет добавлен сотрудник, указано в заголовке окна.

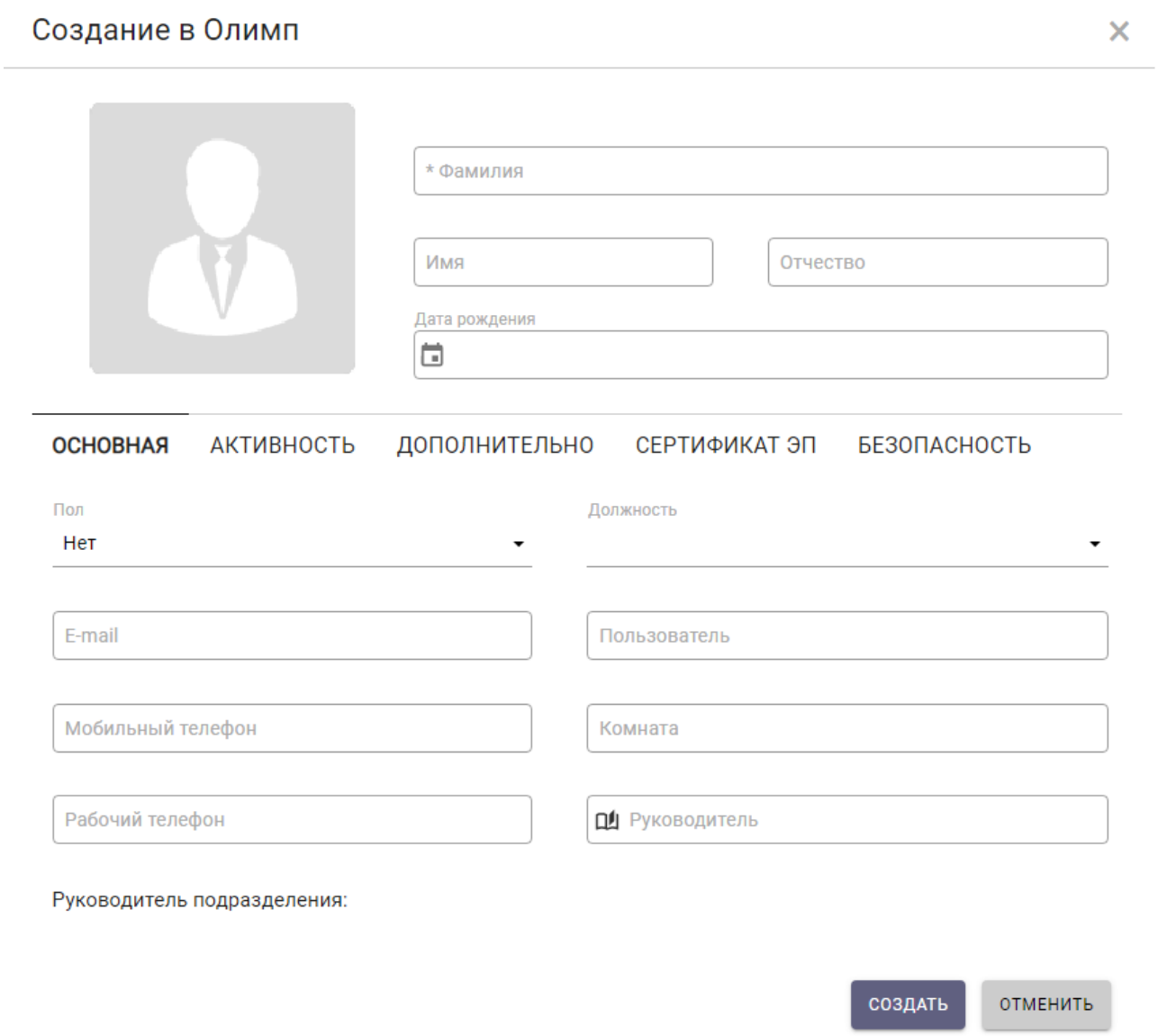


Рисунок 343. Создание нового сотрудника

4. Последовательно заполните доступные поля карточки сотрудника.
5. При необходимости добавьте фотографию сотрудника.

Добавить фотографию сотрудника

Чтобы добавить фотографию сотрудника:

1. Откройте карточку сотрудника в режиме создания или редактирования.
2. Наведите курсор на блок фотографии.

> Появится подсказка `Загрузите изображение`.

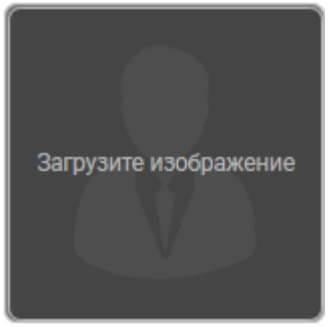


Рисунок 344. Блок добавления фотографии сотрудника

3. Выберите файл с фотографией сотрудника из файловой системы устройства, чтобы добавить фотографию в личную карточку сотрудника.

> Поддерживаются файлы изображений в форматах: `.jpeg` и `.jpg`. Рекомендуемый размер фотографии 130x130 пикселей, изображения большего размера будут сжаты до соответствующего разрешения.

> Если в Windows-клиенте к карточке сотрудника прикреплено несколько фотографий, в Web-клиенте отображается только первая из добавленных.

Вы можете заменить фотографию на новую, повторно наведя курсор на фотографию, или удалить текущую фотографию.

Удалить фотографию сотрудника

Чтобы удалить фотографию сотрудника:

1. Откройте карточку сотрудника в режиме редактирования.
2. Наведите курсор на блок фотографии.

> Поверх фото появится подсказка `Загрузите изображение`, а в верхнем правом углу появится кнопка .

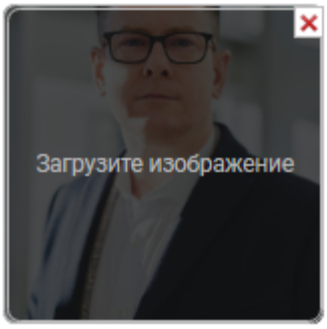


Рисунок 345. Удаление фотографии сотрудника

3. Нажмите на кнопку , чтобы удалить фотографию сотрудника.

Вы также можете заменить текущую фотографию новой.

Изменить сведения о сотруднике

Чтобы изменить сведения о сотруднике:

1. Откройте справочник сотрудников на нужной вкладке.
2. Выберите нужного сотрудника из списка в правой области.
3. Вызовите контекстное меню на нужном сотруднике и выберите *Редактировать*.
4. Внесите необходимые изменения в доступные поля карточки сотрудника.
5. Выйдите из режима редактирования, нажав **Сохранить**.

Удалить сотрудника

Удалять сотрудника не рекомендуется, вместо этого предлагается установить статус неактивности, см. подробнее "Период недоступности сотрудника и замещение".

Чтобы удалить ранее созданного сотрудника:

1. В правой области справочника выберите сотрудника, которого необходимо удалить.
2. Вызовите контекстное меню на выбранном сотруднике.
3. Выберите в контекстном меню пункт *Удалить*.
4. Появится предупреждение, подтвердите действие кнопкой **ОК**.

Сотрудник будет полностью удалён из справочника без возможности восстановления. Если вы не хотите удалять сотрудника, его можно перевести в состояние временной

неактивности.

Задать тип маршрутизации для сотрудника

Маршрутизация для сотрудника может быть задана во вкладке *Дополнительно* карточки сотрудника. Для работы маршрутизации необходимо указать E-mail сотрудника.

Типы маршрутизации

Онлайн задание

Задание приходит в настроенную виртуальную папку Docsvision.

Задача Почтового клиента

На настроенный электронный почтовый ящик приходит письмо со ссылкой для доступа к карточке вида *Задание в системе Docsvision*. Также пользователю предоставляется возможность выполнять задания непосредственно в почтовом клиенте с помощью ответных писем.

Не маршрутизировать

При отправке задания в системной папке *Карточки* (в ветке Задание бизнес-процесса библиотеки Делопроизводство) появляется карточка задания. Вся работа пользователя с заданием осуществляется в рамках работы с приложением Web-клиент.

Письмо с вложениями

На настроенный электронный почтовый ящик приходит письмо со ссылкой для доступа к карточке вида *Задание* в системе Docsvision и текстом, который указан в XSLT-преобразовании.

Ссылка на задание

На настроенный электронный почтовый ящик приходит письмо со ссылкой для доступа к карточке вида *Задание* в системе Docsvision.

Поля и вкладки карточки сотрудника

Поля с общей информацией в карточке сотрудника

Поля общих данных сотрудника всегда отображаются в верхней области карточки. Содержимое области, расположенной под общими данными, зависит от выбранной вкладки, расположенной в нижней части карточки: *Основная*, *Активность*, *Дополнительно*.

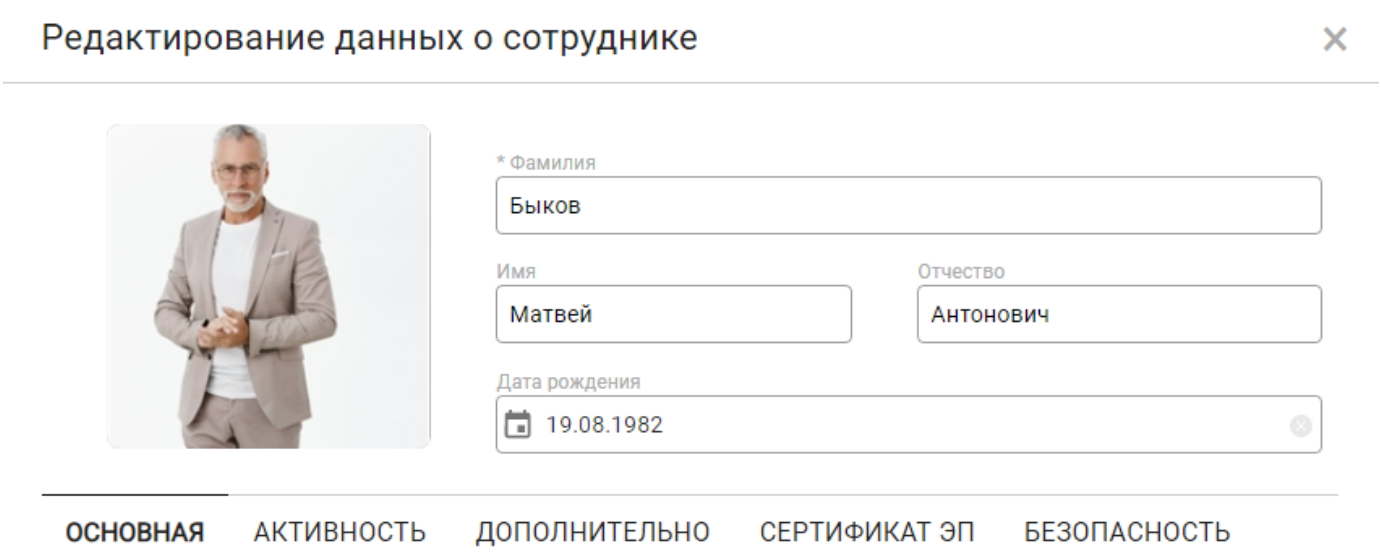


Рисунок 346. Общая информация и вкладки карточки сотрудника в режиме редактирования

Фото	Область фотографии слева от общих данных предназначена для прикрепления фотографии к личной карточке сотрудника.
Фамилия, Имя и Отчество	Поля для указания ФИО сотрудника. Обязательным для заполнения является только поле `Фамилия`.
Дата рождения	Можно выбрать дату рождения сотрудника из календаря.

Поля вкладки "Основная" в карточке сотрудника

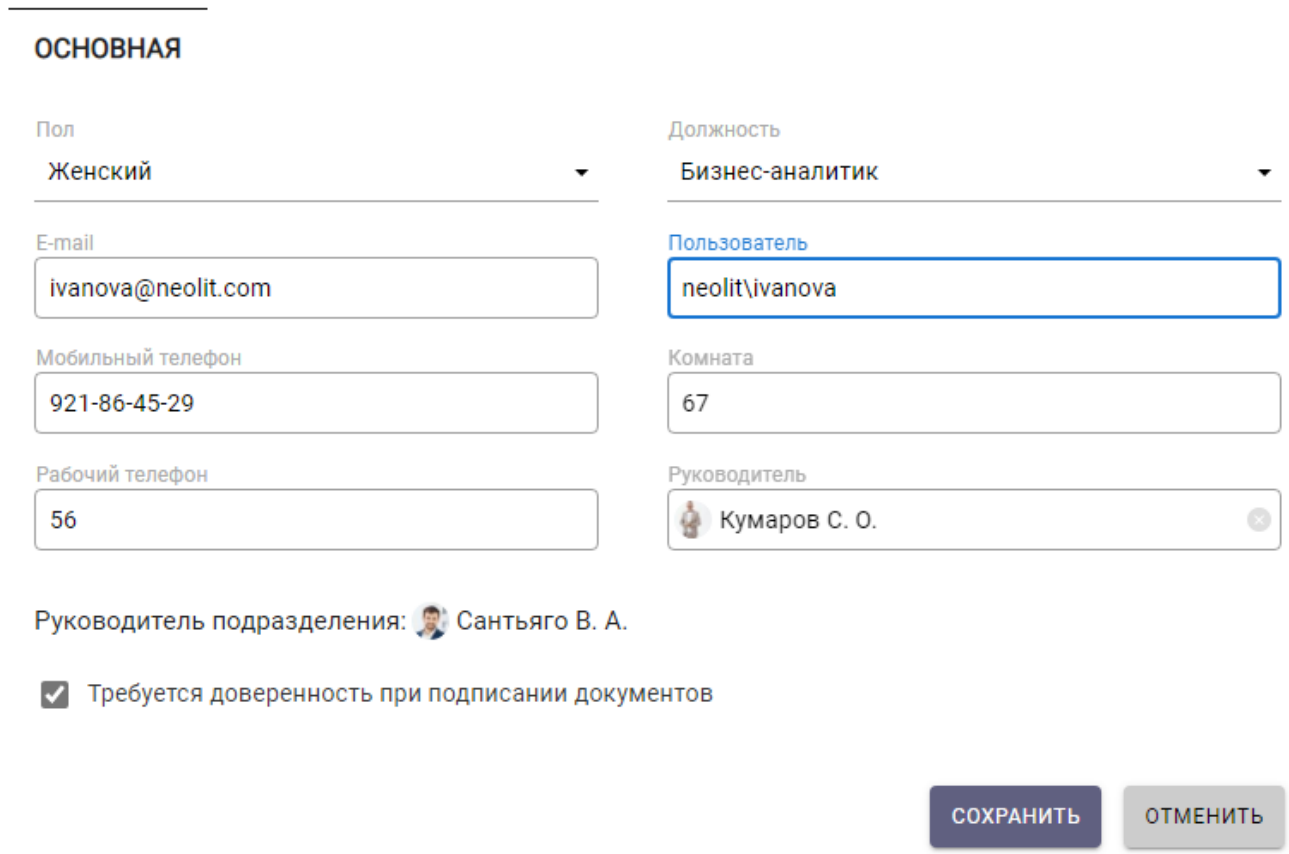


Рисунок 347. Карточка сотрудника на вкладке "Основная" в режиме редактирования

Пол	В поле указывается пол сотрудника. Пол можно выбрать из двух значений: ***Мужской*** и ***Женский***, либо не указывать, оставив в значении ***Нет***.
E-mail	Адрес электронной почты сотрудника нужен, например, для возможности отправки заданий по почте, если включена маршрутизация. Можно указать несколько адресов без пробелов, разделив их символом ;. При этом, если включена маршрутизация, задания будут отправляться на все указанные адреса.
Должность	Должность сотрудника можно выбрать из списка. Если необходимая должность не представлена в списке, добавьте её.

Пользователь	Поле связывает карточку сотрудника с учётной записью сотрудника в службе доменов. Учётная запись обязательно должна быть указана полностью, то есть содержать название домена и имя компьютера. Указание только имени учётной записи сотрудника допускается, если учётная запись создана на текущем компьютере. [CAUTION] ==== Для аутентификации при использовании коннектора к ALD Pro необходимо указывать учётные данные пользователя в формате UID: `user@domain` в том числе и в справочнике сотрудников, см. подробнее в документации Windows-клиента и Web-клиента. ==== WARNING: Поле является обязательным. Если доменная УЗ отсутствует, поле можно заполнить произвольными данными. В остальных случаях УЗ обязательно указывать полностью, включая название домена или имя компьютера (для локального пользователя).
Мобильный телефон и Рабочий телефон	В поле указываются номера телефонов для связи с сотрудником.
Комната	В поле указывается номер помещения, в котором работает сотрудник.
Руководитель	Указывается непосредственный руководитель сотрудника. Нажмите на кнопку 🕮, чтобы выбрать сотрудника из справочника, или начните вводить ФИО сотрудника, чтобы активировать быстрый поиск. ⚠ Сотрудник не может быть руководителем самого себя. Сотрудник также не может быть руководителем своего руководителя.

Руководитель подразделения	В поле отображается руководитель ближайшего узла организации или подразделения, к которому относится сотрудник. Поле заполняется автоматически и не может быть изменено.
Требуется доверенность при подписании документов	По умолчанию флаг установлен. Флаг влияет на алгоритм выбора МЧД при подписании документа и учитывается при проверке необходимости использования МЧД для сотрудника. • Если флаг установлен, выполняется переход к следующим этапам. • Если флаг не установлен, процесс подбора завершается, МЧД не требуется, дальнейшие операции не выполняются.

Поля вкладки "Активность" в карточке сотрудника

Во вкладке *Активность* карточки сотрудника доступна следующая информация:

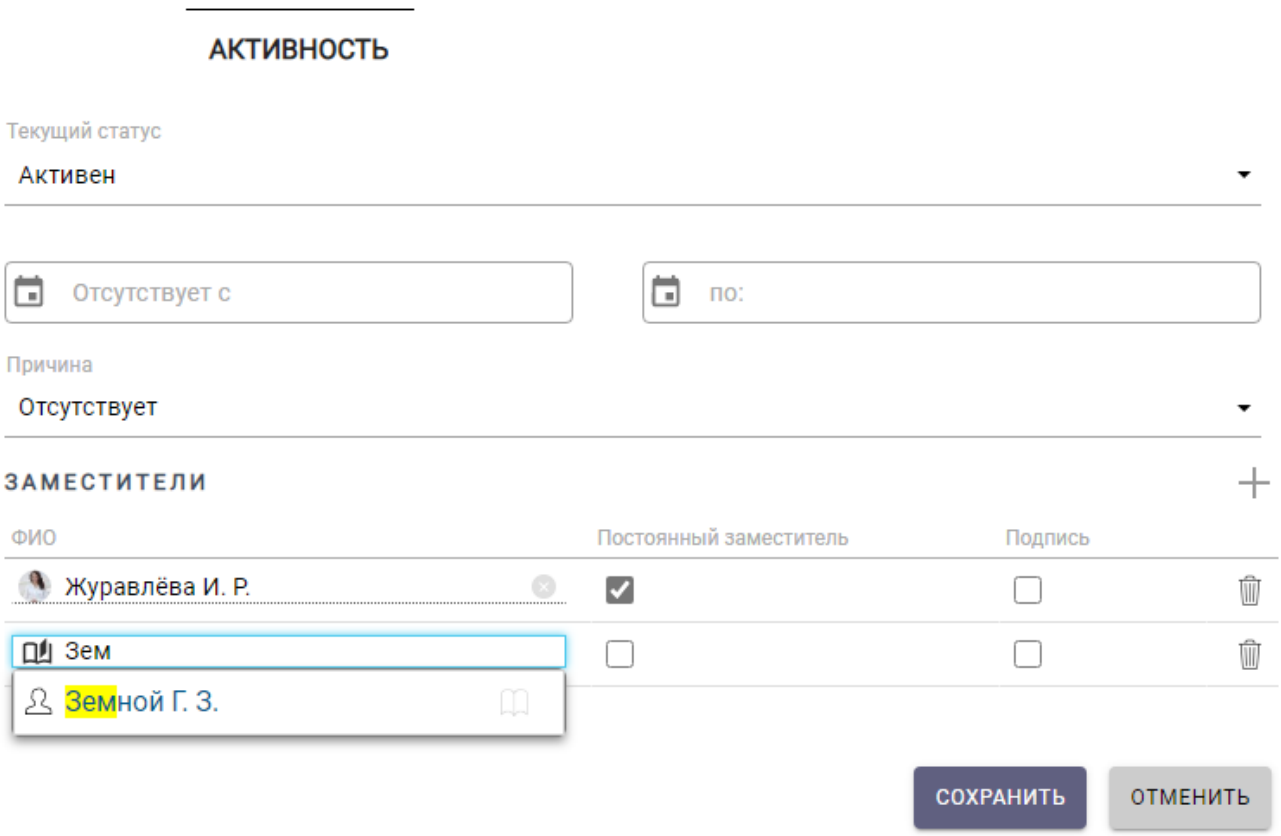


Рисунок 348. Вкладка "Активность" в режиме редактирования

Текущий статус	Статус сотрудника может быть выбран из списка в режиме редактирования или создания, а также задан самим сотрудником из профиля сотрудника. Подробнее о статусах сотрудников см. ниже.
Период недоступности сотрудника	В поле под статусом задаётся период недоступности сотрудника. Подробнее о доступности сотрудника в пункте Период недоступности сотрудника и замещение.
Заместители	В нижней части окна сотрудника отображается область *Заместители*. В данной области отображаются сотрудники, уполномоченные замещать текущего сотрудника (см. недоступность и замещение). При просмотре информации о сотруднике период и причина отсутствия, заместители сотрудника отображаются, только если задан период недоступности. Если период недоступности не задан, отображается только статус.

Поля вкладки "Дополнительно" в карточке сотрудника

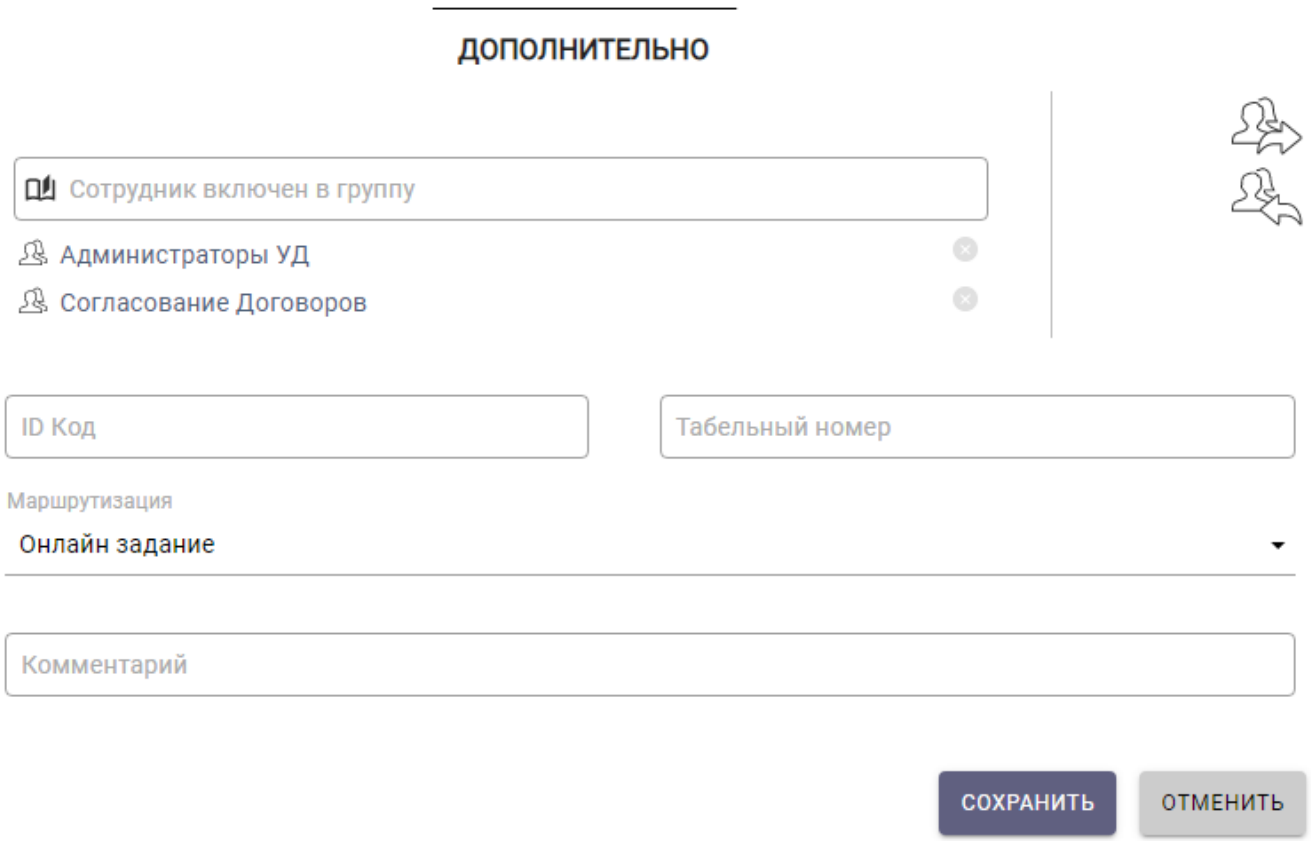


Рисунок 349. Вкладка "Дополнительно" в режиме редактирования

Сотрудник включён в группу	Поле позволяет добавить сотрудника в группу или исключить из группы.
ID код	Поле позволяет ввести уникальный идентификатор сотрудника.
Табельный номер	Поле предназначено для добавления табельного номера сотрудника.
Маршрутизация	При необходимости из раскрывающегося списка поля есть возможность задать путь, который будет использоваться по умолчанию для доставки сотруднику заданий или уведомлений о заданиях.
Комментарий	Поле предназначено для добавления дополнительной информации к карточке сотрудника. Поле поддерживает HTML синтаксис.

Поля вкладки "Сертификат ЭП"

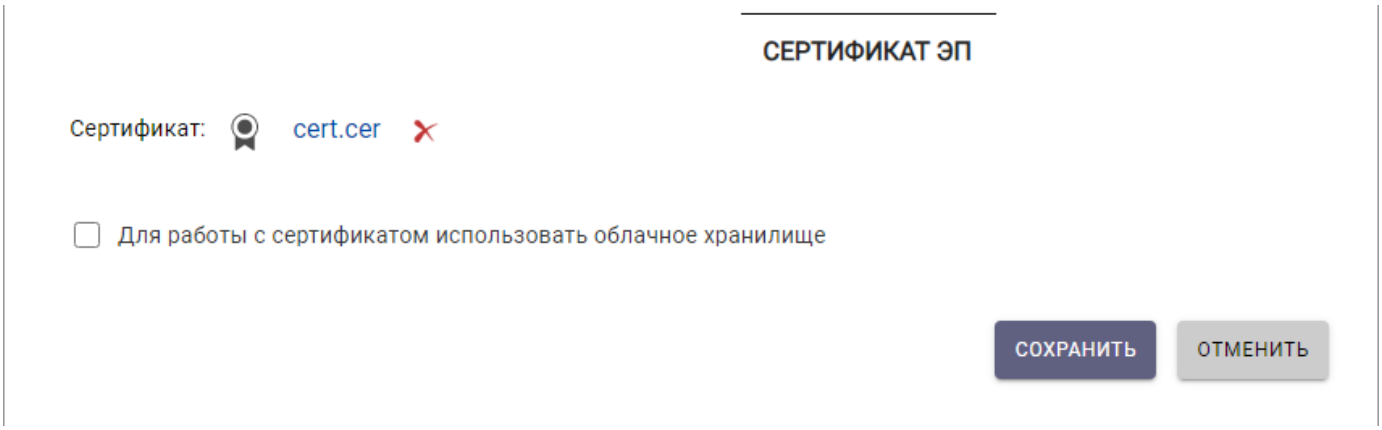


Рисунок 350. Вкладка "Сертификат ЭП" в режиме редактирования

На вкладке *Сертификат ЭП* можно указать сертификат, который будет использоваться сотрудником при выполнении заданий на подписание.

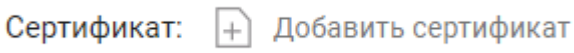


Рисунок 351. Добавить сертификат

Чтобы добавить сертификат, откройте вкладку, нажмите мышкой в поле *Сертификат* и выберите файл из файловой системы. Поддерживаются файлы сертификатов в следующих форматах: `.pem`, `.crt`, `.cer`, `.key`, `.der`, `.pkcs`.

Загрузка сертификатов в формате `.pfx` не поддерживается.

Установите флаг **Для работы с сертификатом использовать облачное хранилище**, чтобы загрузить сертификат в облачное хранилище.

Поля вкладки "Безопасность"

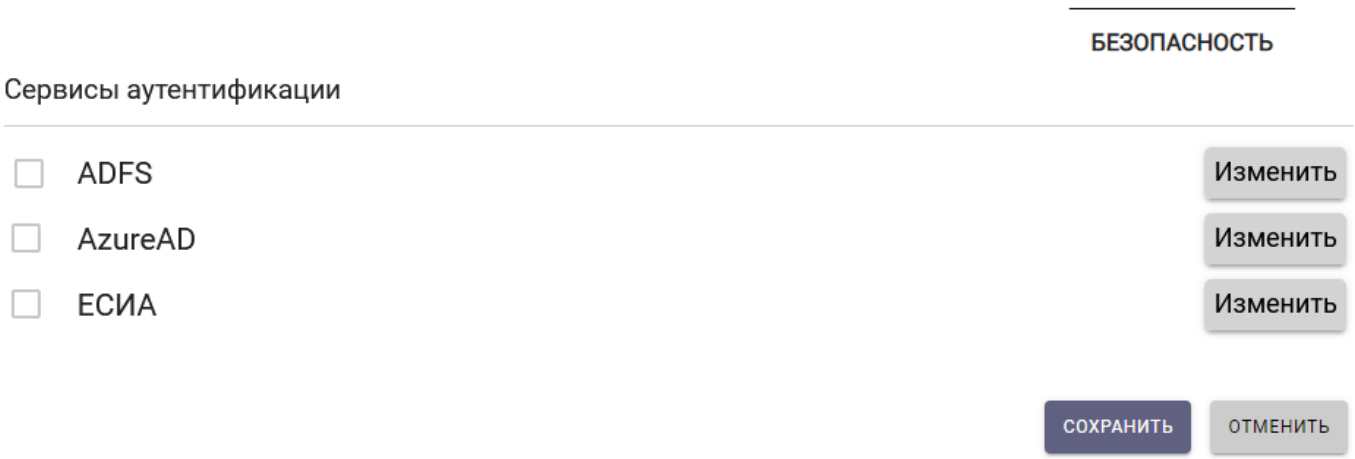


Рисунок 352. Вкладка "Безопасность" в режиме редактирования

- На вкладке *Безопасность* отображаются дополнительные сервисы авторизации. См. подробнее "Настройка способов входа".
- Для использования дополнительных сервисов авторизации требуется приобрести дополнительную лицензию, см. "Требования к лицензии".
- При отсутствии лицензии на использование сервиса авторизации, строка способа аутентификации недоступна для активации.
- Возможность настройки обеспечивается элементом управления "Способы аутентификации".
- На вкладке представлен список доступных методов авторизации. Список отображается в порядке, получаемом из API.

При нажатии на кнопку **Изменить** появится окно, в котором администратор может указать имя пользователя для данного сервиса.

Администратор может отключить определённый способ авторизации при помощи флага слева от сервиса. Если флаг снят — сервис недоступен. Флаг выставляется автоматически после нажатия на кнопку **Изменить**, когда заполнены данные для входа. Снять и поставить флаг можно, только когда указан логин.

Изменение данных для входа в сервис

При нажатии на кнопку **Изменить** открывается окно для указания имени пользователя и пароля для конкретного сервиса.

Некоторые сервисы позволяют отправить запрос на проверку пользователя. В таких случаях доступна кнопка **Проверка**.

- Если пользователь найден и проверка прошла успешно, справа появляется подтверждение.
- Если имя уже используется, сообщение об этом появится при нажатии кнопки **Ок**. Если имя ещё не использовалось, окно закрывается.
- Для сервиса авторизации ADFS потребуется указать пользователя в формате UPN: `user@domain.com`.

Статусы сотрудника

Статусы сотрудников позволяют задать текущий статус активности сотрудника без потери данных. Например, если установить для сотрудника статус *Уволен*, сотрудник окажется неактивен. При этом рабочие данные, связанные с

сотрудником, не пропадут из карточек.

Любой статус подсвечивается серым цветом, кроме статуса *Активен*. Статус *Активен* подсвечивается зелёным.

Для отслеживания и изменения статуса на время периода неактивности и возврата к активному статусу по окончанию данного периода используется служебный бизнес-процесс.

Статус	**Описание**
Активен	Сотрудник активен и может выполнять свои рабочие обязанности. Статус можно изменить из *Справочника сотрудников* и из профиля сотрудника в Web-клиенте.
Болен	Сотрудник отсутствует по состоянию здоровья. Статус можно изменить из *Справочника сотрудников* и из профиля сотрудника в Web-клиенте.
В отпуске	Сотрудник находится в отпуске. Статус можно изменить из *Справочника сотрудников* и из профиля сотрудника в Web-клиенте.
В командировке	Сотрудник находится в командировке. Статус можно изменить из *Справочника сотрудников* и из профиля сотрудника в Web-клиенте.
Отсутствует	Сотрудник отсутствует по каким-либо другим причинам. Статус можно изменить из *Справочника сотрудников* и из профиля сотрудника в Web-клиенте.
Уволен	Сотрудник уволен. Статус доступен для смены только из *Справочника сотрудников*.

Статус	Описание
Переведен	Сотрудник переведен в другое подразделение. Статус доступен для смены только из *Справочника сотрудников*.
Уволен без возможности восстановления	Запрет любых изменений сведений о сотруднике, включая повторное изменение состояния. Работа с системой для сотрудников в данном состоянии запрещена. Статус доступен для смены только из *Справочника сотрудников*.

Период недоступности сотрудника и замещение

Доступность дополняет выбранный статус сотрудника. Если сотрудник недоступен, в справочнике можно указать период недоступности сотрудника и его заместителей. Сотрудник также может настроить период недоступности самостоятельно из профиля

Чтобы настроить недоступность сотрудника:

1. Откройте карточку сотрудника в режиме редактирования или создания и перейдите на вкладку *Активность*.

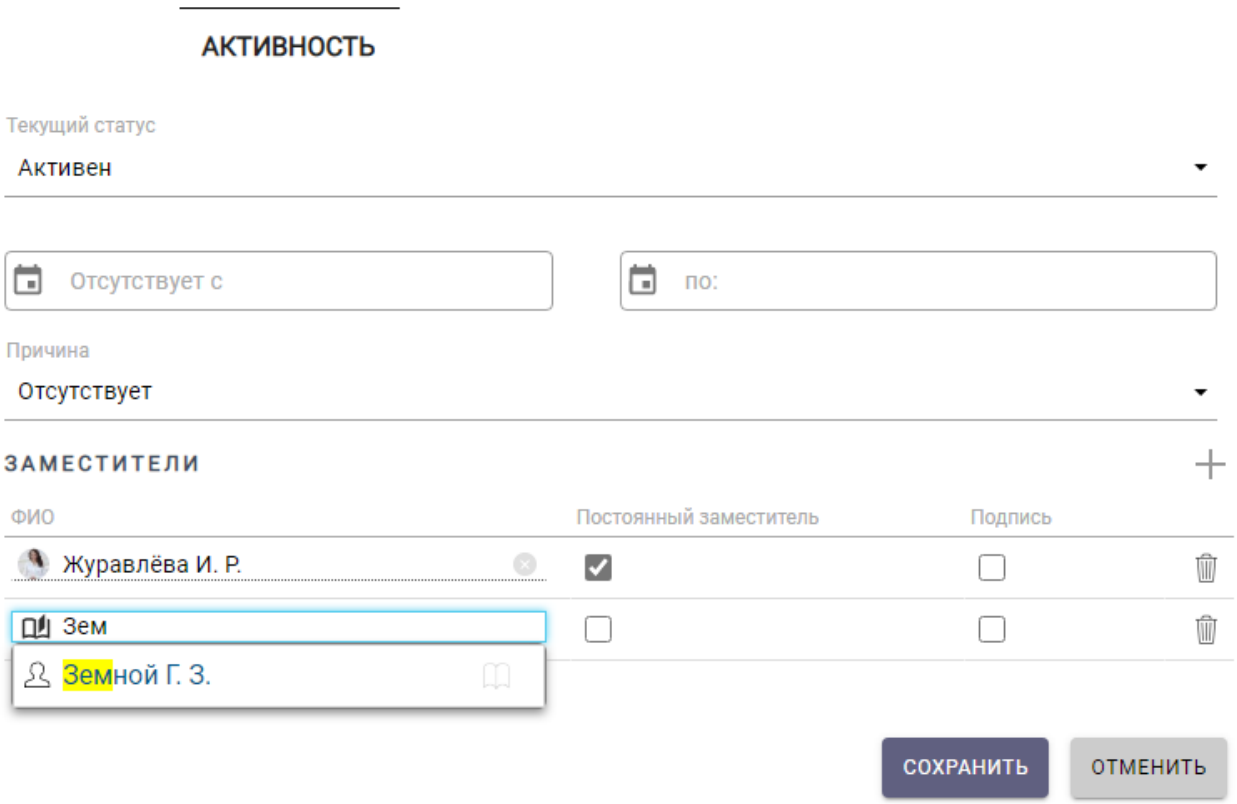


Рисунок 353. Вкладка "Активность" в режиме редактирования

2. Выберите текущий статус сотрудника из раскрывающегося списка.
3. В раскрывающемся списке *Причина* выберите причину неактивности сотрудника.

   Для выбора доступны значения:

   - ***Болен***.
   - ***В отпуске***.
   - ***В командировке***.
   - ***Отсутствует***.

   Изменить причину неактивности сотрудник может самостоятельно из профиля сотрудника в Web-клиенте.

4. Задайте даты начала и окончания периода отсутствия сотрудника в формате `с: ⋯ по: ⋯`.

   Указанная дата `с ⋯` должна быть раньше даты `по ⋯`. Обе даты

должны быть либо позже текущей даты, либо быть пустыми. Установить период неактивности в прошлом нельзя.

5. В области *Заместители* можно указать заместителя для сотрудника.

Работа с заместителями

Вкладка *Активность* содержит область *Заместители*. В зависимости от текущего режима открытия карточки сотрудника (просмотр, редактирование или создание), в данной области возможно добавить или посмотреть заместителей.

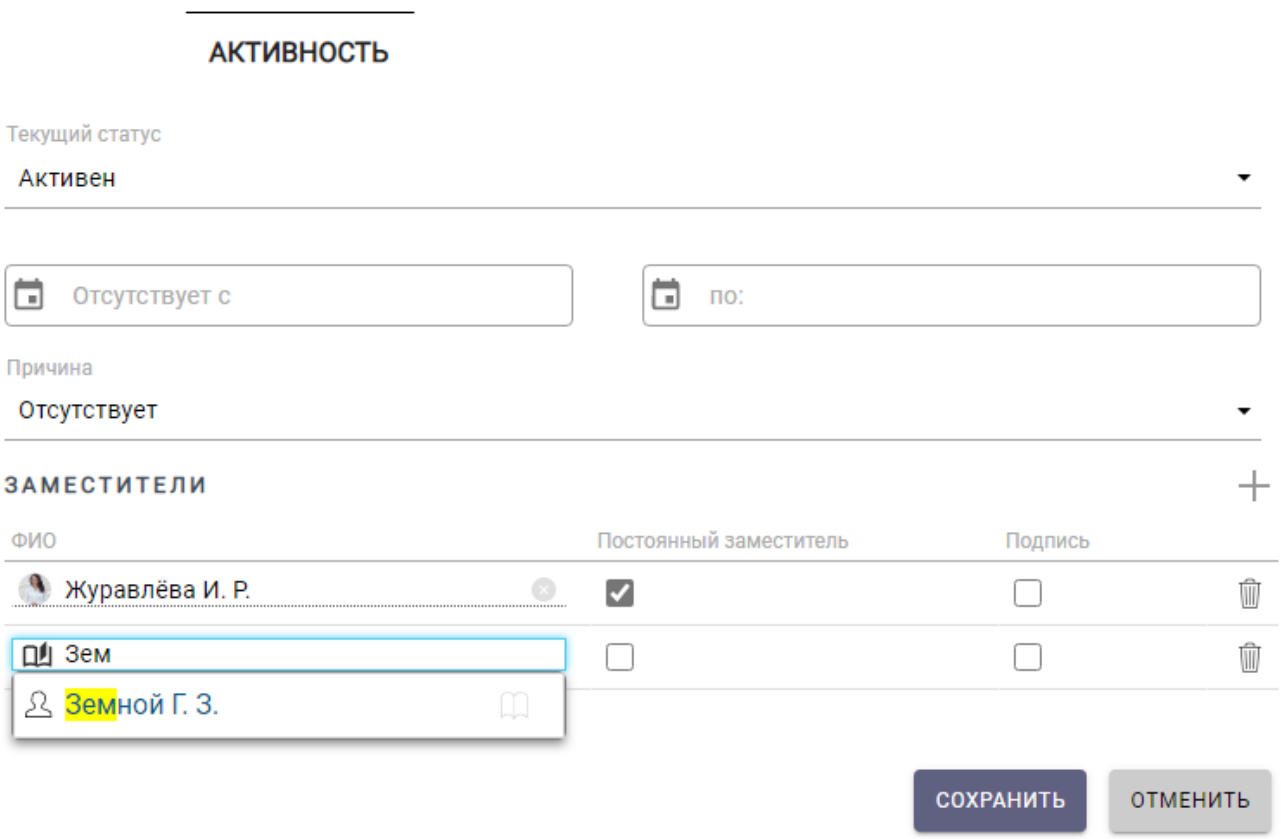


Рисунок 354. Вкладка "Активность" в режиме редактирования

Период и причина отсутствия, заместители сотрудника отображаются, только если задан период недоступности.

Добавление заместителей

Для сотрудника можно указать коллег по организации, которые, в случае его отсутствия, смогут выполнить за него задания или поставить вместо него электронную подпись под документом.

Чтобы добавить заместителей для сотрудника:

1. Откройте карточку сотрудника и перейдите на вкладку *Активность*.
2. В области *Заместители* нажмите на кнопку +, чтобы открыть строку для указания заместителя.
3. В строке выберите заместителя. Нажмите на кнопку 🕮, чтобы выбрать сотрудника из справочника, или начните вводить ФИО сотрудника, чтобы активировать быстрый поиск.

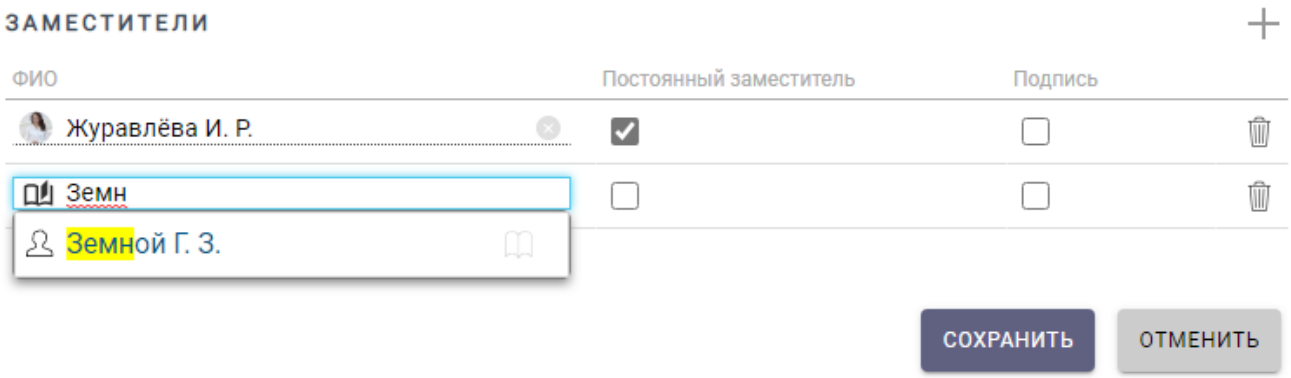


Рисунок 355. Добавление нового заместителя

4. Выберите сотрудника, который будет заместителем.

> В качестве заместителя можно добавить только сотрудников. Добавлять организации, подразделения, группы или должности в качестве заместителей не допускается.

5. Чтобы заместитель мог находить, просматривать и исполнять задания замещаемого, даже если основной исполнитель активен и сам его исполняет, установите флаг **Постоянный заместитель**.
6. Чтобы уполномочить заместителя ставить на документах ЭП от имени замещаемого им сотрудника, установите флаг **Подпись**.

> Вы можете добавить ещё нескольких заместителей, повторив шаги 2 — 6.

Просмотр заместителей сотрудников

Во вкладке *Активность* отображается список заместителей текущего сотрудника, а также их права в качестве заместителя.

Чтобы ознакомиться со списком заместителей сотрудника:

1. Откройте информацию о сотруднике и перейдите на вкладку *Активность*.

2. Просмотрите список заместителей в области *Заместители*.

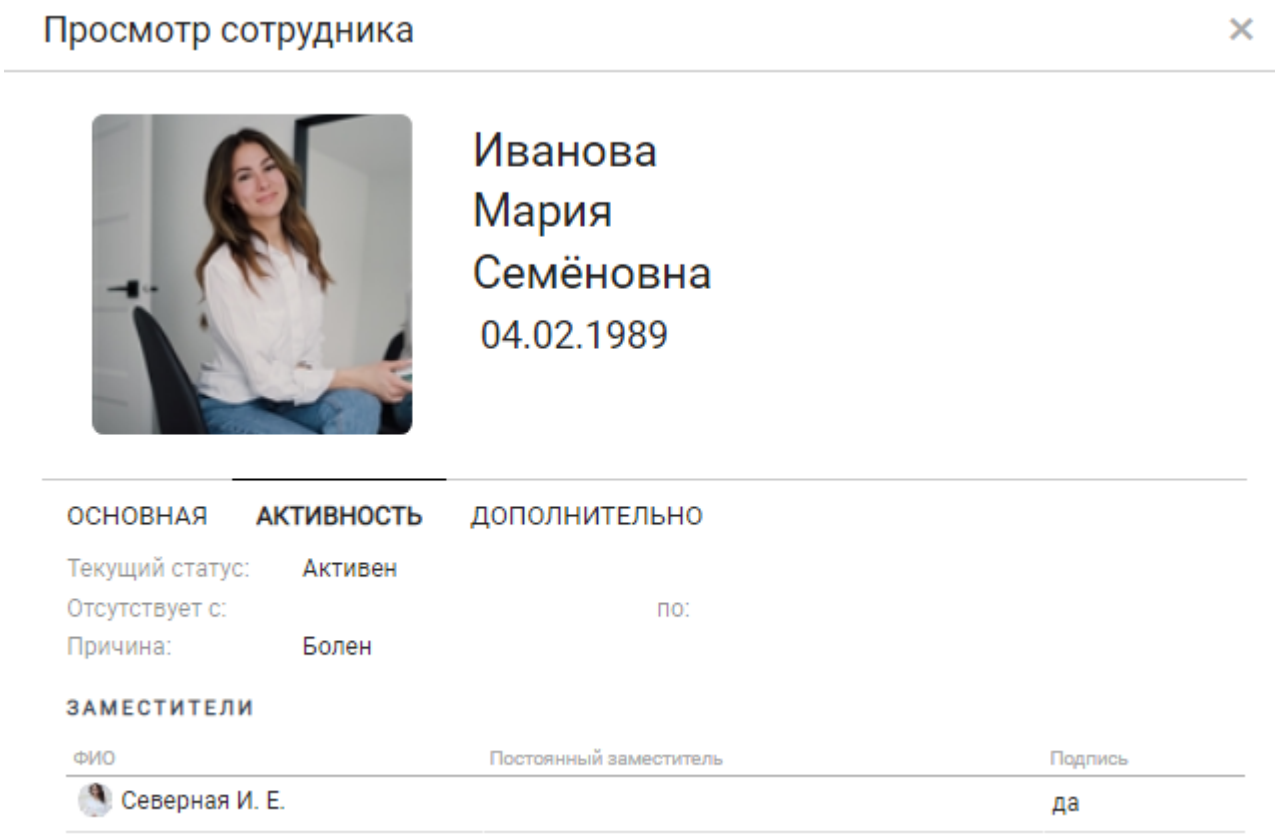


Рисунок 356. Просмотр заместителей сотрудника

Если период некативности указан, в информации о сотруднике отображаются период и причина отсутствия, заместители сотрудника отображаются. Если период недоступности не задан, отображается только статус.

Поиск по справочнику сотрудников

В справочнике реализована функция поиска. Поиск активируется нажатием на в верхней части пользовательского интерфейса Web-клиента. Результата поиска можно показать или скрыть нажатием на кнопку . Если список найденного слишком большой, можно развернуть и свернуть результаты поиска кнопками .

Поиск осуществляется по всем узлам справочника. В выпадающем списке слева от поисковой строки можно выбрать область поиска только по *Организациям и Подразделениям*, *Сотрудникам*, *Группам* и *Должностям*:

- Поиск по *Организациям и Подразделениям* осуществляется по полям

Наименование, *Полное наименование*, *ИНН*.

- Поиск по сотрудникам осуществляется по полям *Фамилия*, *Имя*, *Отчество*.
- Поиск по *Группам* и *Должностям* осуществляется по полю *Название* группы или должности.

В результатах поиска отображаются все найденные результаты, а также общее количество найденных результатов. Цветом выделяются совпадения с поисковым запросом. Нажатие на результат поиска переводит на соответствующий узел справочника.

Если поиск осуществлялся по *Организациям и Подразделениям* или *Группам*, структура справочника раскрывается до выбранного раздела и фокус смещается на найденный узел.

Если поиск осуществлялся по сотрудникам, то структура справочника раскрывается до родительского узла, и в правой области выбирается найденный сотрудник.

Настройки безопасности справочника сотрудников

В зависимости от текущих прав доступа пользователя, справочник:

- Откроется в режиме просмотра, если пользователю запрещено редактирование справочника.
- Откроется в режиме редактирования, если пользователю разрешено редактирование справочника.
- Не откроется, и появится предупреждение об отсутствии прав доступа, если пользователь не имеет прав на просмотр справочника.

Настройки дискреционной безопасности для всего справочника сотрудников

Администратор может задать права доступа для всего справочника при помощи элемента управления , расположенного над правой областью.

Если администратор Web-клиента установил ограничение для работы со справочником сотрудников, возможности работы с узлами справочника, включая *Организации*, *Подразделения*, *Группы*, а также работа с *Сотрудниками* и *Должностями*, могут быть ограничены. Данное ограничение также распространяется на возможность поиска по справочнику.

Чтобы настроить дискреционную безопасность для всего справочника:

1. Перейдите в справочник по ссылке из основного меню.
2. Нажмите .
3. В появившемся окне задайте параметры доступа для пользователей согласно инструкции, данной в пункте Настройка безопасности объектов.
4. Подтвердите настройки, нажав **Применить** или **ОК**.



Настройки дискреционной безопасности наследуются дочерними элементами. Это означает, что, если наследование не отключено, настройки дискреционной безопасности для подразделений и организаций будут унаследованы от родительских узлов.

При одновременном редактировании прав доступа к справочнику с разных окон или вкладок браузера изменения сохранятся только из последнего закрытого окна.

Настройки дискреционной безопасности для узлов справочника сотрудников

Администратор имеет возможность задать права доступа к следующим узлам справочника: *Организации*, *Подразделения*, *Группы*.

Чтобы задать права доступа к узлу справочника:

1. Выберите необходимый узел.
2. Вызовите контекстное меню на выбранном узле.
3. В контекстном меню выберите пункт *Права доступа*.
4. В появившемся окне задайте параметры доступа для пользователей согласно инструкции, данной в пункте Настройка безопасности объектов.
5. Подтвердите настройки, нажав **Применить** или **ОК**.

Копирование узлов справочника сотрудников

В справочнике реализована возможность *Копировать* и *Вырезать* узлы и сотрудников. Операции копирования или вырезания могут быть вызваны из контекстного меню выбранного узла.

Вокруг скопированных или вырезанных узлов и сотрудников появляется пунктирная рамка. Вырезанный узел дополнительно обозначается серым цветом. Цвет скопированных узлов и сотрудников не меняется.

Копировать

Операция копирует только иерархию узла, но не копирует сотрудников.

Чтобы переместить узлы вместе с сотрудниками, воспользуйтесь операцией *Вырезать*.

Вырезать

Операция перемещает узел со всеми дочерними узлами, включая сотрудников. При выполнении операции исходный узел удаляется в старом месте и перемещается в новое. Вырезать можно также отдельных сотрудников.

Вставить

Операция завершает копирование или вырезание. Операция недоступна, пока раздел не был скопирован или вырезан.

Операция **Вставить** отправляет запрос на сервер. Это значит, что узел будет скопирован или вырезан только после выбора *Вставить*.

Чтобы скопировать или переместить узел:

1. Выберите узел-источник, который желаете скопировать или вырезать.
2. Вызовите контекстное меню узла-источника.
3. Из контекстного меню узла выберите *Копировать* или *Вырезать*.



Скопированный или вырезанный узел или сотрудник помещаются в буфер обмена. После этого справа от узла или сотрудника появляется кнопка **Отменить**. Вместо пунктов *Копировать* или *Вырезать* контекстного меню появляется пункт *Отменить*. Нажатие на эту кнопку или пункт в контекстном меню очищает буфер обмена.

4. Вызовите контекстное меню на целевом узле, в который необходимо добавить или переместить текущий.
5. В контекстном меню выберите пункт *Вставить*.



Перемещать и копировать организации и подразделения можно только в организации или подразделения, группы — только в группы, ограничений на перемещение сотрудников нет.

Незавершённые операции будут отменены в следующих случаях:

- При повторном открытии справочника или обновлении страницы.
- При переходе на новую вкладку или при закрытии текущей.

- При нажатии кнопки `Esc` на клавиатуре.
- При выборе пункта *Отменить* из контекстного меню.
- При нажатии кнопки **Отменить** справа от узла.
- При копировании или вырезании другого узла.

При выполнении операций проверяются права доступа на редактирование узлов справочника.

Чтобы вырезать узлы и сотрудников, пользователь должен иметь права на редактирование исходного и родительского узлов. Чтобы вставить узлы и сотрудников, пользователь должен иметь права на редактирование целевого узла.

После выполнения операций, настройки прав доступа исходного узла будут заменены настройками, унаследованными от нового родительского узла.

Справочник полномочий для доверенностей

Справочник предназначен для хранения кодов полномочий для машиночитаемых доверенностей.

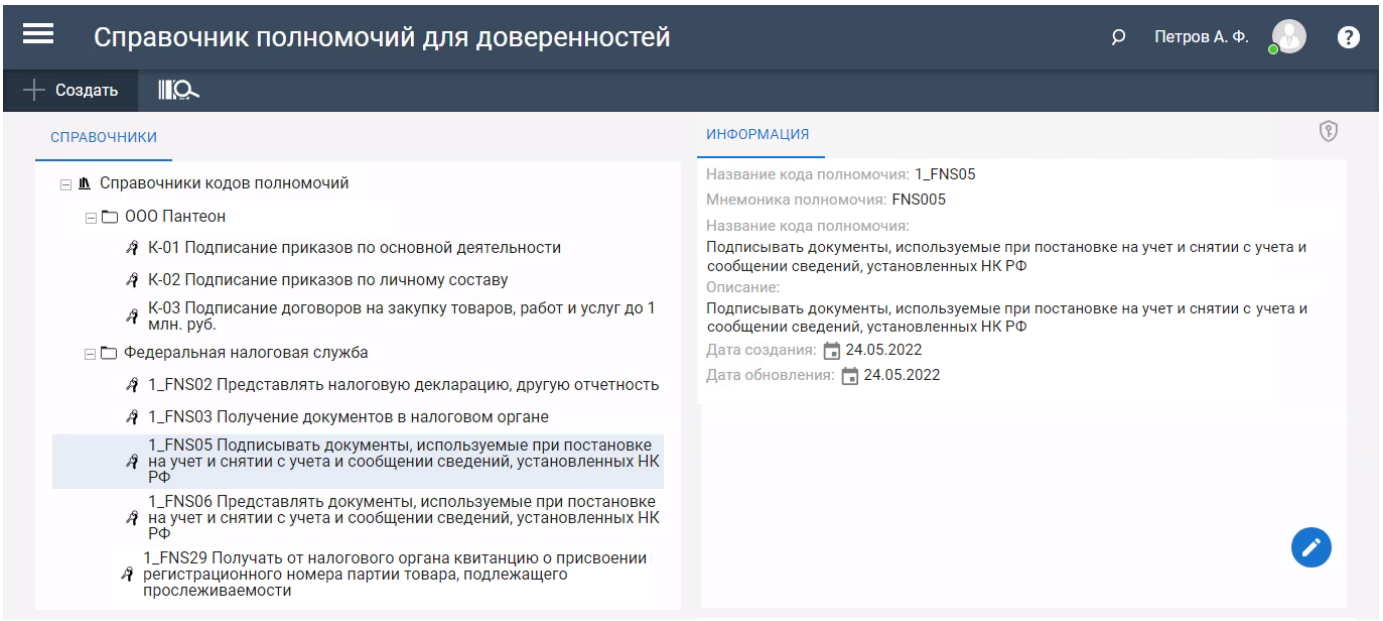


Рисунок 357. Справочник полномочий для доверенностей

Справочник доступен по адресу: `http://адрес-сервера-Web-клиента:ПОРТ/#/DirectoryView/30EB75E2-16F8-4150-A2D4-4A82A38A1849`, ссылка на справочник полномочий также добавлена в разметку *Основного меню* в базовом решении.

В левой области справочника отображается вкладка *Справочники*. Это список справочников кодов полномочий, отсортированный в алфавитном порядке. Коды полномочий отображаются в формате Код полномочий Полномочия. При необходимости полномочия можно найти при помощи поиска по справочнику.

Структура справочника

Справочник представляет собой двухуровневую иерархическую структуру:

- На первом уровне справочника представлен список справочников полномочий.
- В каждом справочнике представлены коды полномочий.

Отредактировать справочник или код полномочий можно, выбрав в контекстном меню команды *Изменить код полномочий* или *Изменить справочник*, а также при помощи кнопки .

Добавление нового справочника

Добавить справочник кодов в список можно, выбрав в контекстном меню узла *Справочник кодов полномочий* команду *Создать справочник*, или выбрать корневой узел, нажать кнопку , выбрать **Добавить справочник**.

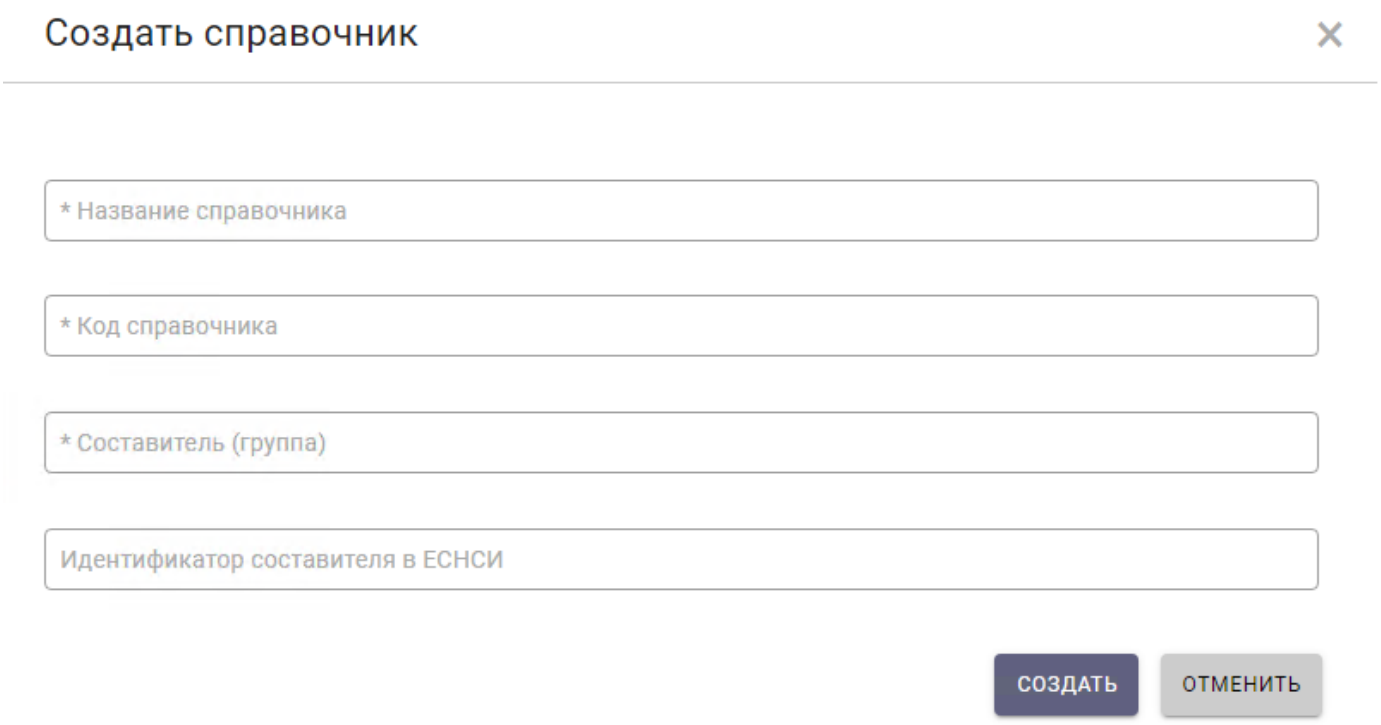


Рисунок 358. Создание нового справочника полномочий

Появится всплывающее окно с полями для добавления нового справочника:

1. *Название справочника* — удобочитаемое название справочника. Поле обязательно для заполнения.
2. *Код справочника* — уникальный код справочника полномочий. Поле обязательно для заполнения.
3. *Составитель (группа)* — пользователь или группа составителей справочника. Поле обязательно для заполнения.
4. *Идентификатор составителя в ЕСНСИ* — необходим при <импорте полномочий из ЕСНСИ. Идентификатор в справочнике сравнивается с идентификатором в XML-файле ЕСНСИ. В зависимости от результата сравнения полномочия будут импортированы или помечены неактивными. См. подробнее ниже, в разделе Импорт справочника из ЕСНСИ.

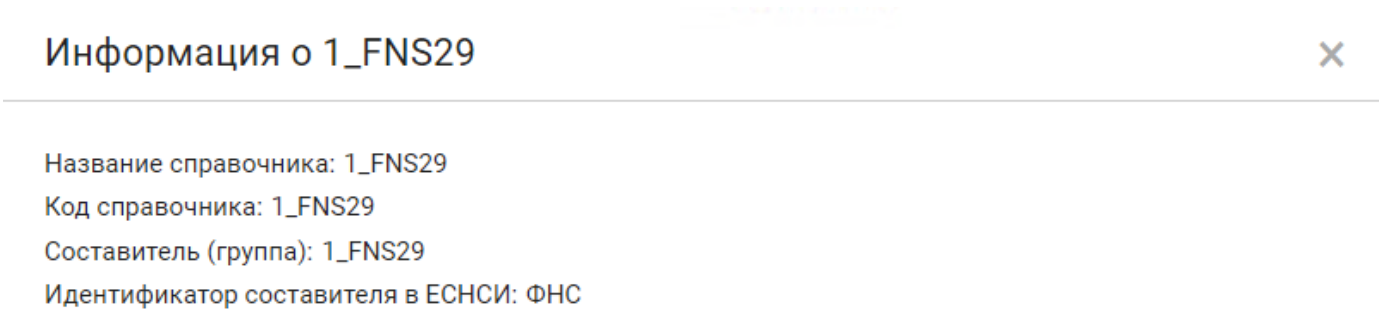


Рисунок 359. Информация о справочнике кодов полномочий

Информацию о справочнике кодов полномочий можно изменить, выбрав код в списке слева и нажав кнопку .

Импорт справочника из ЕСНСИ

Система позволяет импортировать полномочия из ЕСНСИ через загрузку данных из xml-файла. Поддерживается импорт только из справочника полномочий ЕСНСИ, доступного по адресу[3] https://esnsi.gosuslugi.ru/classifiers/9532/data?p=1.

1. Перед импортом необходимо создать справочники с NSI_ID органов, чьи коды полномочий нужны. Для каждого справочника требуется свой код.
2. Импортировать полномочия в список можно, выбрав в контекстном меню узла *Справочник кодов полномочий* команду *Импортировать полномочия*, или выбрать корневой узел, нажать кнопку , выбрать **Импортировать**

полномочия.

3. При нажатии на кнопку **Импортировать полномочия** открывается диалоговое окно для выбора xml-файла с полномочиями из файловой системы.
4. После выбора XML-файла будет выполнен импорт полномочий.

   Во время импорта будет выполнена проверка существования полномочий по Идентификатору составителя в ЕСНСИ.

   - Если в XML-файле нет кода полномочий, соответствующего Идентификатору составителя в ЕСНСИ, группа в справочнике не обрабатывается.
   - Если в справочнике будут обнаружены дублирующие коды полномочий, которых нет в xml-файле, код полномочий будет помечен флагом `Отменён`.
   - По кодам из XML-файла, которых нет в локальном справочнике будут созданы *Коды полномочий* каждой имеющейся группы.
   - Если в XML-файле будут обнаружены коды полномочий, которых нет в локальном справочнике, в справочник будут импортированы соответствующие полномочия.
5. По завершении импорта появится информационное окно с результатом:
   - При успешном импорте отобразится список групп, в которые были загружены коды.
   - Если соответствий полномочиям из XML не найдено, будет отображено сообщение: `Ни одна из записей не была импортирована`.
   - Если в XML будут обнаружены только уже имеющиеся в локальном справочнике полномочия, будет отображено сообщение: `Все доступные полномочия уже имеются в справочнике`.
6. После обновления страницы в справочнике будут отображены созданные новые записи с заполненными полями значений кодов полномочий, мнемоники, названия, описания, датами создания и обновления.

Добавление нового кода полномочий

Добавить новый код полномочий можно, выбрав в контекстном меню любого справочника команду *Создать код*, или выбрать справочник, нажать кнопку .

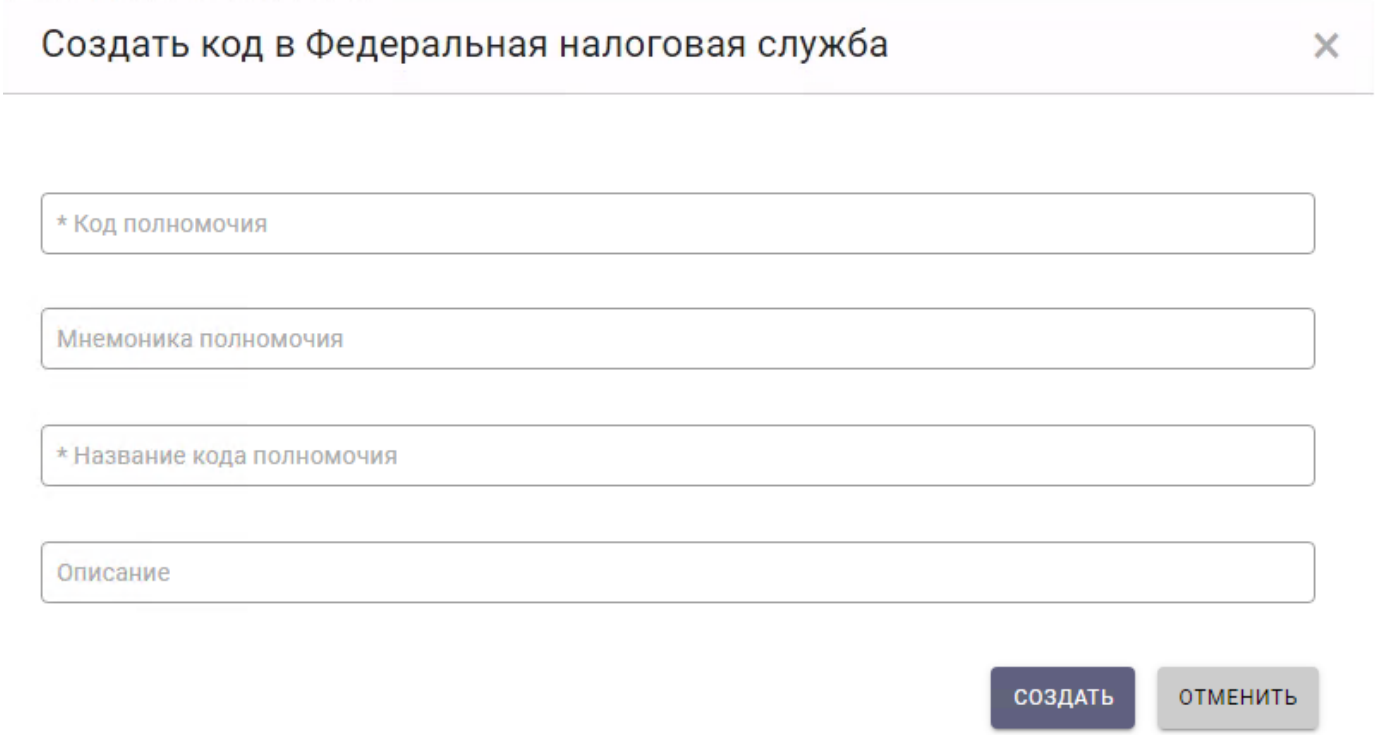


Рисунок 360. Создание нового кода полномочий

При добавлении кода появится окно с доступными для заполнения полями:

1. *Код полномочий* — уникальный код полномочий. Поле обязательно для заполнения.
2. *Мнемоника полномочия* — мнемоника полномочия в реестре полномочий. Необязательное поле, может совпадать со значением *Кода полномочий*. Значения *Кода* и *Мнемоники* полномочий должны быть уникальными.
3. *Название кода полномочий* — название полномочия. Поле обязательно для заполнения.
4. *Описание* — подробное описание кода полномочий.

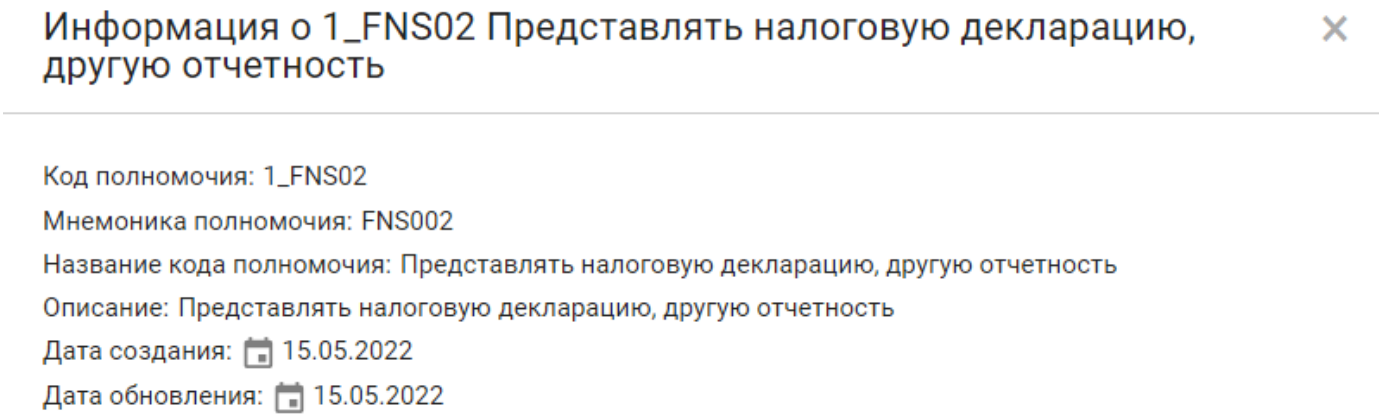


Рисунок 361. Информация о коде полномочий

Когда в левой части выбран код полномочий, в правой части отображается информация о коде полномочий, включая два дополнительных поля, которые заполняются системой автоматически:

5. *Дата создания* — дата создания кода.
6. *Дата обновления* — дата обновления кода.

Информацию о коде полномочий можно изменить, выбрав код в списке слева и нажав кнопку .

Коды полномочий не могут быть удалены, если они были использованы в карточке СКД через вызов API создания и передоверия МЧД.
При попытке удалить такие коды полномочий будет выдано предупреждение `Невозможно удалить полномочия, т.к. данный код используется в сформированных МЧД`.

Если хотя бы один код в группе использовался в карточке СКД, при попытке удалить группу будет выдано предупреждение `Невозможно удалить справочник, т.к. он содержит полномочия, которые используются в сформированных МЧД`.

Настройки безопасности справочника

В зависимости от текущих прав доступа пользователя, справочник:

- Откроется в режиме просмотра, если пользователю запрещено редактирование справочника.
- Откроется в режиме редактирования, если пользователю разрешено редактирование справочника.

- Не откроется, и появится предупреждение об отсутствии прав доступа, если пользователь не имеет прав на просмотр справочника.

Администратор может задать права доступа для всего справочника при помощи элемента управления , расположенного над правой областью, или из контекстного меню узла.

Настройка безопасности справочника полномочий выполняется по аналогии со справочником сотрудников.

Выбор полномочий из справочника

Для выбора полномочий из справочника предназначен ЭУ "Полномочия". ЭУ должен быть добавлен в разметки карточек.

Приложения

Работа в Web-клиенте

Приложения

Дополнительные разделы документации, содержащие информацию, полезную при работе с Web-клиентом Docsvision.

- Приложение А. Основные сценарии работы с документами
- Приложение Б. Описание элементов управления разметок карточек
- Больше информации в навигационном меню слева…

Приложение А. Основные сценарии работы с документами

Сценарий работы с входящей корреспонденцией

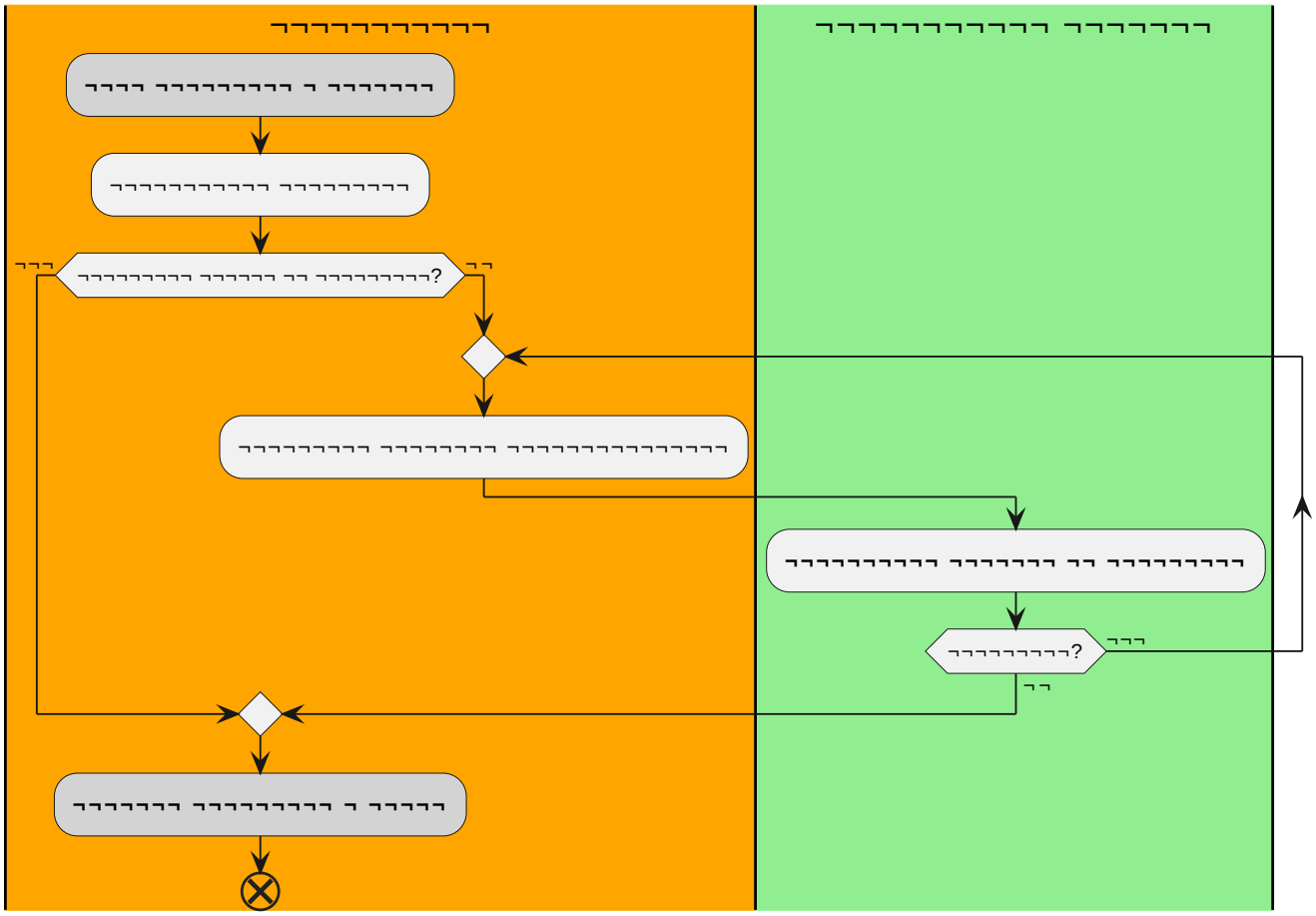


Рисунок 362. Обработка входящей корреспонденции

Сценарий работы с исходящей корреспонденцией

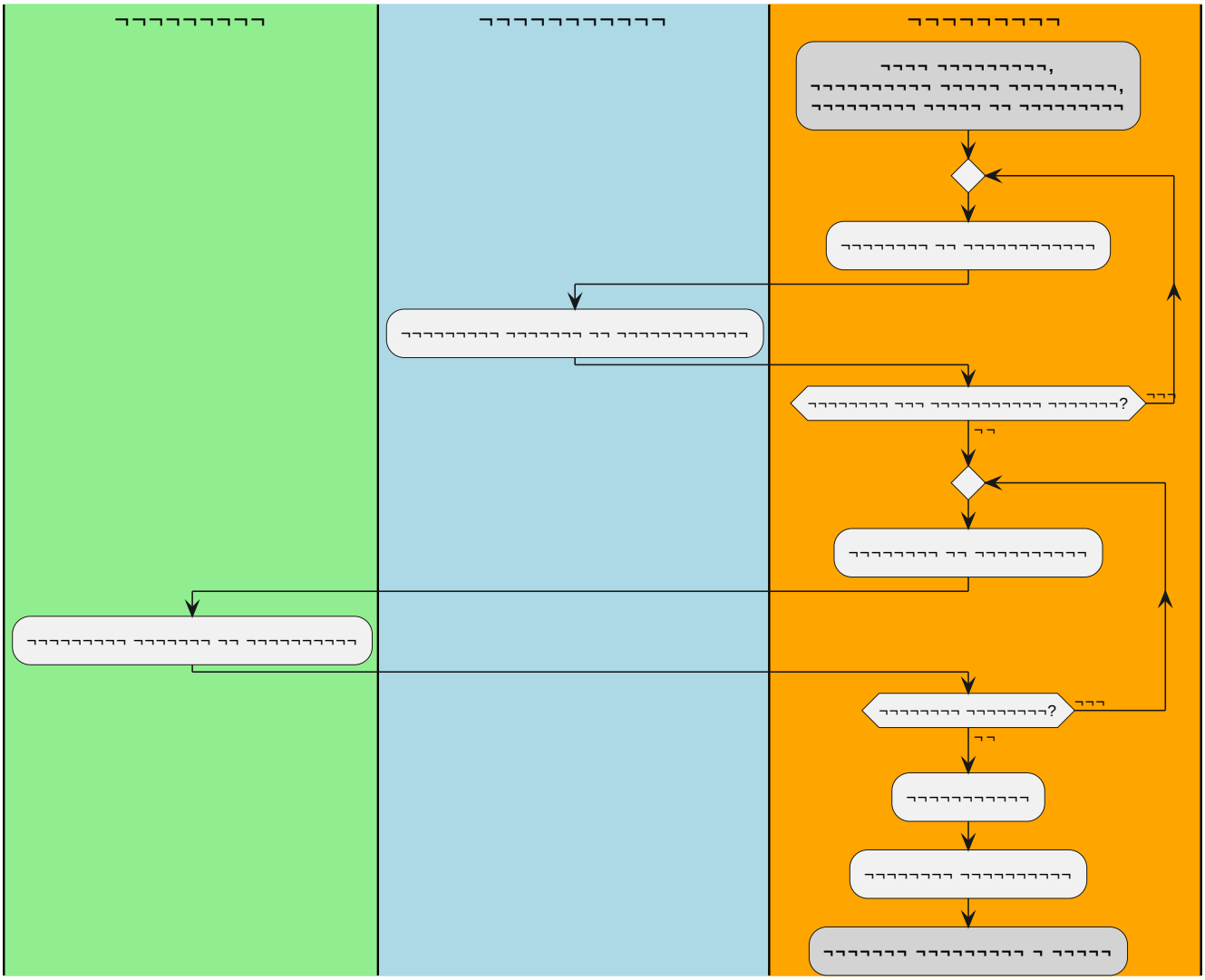


Рисунок 363. Работа с исходящей корреспонденцией

Сценарий работы с организационно-распорядительными документами

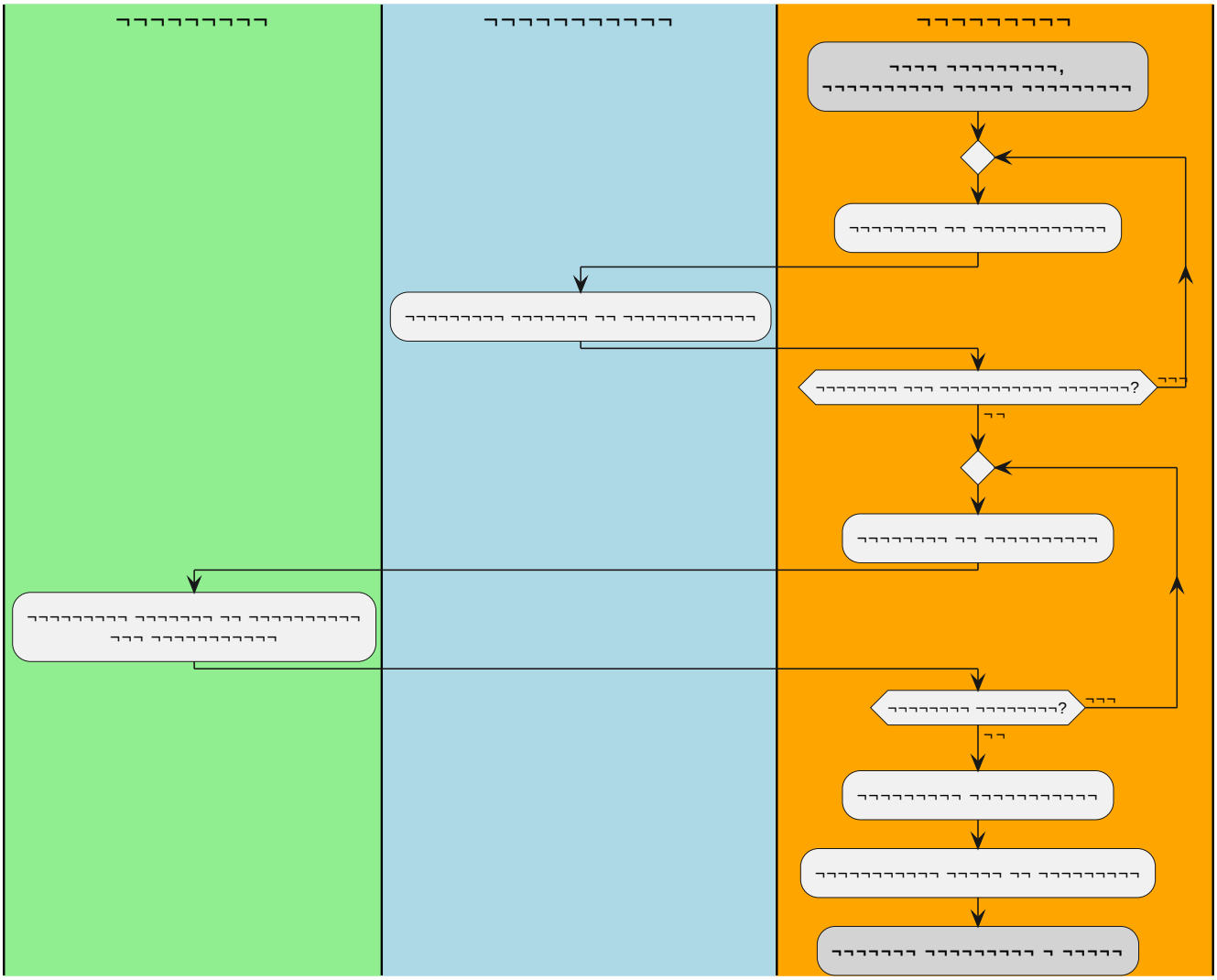


Рисунок 364. Работа с организационно-распорядительными документами

Сценарий работы со служебными записками

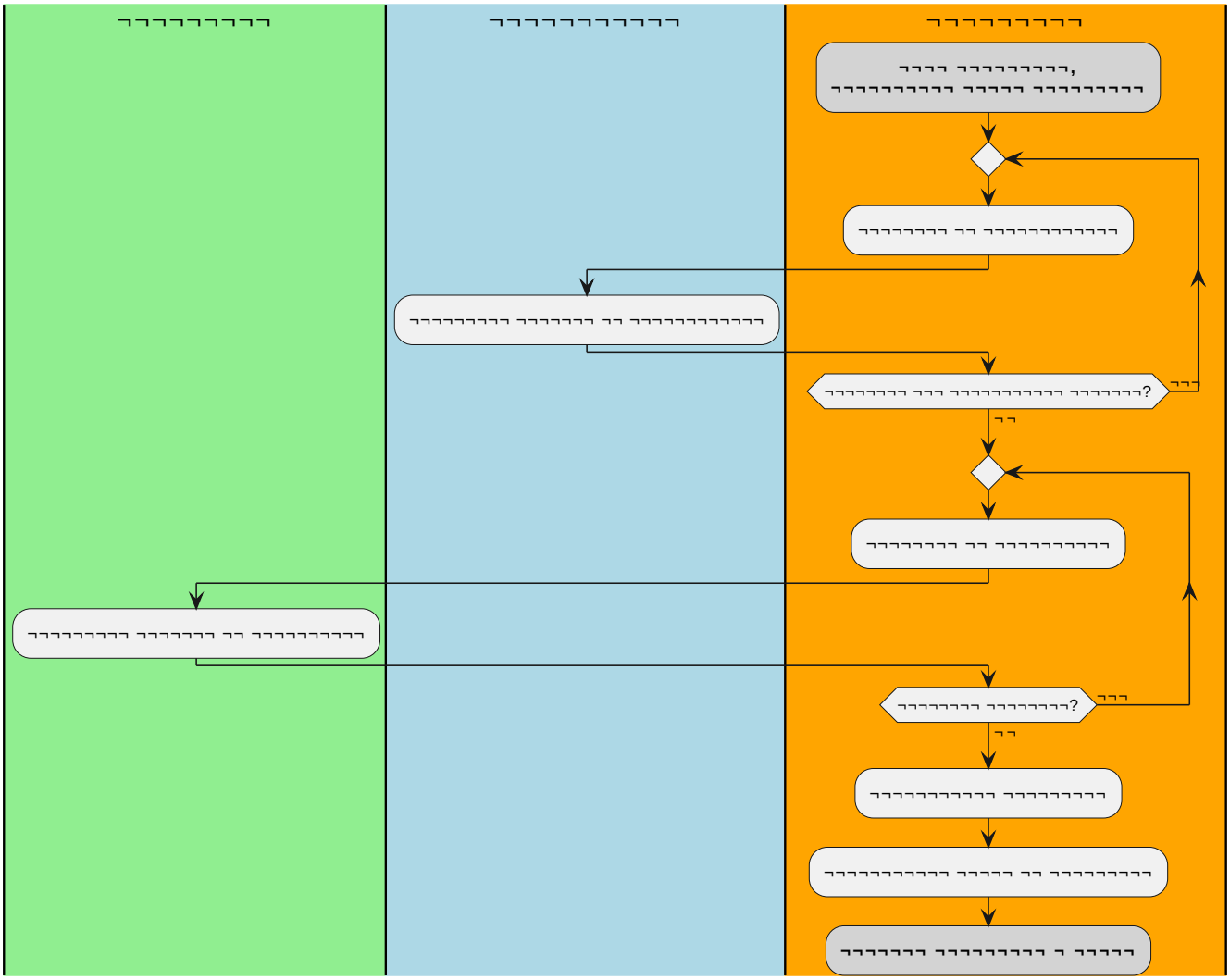


Рисунок 365. Работа со служебными записками

Сценарий работы с заявками

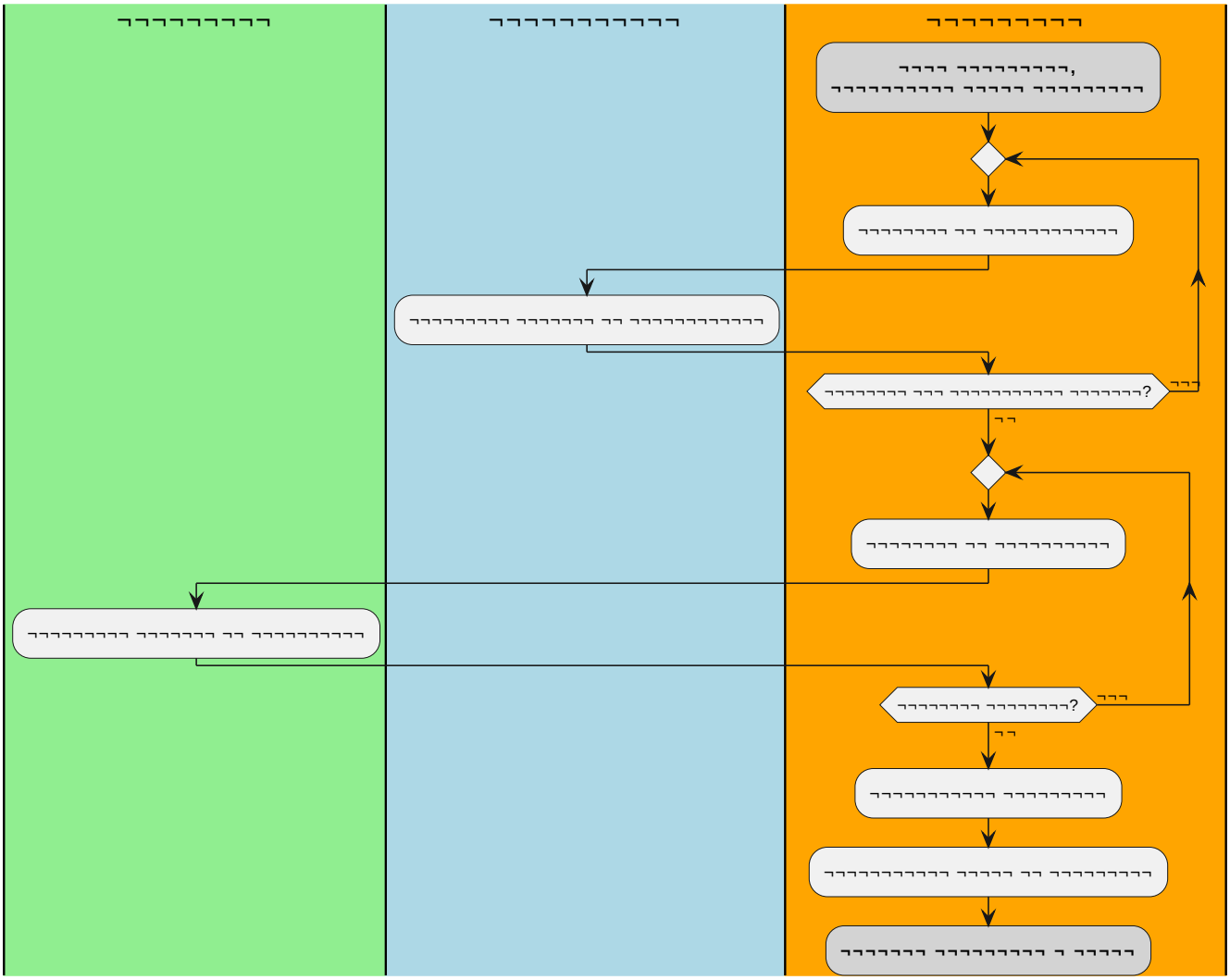


Рисунок 366. Работа с заявками

Сценарий обращения с рабочей документацией

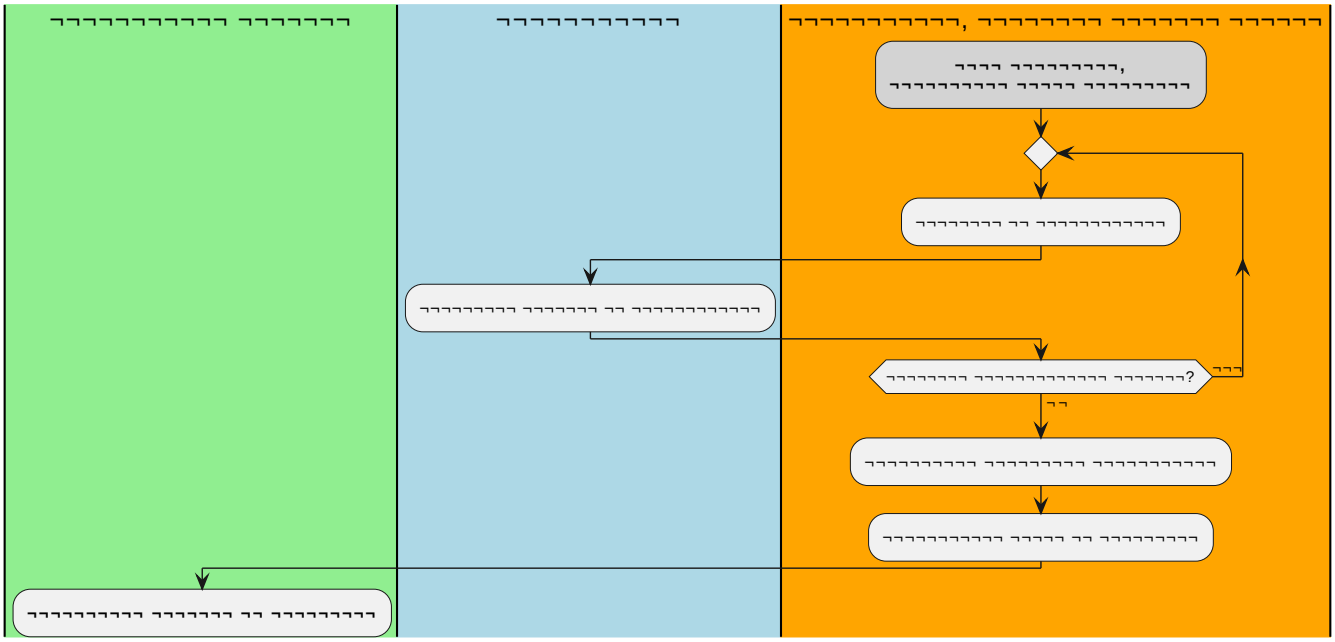


Рисунок 367. Обработка рабочего документа

Приложение Б. Особенности работы консолидации

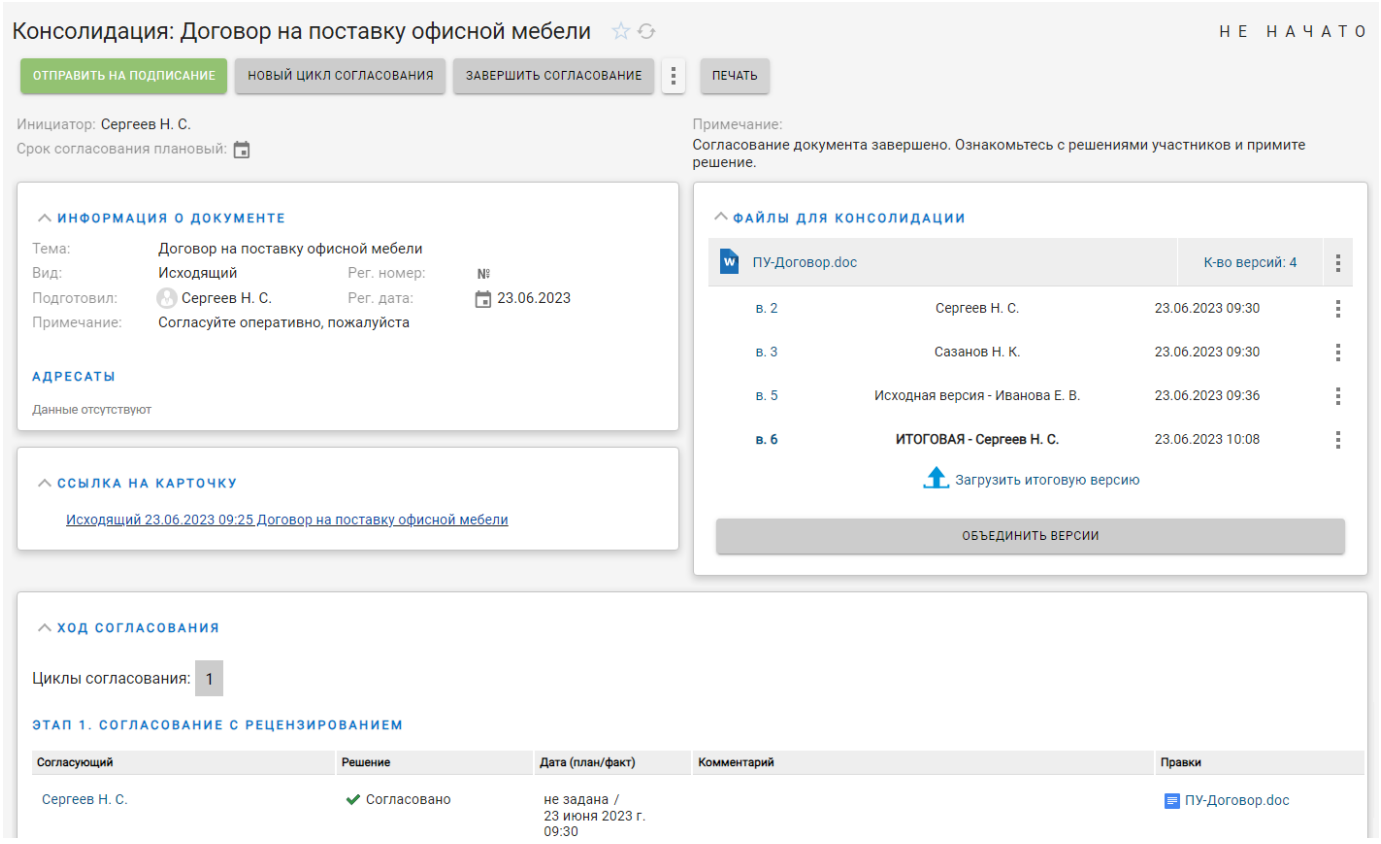


Рисунок 368. Вид задания на консолидацию

Консолидация — процесс формирования итоговой версии согласуемого документа из всех версий, добавленных согласующими. После консолидации итоговая версия может быть отправлена на окончательное подписание, или быть

использована в качестве стартовой в новом цикле согласования. Без выполнения консолидации или копирования какой-либо версии в итоговую согласующие или подписывающие в заданиях на согласование или на подписание всегда будут получать копию файла, отправленного при старте согласования.

Консолидация запускается из карточки задания на консолидацию при нажатии кнопки **Объединить версии**. До Web-клиент 18 (6.1) для консолидации использовался установленный на компьютере сотрудника Microsoft Word. В текущей версии для консолидации необходим Р7-Офис. Сервер документов, размещенный на серверной машине. Инструкция по установке и настройке Р7 Офис приведена в разделе администрирования модуля: "Настройка "Р7-Офис. Сервер документов"".

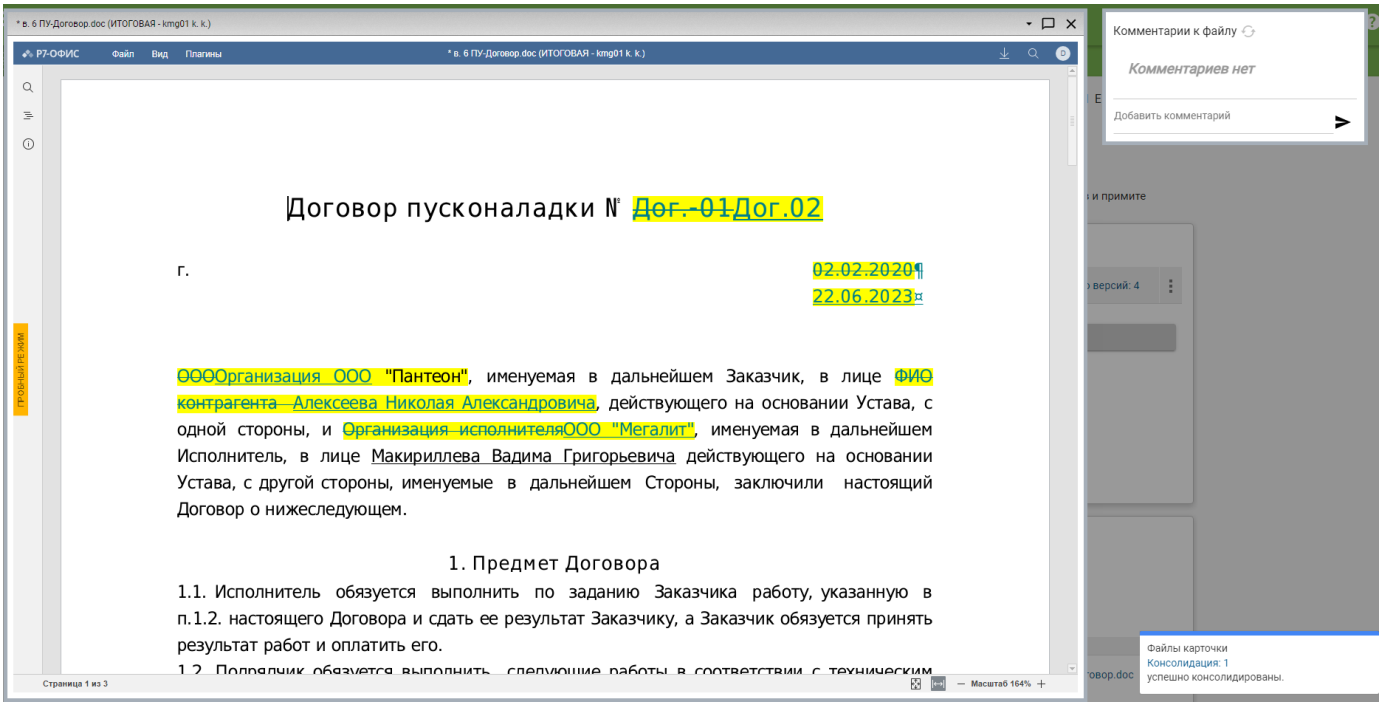


Рисунок 369. Серверная консолидация при помощи Р7

Консолидация при использовании Р7 отличается от консолидации при помощи Microsoft Word. Раньше Word сводил правки от всех согласующих в одну версию, можно было оставить одни правки и удалить другие. Например, при наличии трёх версий файла с текстом Версия 1, Версия 2,Версия 3, то результирующая версия выглядела как Версия 123

Р7 принимает последнюю добавленную правку, сохраняет историю изменений на панели рецензирования. Например, при тех же трёх исходных версиях, результирующая версия будет выглядеть как Версия 3, изменения всех версий будут отображаться на панели рецензирования.

В процессе консолидации через Р7-Офис участвуют все версии файла, включая итоговую, которая может быть загружена вручную. При повторной консолидации через Р7 вкладка *Совместная работа* недоступна, т.к. сервер документов сравнивает все версии и принимает изменения из итоговой версии документа, полученной ранее. Вкладка *Совместная работа* не отображается, т.к. все изменения с предыдущего согласования уже приняты.

Приложение Б. Описание элементов управления разметок карточек

Работа в Web-клиенте

Приложение Б. Описание элементов управления разметок карточек

Дополнительные разделы документации, содержащие информацию, полезную при работе с Web-клиентом Docsvision.

- Простые поля ввода
- Нумератор
- Файловая панель
- Больше информации в навигационном меню слева…

Общие

Работа в Web-клиенте

Общие

Дополнительные разделы документации, содержащие информацию, полезную при работе с Web-клиентом Docsvision.

- Простые поля ввода
- Дата и время
- Текст
- Больше информации в навигационном меню слева…

Простые поля ввода

Поля ввода предназначены для ввода текстовых или числовых значений. Поля для ввода текстовых и числовых данных имеют идентичный вид:



Рисунок 370. Пример элемента "Строка"

Дата и время

Если поле разметки предназначено для ввода даты и времени, оно будет иметь вид:



Рисунок 371. Поля для ввода даты и времени

Если поле настроено только для ввода даты, то область ввода/отображения времени будет скрыта.

Если нужно закрыть календарь до завершения ввода, щелкните по любой области окна Web-клиента вне календаря.

Перейти к редактированию нужного фрагмента даты/времени (например, года) можно щелкнув по данному фрагменту в верхней части окна календаря.

Ввод даты и времени

Последовательно введите цифры даты и времени в формате `ДД.ММ.ГГГ.ЧЧ.ММ` или ином, предусмотренном региональными настройками. Блок ввода времени будет недоступен, если указывать время не требуется.

Выбор даты и времени из календаря

Чтобы открыть календарь, нажмите кнопку .

Выбор даты осуществляется на странице .

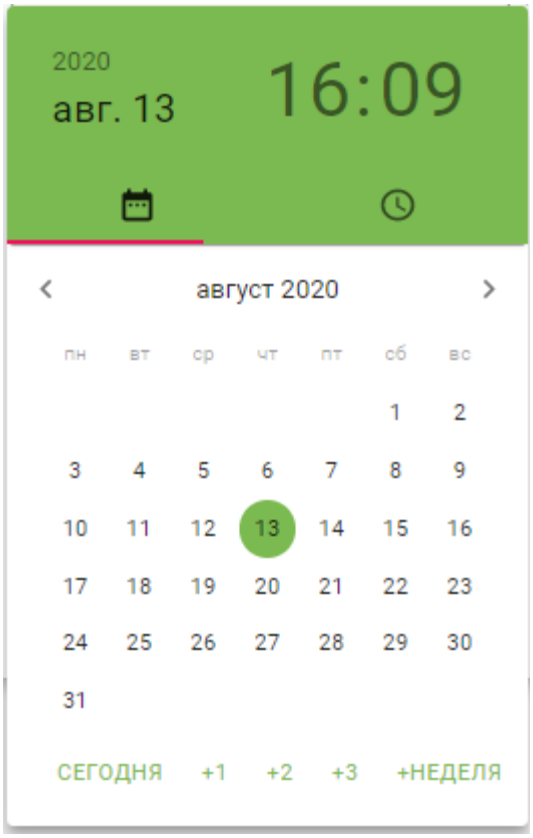


Рисунок 372. Календарь

Чтобы изменить год, нажмите на текущее значение. Будет открыт список лет.

Чтобы изменить месяц, используйте переключатели в строке с месяцем.

С помощью кнопок **+1**, **+2**, **+3**, **+Неделя** можно выбрать дату с заданным смещением относительно текущей даты и текущим временем.

Выбор времени осуществляется на странице .

Возможность выбирать время может быть отключена для элемента управления.

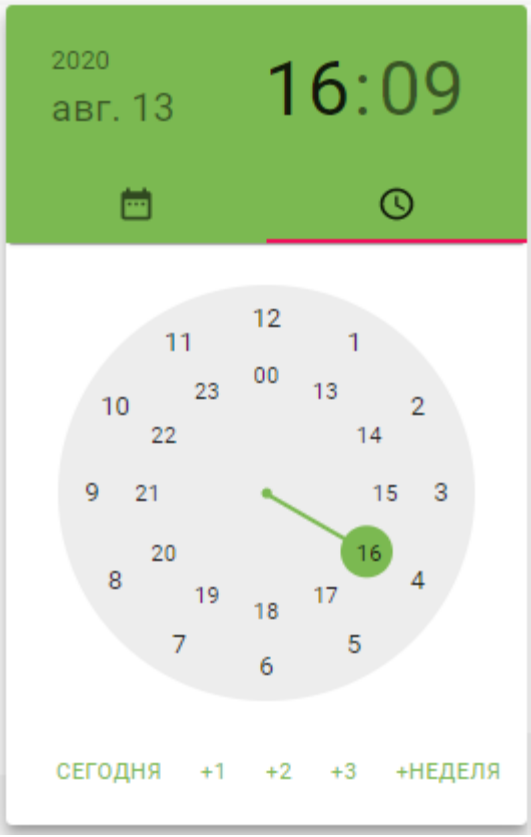


Рисунок 373. Выбор времени

Выберите часы, затем минуты.

Очистка значения

Нажмите кнопку ⊗, чтобы очистить значение элемента управления.

Проверка значения

Элемент управления автоматически проверяет корректность введённого значения. Если значение не является корректным будет показана ошибка:

- `Неверный формат даты (маска для ввода ⋯)`. Если введённое значение не соответствует указанному (в ошибке) формату;
- `Дата не должна быть позже ⋯` или `Дата не должна быть раньше ⋯`. Если введённое значение выходит за пределы разрешённого диапазона дат (определяется при настройке элемента).

Текст

Элемент управления `Текст` предназначен для работы с многострочным текстом с возможностью форматирования.

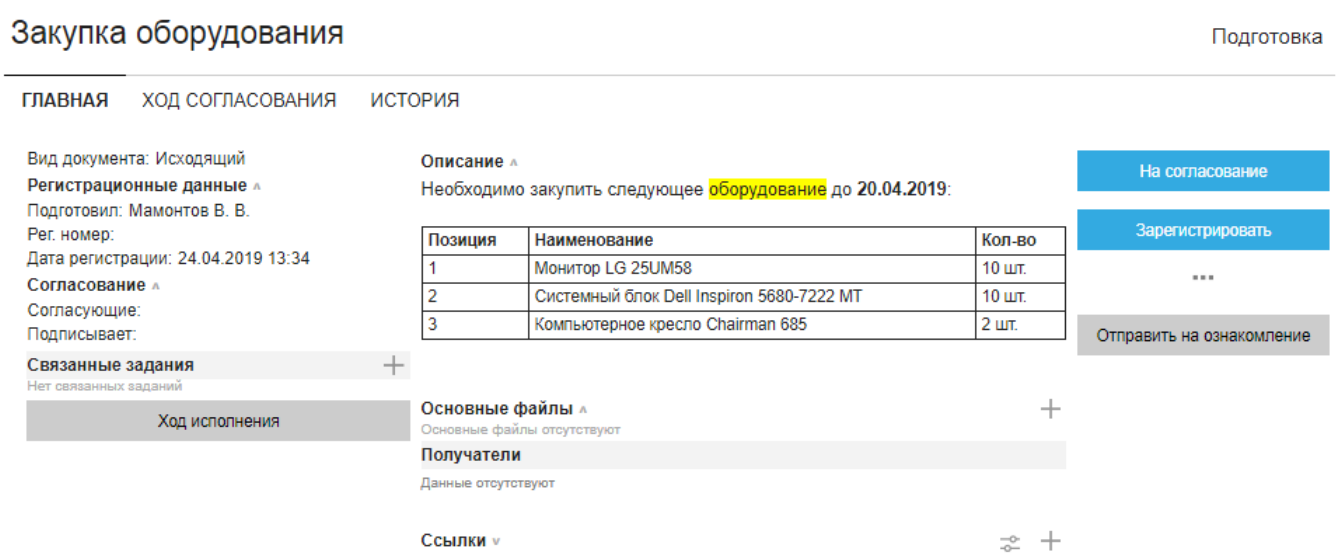


Рисунок 374. Пример отображения текста с форматированием в блоке "Описание" карточки

Если элемент управления Текст настроен на работу с расширенным форматированием, в блоке ввода текста будет показана панель инструментов. В обычном режиме панель инструментов скрыта.
В Windows-клиенте и PMA HTML всегда отображается в виде строки кода.

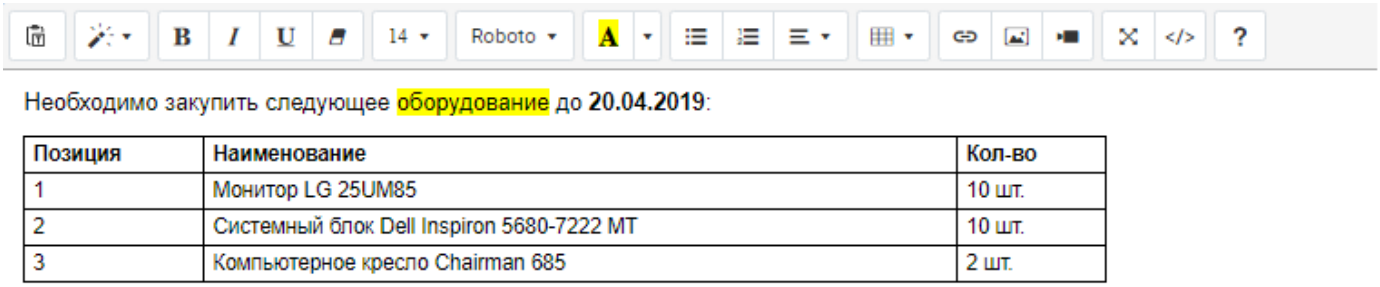


Рисунок 375. Панель инструментов элемента управления "Текст"

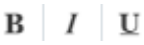
Панель инструментов содержит следующие элементы:



Вставляет текст из буфера обмена без сохранения форматирования.



Устанавливает стиль текста. Доступные варианты стиля: Нормальный, Код, Заголовок 1, Заголовок 2, Заголовок 3, Заголовок 4, Заголовок 5, Заголовок 6.



Устанавливает стиль начертания.

	Сбрасывает настройки стиля для выбранного текста.
14	Устанавливает размер шрифта.
sans-serif	Устанавливает шрифт.
	Устанавливает цвет выделения текста.
	Добавляет нумерованный список.
	Устанавливает способ выравнивания текста; добавляет отступы.
	Добавляет таблицу.
	Добавляет ссылку на внешний ресурс, изображение или видео с внешнего ресурса.
	Разворачивает/сворачивает блок редактора на всё окно.
</>	Переключает редактор в режим ввода исходного (HTML) кода.
?	Вызывает справку по работе с текстом.

При работе с компонентом допускается вставлять изображения из буфера обмена.

При вставке изображения в браузере Internet Explorer, буфер обмена будет очищен сразу после первой вставки изображения. Для повторной вставки изображения потребуется снова его копировать.

Подразделение

Элемент разметки Подразделение используется для выбора сотрудника из *Справочника сотрудников*.

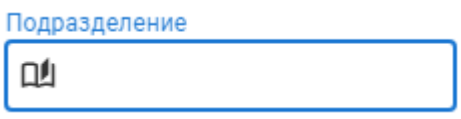


Рисунок 376. Подразделение

Выбор организации или подразделения из справочника

Чтобы выбрать организацию или подразделение из Справочника сотрудников:

1. Нажмите кнопку , чтобы открыть *Справочник сотрудников*.

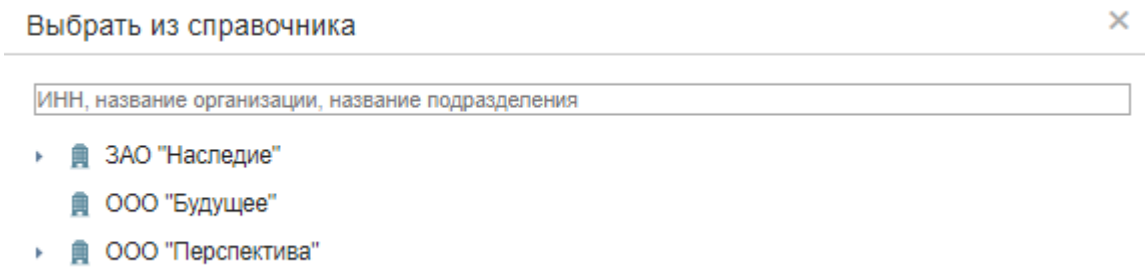


Рисунок 377. Справочник сотрудников

2. Выберите организацию или подразделение из списка.

   Воспользуйтесь строкой поиска, чтобы найти организацию или подразделение. Поиск доступен по названию и по ИНН. Чтобы отобразить дочерние организации или подразделения, нажмите кнопку .

Поиск организации или подразделении по справочнику

Чтобы воспользоваться быстрым поиском, введите часть названия или ИНН организации в поле и нажмите на название искомой организации в списке результатов.

Очистка значения

Нажмите кнопку , чтобы удалить значение.

Сотрудник

> Поддержка элемента управления прекращена, элемент устарел и недоступен для выбора в библиотеке. Рекомендуется использовать ЭУ "Записи справочника сотрудников".

Элемент разметки `Сотрудник` используется для выбора сотрудника из *Справочника сотрудников*.



Рисунок 378. Сотрудник

Выбор сотрудника из списка

Нажмите кнопку и выберите сотрудника из списка. Если настроено, последние выбранные сотрудники будут отображаться в начале списка.

Поиск сотрудника по справочнику

Чтобы воспользоваться быстрым поиском, введите в поле фамилию, имя и/или отчество сотрудника полностью или частично через пробел.

Очистка значения

Нажмите кнопку , чтобы удалить значение.

Сотрудники

> Поддержка элемента управления прекращена, элемент устарел и недоступен для выбора в библиотеке. Рекомендуется использовать ЭУ "Записи справочника сотрудников".

Элемент разметки Сотрудники используется для выбора нескольких сотрудников из *Справочника сотрудников*.



Рисунок 379. Сотрудники

Выбор сотрудника из списка

Аналогично выбору в элементе "Записи справочника сотрудников". Выбранный сотрудник добавляется в конец списка выбранных сотрудников и скрывается списка сотрудников, доступных для выбора.

Поиск сотрудника по справочнику

Аналогично поиску в элементе "Записи справочника сотрудников".

Удаление сотрудника из списка

Чтобы удалить сотрудника из списка выбранных, нажмите кнопку справа от его ФИО.

Записи справочника сотрудников

Элемент разметки Записи справочника сотрудников предназначен для выбора записей (сотрудники, организации, подразделения, группы сотрудников, роли,

должности и поисковые слова) из *Справочника сотрудников*.

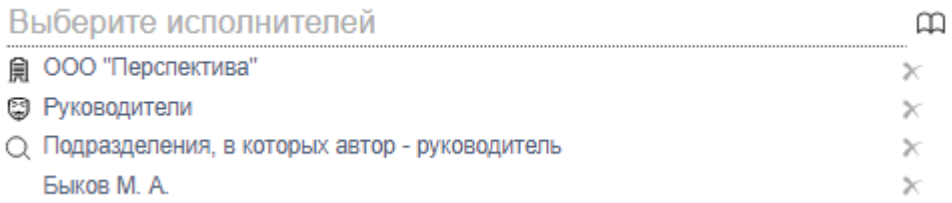


Рисунок 380. Записи справочника сотрудников

В зависимости от настроек могут быть ограничены возможности выбора определённых типов записей, а также области выбора (например, разрешено выбирать только из определённой организации).

Также может быть разрешён выбор только одной или нескольких записей. Или режим, при котором текущий сотрудник будет по умолчанию выбран.

Структура записей справочника сотрудников

Нажмите кнопку 🕮 или выберите элемент разметки и нажмите клавишу \, чтобы открыть *Справочник сотрудников*.

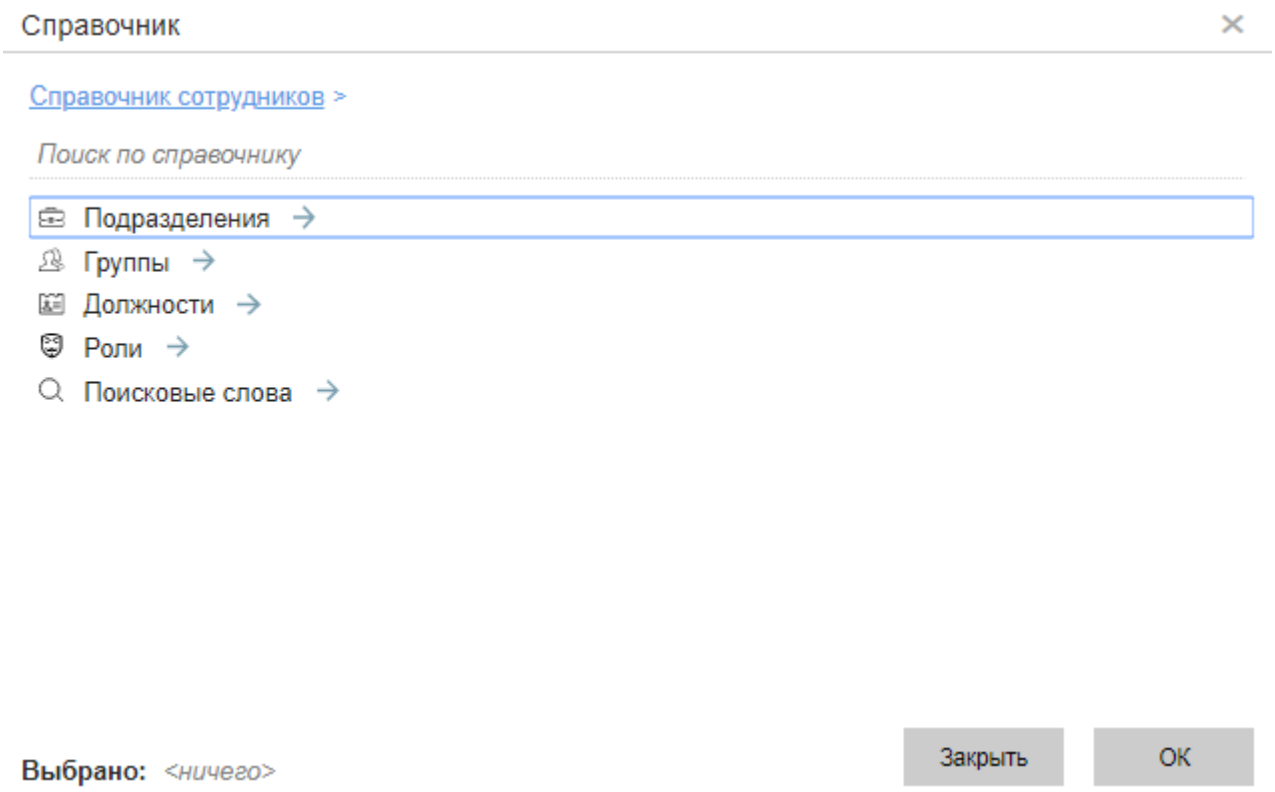


Рисунок 381. Окно выбора значения из Справочника сотрудников

В верхней части окна отображается путь к текущему открытому элементу справочника (например, **Справочник сотрудников › Подразделения › ООО**

"Перспектива). Путь содержит ссылки для перехода к вышестоящим элементам.

Чтобы перейти к дочерним элементам, нажмите кнопку →. Чтобы выйти из содержимого, используйте ссылки в пути к текущему открытому элементу справочника.

Для перемещения между элементами справочника можно использовать клавиши клавиатуры: ↑, ↓, → (для перехода к содержимому выделенного элемента), ← (для выхода из содержимого).

Поиск по справочнику осуществляется из строки поиска. Поиск выполняется по всему справочнику или по открытому элементу (при нахождении внутри элемента). В элементе *Подразделения* поиск выполняется по элементам, тип которых выбран в строке поиска (*Сотрудник* или *Организация*).

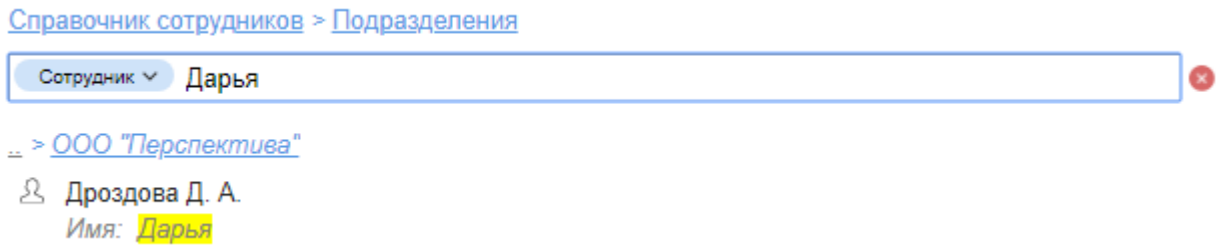


Рисунок 382. Поиск по Справочнику сотрудников

Выбрать запись из справочника

Чтобы выбрать запись, нажмите на неё или выберите с помощью клавиатуры и нажмите клавишу ▯.

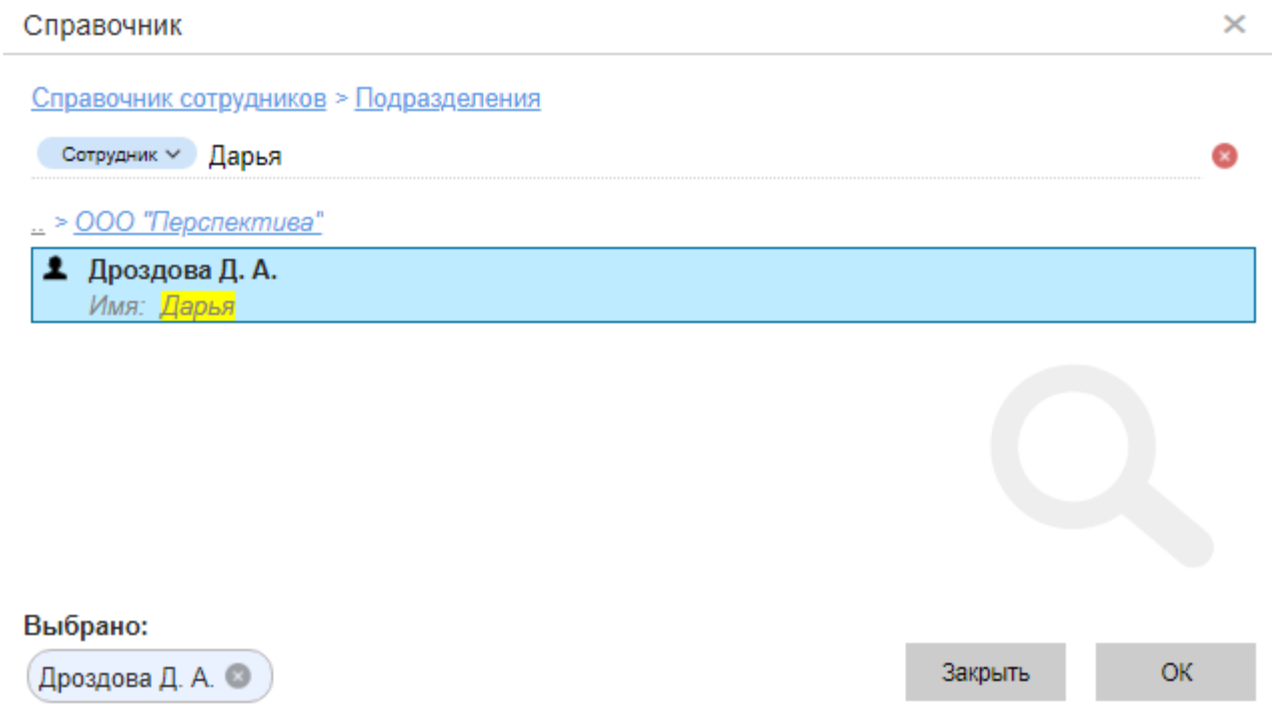


Рисунок 383. Справочник сотрудников с выбранной записью

В режиме множественного выбора элемент будет добавлен в список выбранных элементов в строке *Выбрано:*. Все выбранные записи выделяются жирным шрифтом.

Чтобы снять выбор с элемента в режиме множественного выбора, нажмите кнопку справа от названия элемента в строке *Выбрано:*.

Добавить запись справочника сотрудников

Нажмите кнопку **Выбрать** или **OK** (если разрешается выбирать несколько записей), чтобы добавить запись.

Запись будет добавлена:

- В значение элемента разметки (в режиме выбора одной записи).



Рисунок 384. Элемент разметки в режиме выбора одной записи

- В список значений элемента разметки (в режиме выбора нескольких записей).

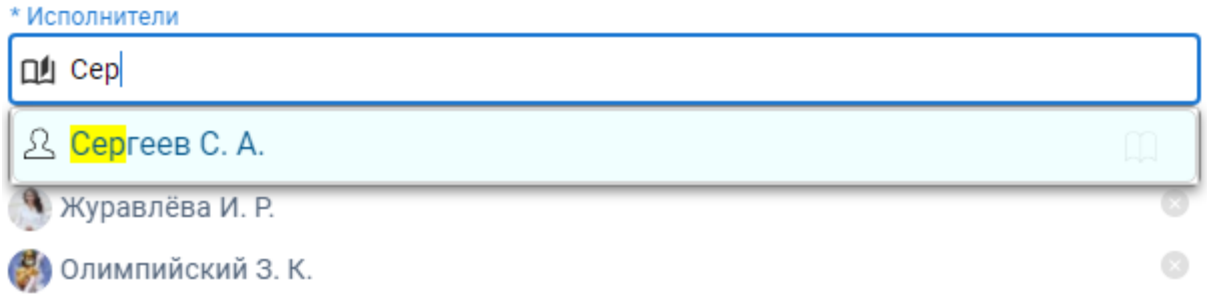


Рисунок 385. Элемент разметки в режиме множественного выбора

Быстрый поиск по справочнику

Чтобы воспользоваться быстрым поиском по справочнику, введите искомое значение в поле элемента управления. Записи, соответствующие условиям запроса, будут показаны в списке. Записи, которые были выбраны, в результатах поиска не отображаются.

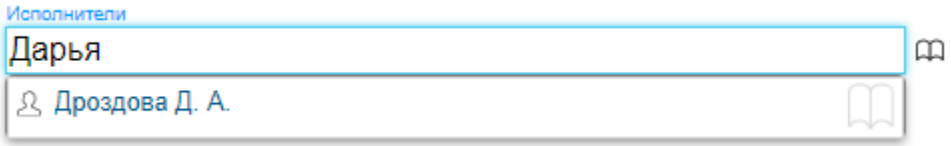


Рисунок 386. Быстрый поиск по справочнику

При поиске действуют общие ограничения по типам элементов и области выбора.

Последние выбранные сотрудники

В списке последних выбранных сотрудников отображаются:

- Последние выбранные исполнители, если элемент используется в карточке задания или группы заданий.
- Последние выбранные записи отображаются во всех других карточках.

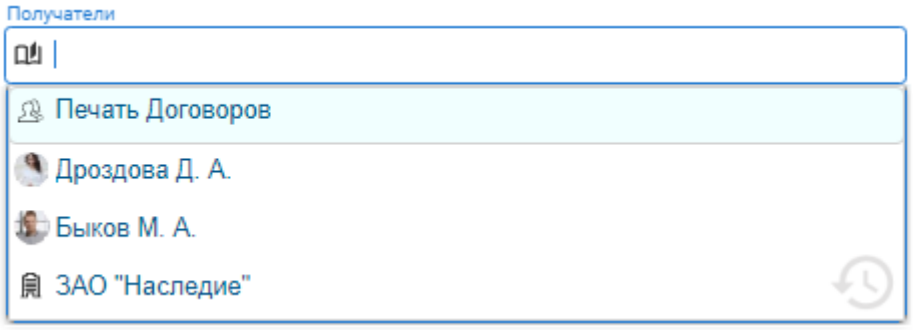


Рисунок 387. Последние выбранные записи

Чтобы открыть список последних, выберите элемент разметки и нажмите

клавишу ↓ или . Поле для ввода текста должно быть пустым, иначе будет вызван быстрый поиск.

Список последних выбранных сотрудников не очищается при перезапуске службы Сервера Web-клиента или при нажатии использовании функции "Вернуть настройки по умолчанию" в профиле пользователя.

Удалить сотрудника из списка в режиме множественного выбора

Нажмите кнопку в строке сотрудника, чтобы удалить его из списка.

Записи справочников

Элемент разметки Записи справочников предназначен для выбора одного или нескольких значений из одного или нескольких справочников.

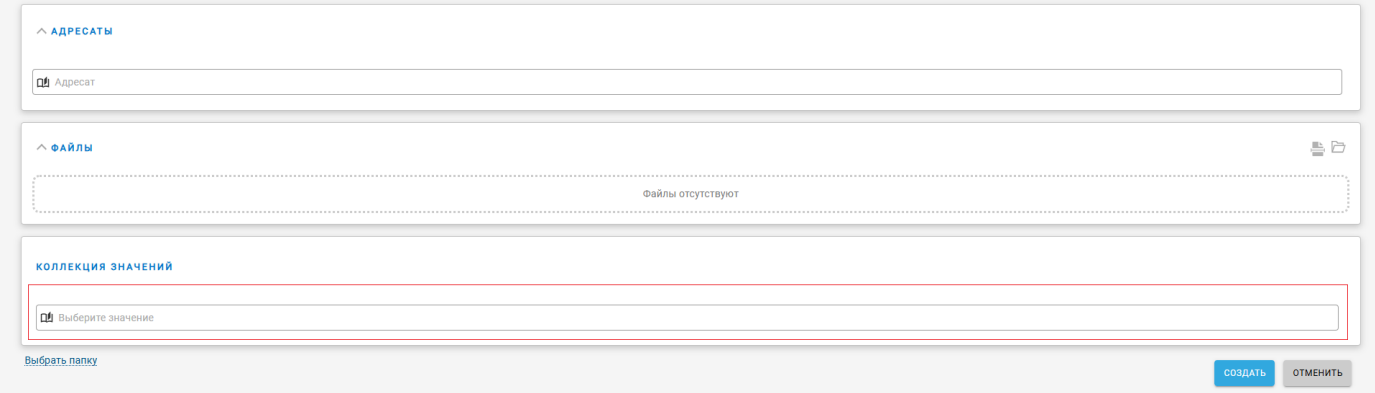


Рисунок 388. ЭУ "Записи справочников" в разметке с режимом создания

В зависимости от настроек могут быть ограничены возможности выбора определённых типов записей, а также области выбора (см. подробнее "Записи справочников").

Также может быть разрешён выбор только одной или нескольких записей. Или режим, при котором определённое значение будет по умолчанию выбрано.

Режим отображения

В зависимости от режима редактирования ЭУ отображается по-разному:

- В режиме *Просмотр* и *Редактирование по месту* ЭУ представляет собой текст, содержащий иконки (свойство **CSS-класс иконки**) и отображаемые названия (свойство **Отображаемое поле**) выбранных элементов справочника. Выбранные элементы отображаются либо в одну строку с разделителем **;**, либо вертикально, с новой строки в зависимости от значения свойства **Вертикальная ориентация**.

- В режиме *Редактирование* ЭУ содержит:
  a. Кнопку 🕮, нажатие на которую открывает окно поиска и выбора элементов с текстовым полем для ввода искомого текста.
  b. Раскрывающийся список найденных элементов справочника с возможностью выбора. Список будет отображен при вводе строки, длина которой больше или равна значению, заданному в свойстве **Индекс поиска**.

     Нажатие на найденный элемент добавляет его в список ниже. Для каждого пункта в списке отображается следующая информация:

     - Если выбор осуществляется из нескольких справочников, отображается название справочника и секции, в которой найден данный элемент.
     - Иконка (свойство **CSS-класс иконки**) и отображаемое название элемента (свойство **Отображаемое поле**). Если в отображаемом названии найдены совпадения, они будут выделены жёлтой подсветкой.
     - Дополнительные сведения об элементе (свойство **Дополнительные поля**). Если поле дополнительных сведений также является полем для поиска и в нём найдены совпадения, они будут выделены жёлтой подсветкой.
  c. Список выбранных элементов справочника.

Выбор из нескольких типов значений

Если для выбора доступны несколько типов значений, в режиме редактирования значения ЭУ нажмите кнопку 🕮, чтобы открыть окно выбора элементов.

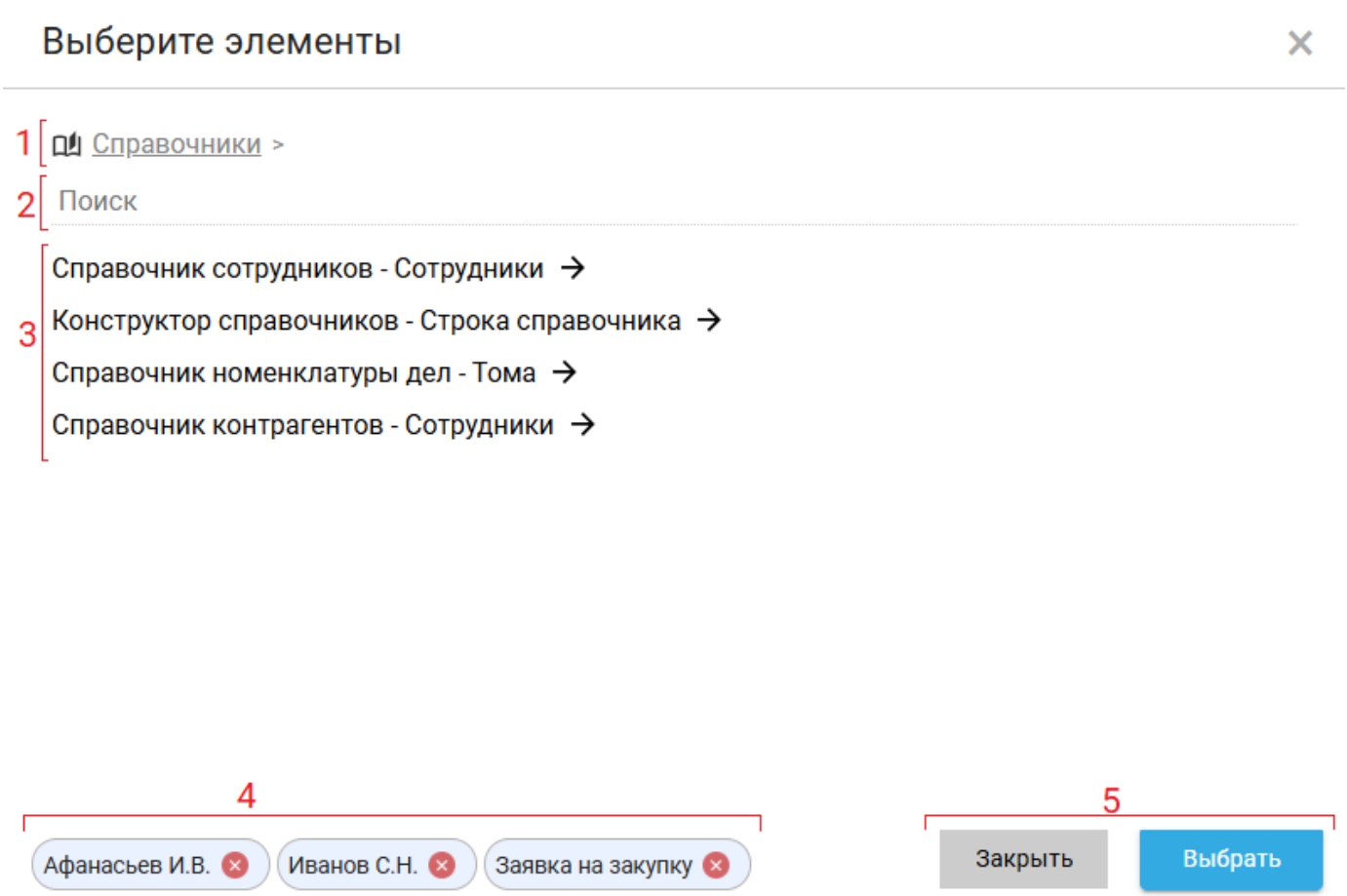


Рисунок 389. Окно поиска и выбора значения элементов при выборе из нескольких типов значений

Окно поиска и выбора элементов содержит:

1. Указатель типа элементов. В заголовке окна отображается путь к текущей секции. Чтобы перейти к дочерним элементам, нажмите кнопку →. Чтобы перейти к промежуточному уровню, используйте ссылки в пути к текущему открытому элементу справочника. Чтобы вернуться к списку типов значений, нажмите на кнопку 🕮.
2. Поле для ввода текстового поискового запроса.
3. Доступные типы значений. Чтобы перейти к выбору значения, нажмите на строку с типом значений.
4. Список выбранных значений с возможностью удаления элементов.
5. Кнопки **Закрыть** и **Выбрать**. Кнопка **Закрыть** закрывает окно без изменения значения ЭУ, аналогичное действие поведение будет при закрытии формы кнопкой X в заголовке, либо при клике по странице вне формы. Кнопка **Выбрать** закрывает окно, значение ЭУ будет установлено как список выбранных в форме элементов.

Выбор из одного типа значений

Если для выбора доступен только один тип значений, в режиме редактирования значения ЭУ нажмите кнопку , чтобы открыть окно выбора элементов.

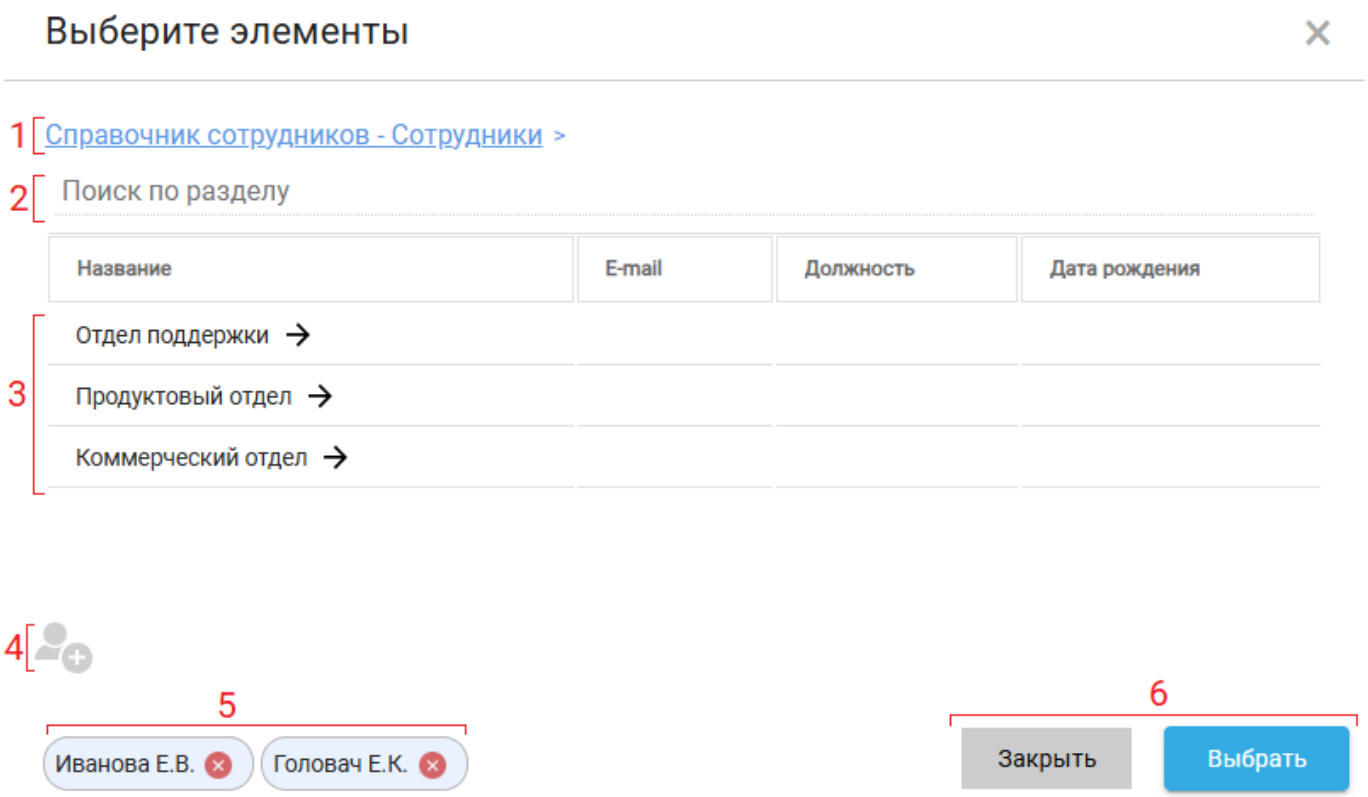


Рисунок 390. Окно поиска и выбора значения элементов при выборе из одного типа значений

Окно поиска и выбора элементов содержит:

1. Указатель типа элементов. В заголовке блока отображается путь к текущей секции. Чтобы перейти к промежуточному уровню, используйте ссылки в пути к текущему открытому элементу справочника. Чтобы вернуться к списку типов значений, нажмите на кнопку .
2. Поле для ввода текстового поискового запроса.
3. Доступные типы значений. Чтобы перейти к выбору значения, нажмите на строку с типом значений.

   Если настроено свойство **Дополнительные поля**, таблица будет содержать столбец с названием элемента (свойство "Отображаемое поле"), а также столбцы для всех полей, указанных в свойстве **Дополнительные поля**.
   Если в таблице какого-либо типа элементов отображается выбранный элемент, он будет выделен жирным шрифтом.

4. Кнопка добавления нового значения в справочник. Отображается, если

соответствующая возможность включена в настройках типа значения и текущий сотрудник обладает правом на редактирование справочника и строки, указанной в свойствах "Узел" или "Ограничение".

Добавлять разрешается следующие новые значения:

- Подразделения справочника контрагентов
- Сотрудники справочника контрагентов
- Подразделения справочника сотрудников
- Сотрудники справочника сотрудников
- Узлы конструктора справочников
- Элементы конструктора справочников

5. Список выбранных значений с возможностью удаления элементов.
6. Кнопки **Закрыть** и **Выбрать**. Кнопка **Закрыть** закрывает окно без изменения значения ЭУ, аналогичное действие поведение будет при закрытии формы кнопкой X в заголовке, либо при клике по странице вне формы. Кнопка **Выбрать** закрывает окно, значение ЭУ будет установлено как список выбранных в форме элементов.

Последние выбранные значения

В списке последних выбранных значений отображаются записи из всех справочников, которые были выбраны ранее.

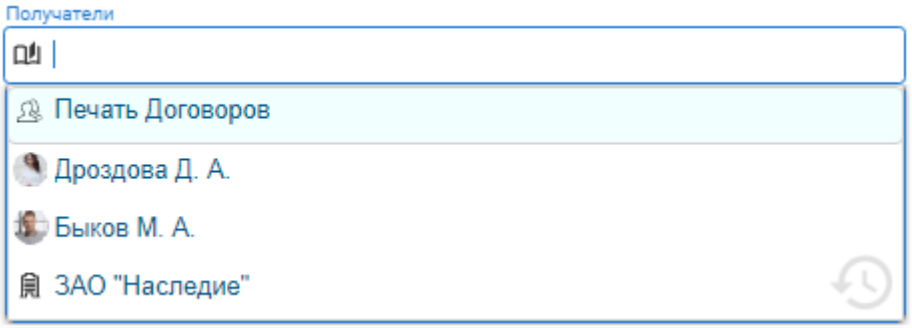


Рисунок 391. Последние выбранные записи

Чтобы открыть список последних, выберите элемент разметки и нажмите клавишу ↓ или . Поле для ввода текста должно быть пустым, иначе будет вызван быстрый поиск.

Список последних выбранных сотрудников не очищается при перезапуске службы Сервера Web-клиента или при нажатии использовании функции "Вернуть настройки по умолчанию" в профиле пользователя.

Добавить запись справочника

Нажмите кнопку **Выбрать** или **OK** (если разрешается выбирать несколько записей), чтобы добавить запись.

Запись будет добавлена:

- В значение элемента разметки (в режиме выбора одной записи).



Рисунок 392. Элемент разметки в режиме выбора одной записи

- В список значений элемента разметки (в режиме выбора нескольких записей).

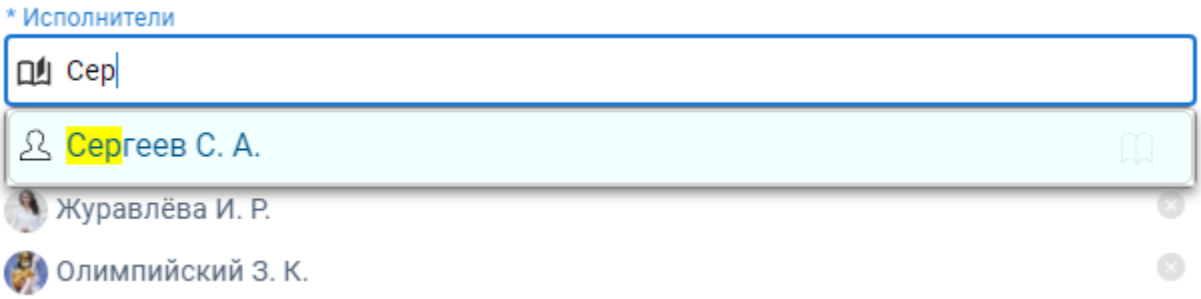


Рисунок 393. Элемент разметки в режиме множественного выбора

Удалить значение из списка в режиме множественного выбора

Нажмите кнопку ✕ в строке значения, чтобы удалить его из списка.

Поиск по справочнику

Для ЭУ доступны два вида поиска значений: быстрый поиск из поля редактирования и поиск из окна элемента управления. Чтобы воспользоваться быстрым поиском по справочнику, введите искомое значение в поле элемента управления. Записи, соответствующие условиям запроса, будут показаны в списке. Записи, которые были выбраны, в результатах поиска не отображаются.



Обратите внимание, поиск активируется при вводе количества символов, равного или превышающего указанное в свойстве "Поисковой индекс"

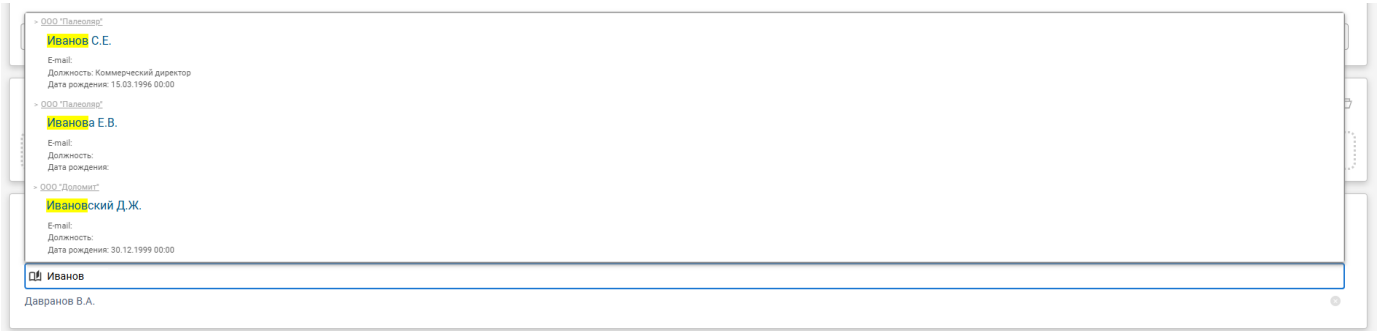


Рисунок 394. Быстрый поиск по записям справочников

Поиск осуществляется по полям секции, указанным в параметре "Отображаемое поле". Результаты сортируются по возрастанию значения отображаемого поля и по справочнику. Справочники в результатах поиска отображаются в порядке, заданном в свойстве "Типы значений".

Количество результатов на "порцию" поиска зависит от свойства "Кол-во загружаемых за раз элементов". Больше результатов можно получить при нажатии на кнопку **Показать ещё**.

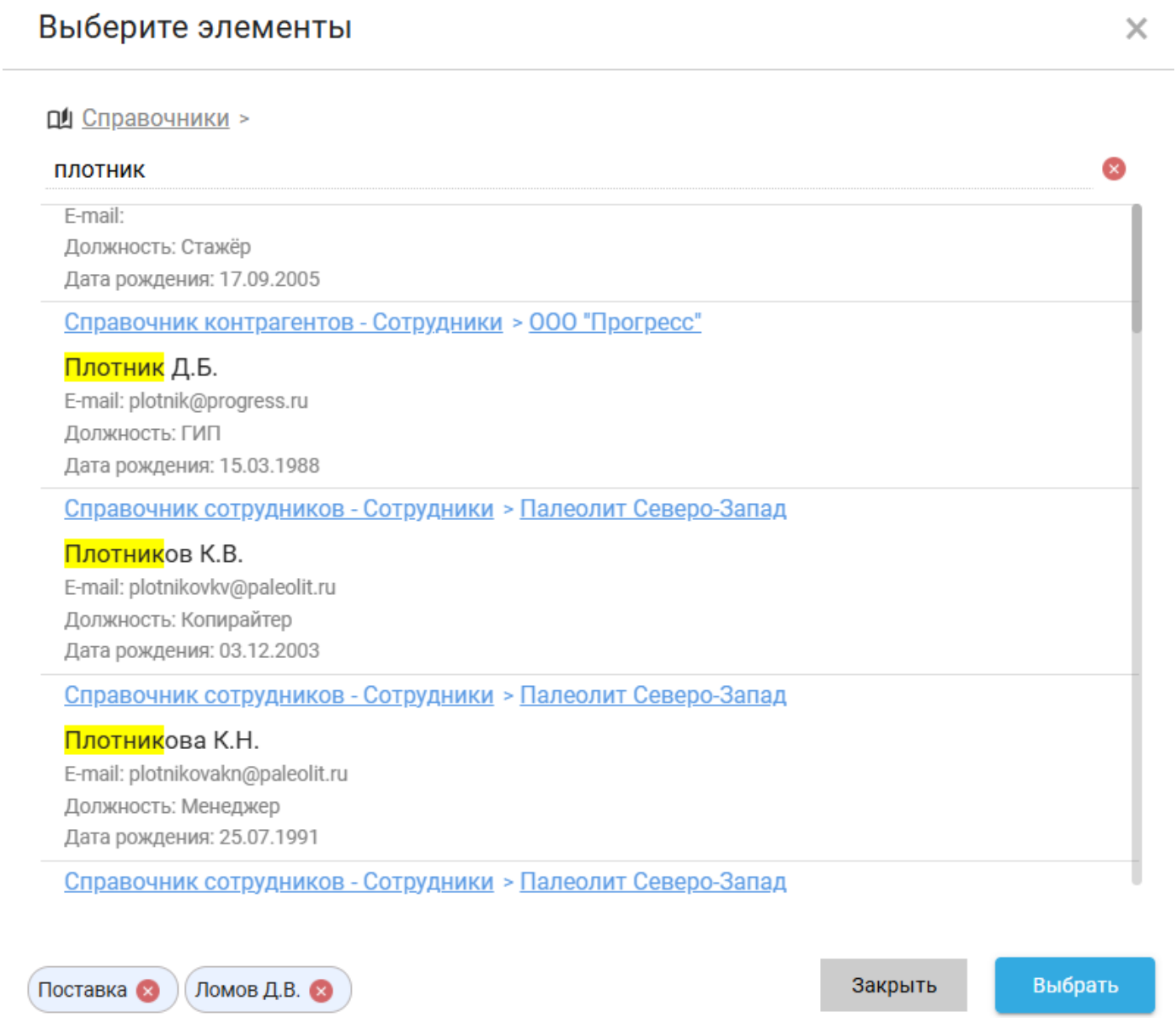


Рисунок 395. Поиск в ЭУ "Записи справочников"

Если также задан параметр "Поле для поиска", поиск будет осуществляться по указанным в нём полям (поле из свойства **Отображаемое поле** в поиске участвовать не будет).

Подразделение контрагента

Для выбора организации или подразделения из *Справочника контрагентов* предназначен элемент управления Подразделение контрагента.

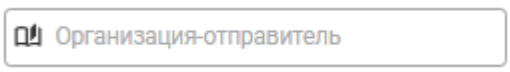


Рисунок 396. Элемент "Подразделение контрагента"

Выбор организации из справочника

Чтобы выбрать организацию-контрагента, откройте *Справочник контрагентов* нажатием кнопки . Затем выберите организацию или подразделение из списка

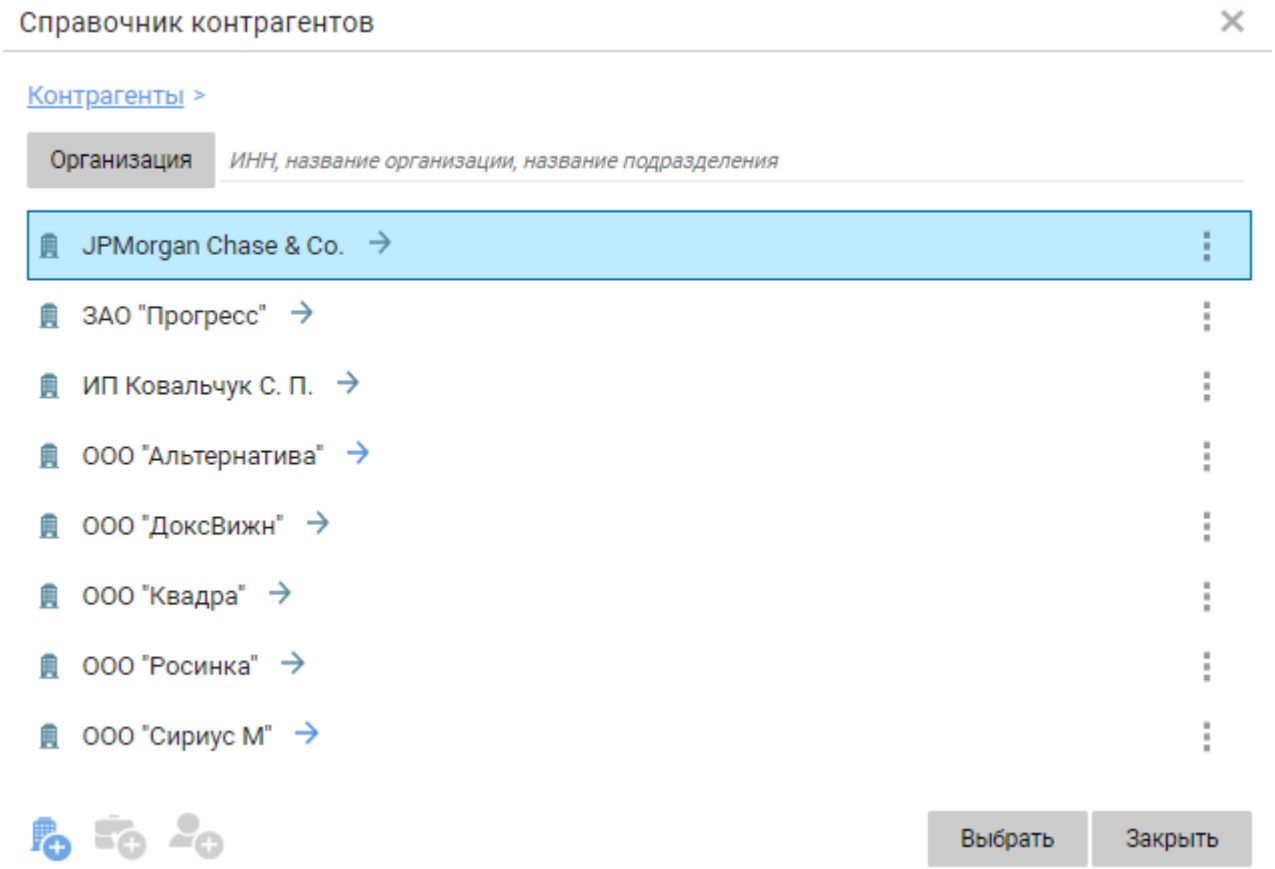


Рисунок 397. Справочник контрагентов

Используйте строку поиска, чтобы найти организации или подразделения. Возможен поиск по названию и по ИНН. Чтобы перейти к дочерним организациям или подразделениям используйте кнопку →. Кнопка отключена, если нет дочерних организаций или подразделений и сотрудников.

Выбирайте организации, подразделений нажатием кнопки **Выбрать**.

Быстрый поиск по справочнику

Чтобы выбрать организации с помощью быстрого поиска введите часть названия организации или подразделения в поле и щелкните по названию искомой организации в списке результатов.



Рисунок 398. Результаты быстрого поиска по Справочнику контрагентов

Просмотр информации об организации/подразделении

При выборе значения из справочника нажатие кнопки ⋮ **> Информация** или пункта *Информация* в контекстном меню узла вызывает разметку *Просмотр информации о …*.

При нажатии кнопки ⓘ в результатах быстрого поиска или наведение курсора на выбранный элемент в разметке (например 🏢) документа вызывает разметку *Справка информации о …*, содержащую краткую информацию об узле.

Очистка значения

Нажмите кнопку ⊗ для удаления значения.

Партнер или сотрудник контрагента

Чтобы выбрать сотрудника из *Справочника контрагентов* используйте элемент управления Партнер.



Рисунок 399. Элемент управления "Партнер"

Выбрать сотрудника из справочника

Чтобы выбрать сотрудника контрагента из справочника:

1. Нажмите кнопку 🕮.

   Будет открыт *Справочник контрагентов*.

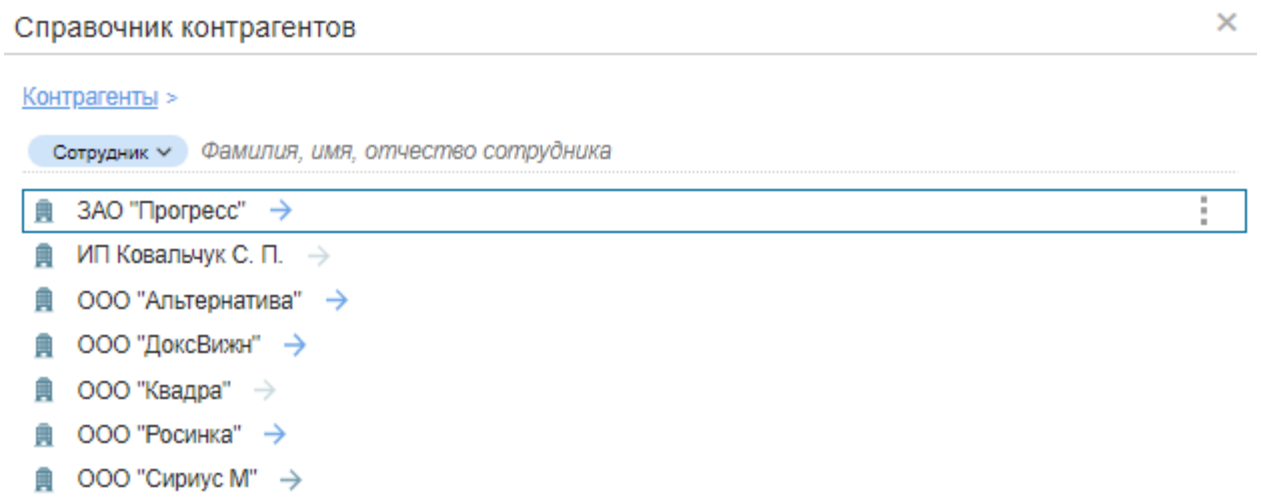


Рисунок 400. Справочник контрагентов

2. Нажмите кнопку →, чтобы перейти к сотрудникам организации.
3. Выберите сотрудника.

> Используйте строку поиска, чтобы найти сотрудника по ФИО.

4. Нажмите кнопку **Выбрать**.

Быстрый поиск по справочнику

Чтобы выбрать сотрудника с помощью быстрого поиска:

1. Введите часть его ФИО в поле и щелкните по записи сотрудника в списке результатов.

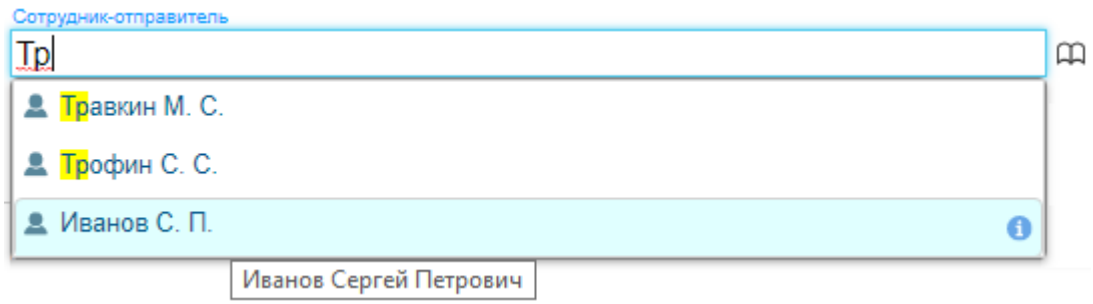


Рисунок 401. Результаты быстрого поиска по Справочнику контрагентов

> Область поиска сотрудника будет ограничена, если предварительно выбрать организацию или подразделение сотрудника. Область поиска будет ограничена организацией или подразделением, если ввести название организации или подразделения и выбрать требуемую из списка.

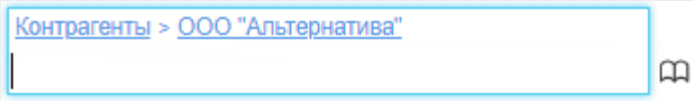


Рисунок 402. Элемент "Партнер" с установленным ограничением области поиска

Посмотреть информацию о сотруднике

Чтобы показать информацию о сотруднике:

1. Нажмите кнопку в результатах быстрого поиска.

   Кнопка появляется при наведении курсора на запись.

2. В качестве альтернативы при выборе сотрудника из *Справочника контрагентов* нажмите кнопку **> Информация**.

Очистить значение

Чтобы очистить значение:

1. Нажмите кнопку .

Строка конструктора справочников

Элемент разметки `Строка конструктора справочников` используется для выбора значения из *Конструктора справочников*.



Рисунок 403. Строка конструктора справочников

Настройки обновления справочников влияют и на конструктор справочников. На период работы с конструктором рекомендуется установить меньший период обновления, см. подробнее "Функциональность "Общие справочники"".

Выбор значения из справочника

Чтобы выбрать значение из Конструктора справочников:

1. Откройте *Конструктор справочников* нажатием кнопки .

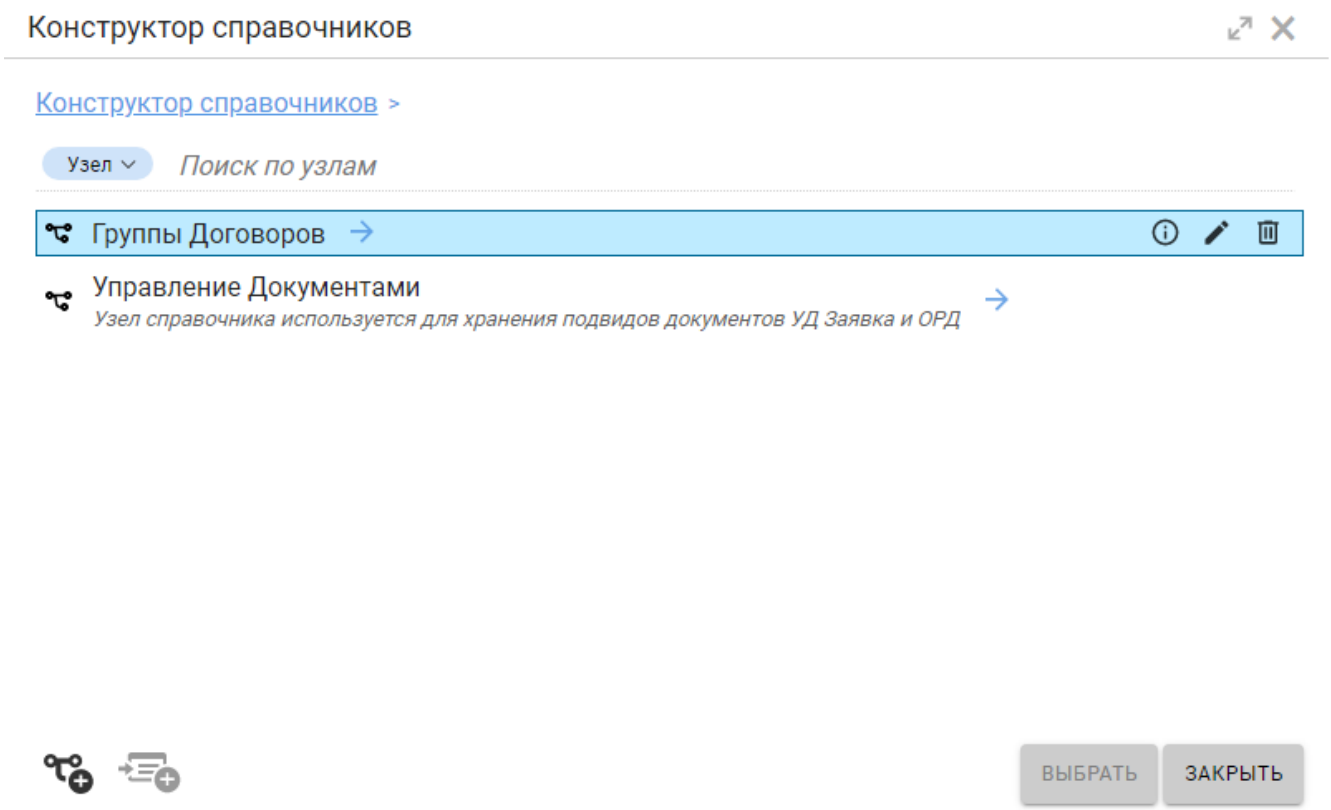


Рисунок 404. Конструктор справочников с разрешёнными изменениями

Данные справочника представлены в виде списка с узлами и записями.

Если при настройке Web-клиента для элемента `Строка конструктора справочников` выбрано отображение записей определённого узла, справочник может содержать только записи, без узлов.

2. Выберите строку со значком из списка.

Используйте поиск, чтобы найти строки и узлы справочника.

Чтобы раскрыть узел *Конструктора справочников*, используйте кнопку .

В строке конструктора справочников значения сортируются по алфавиту от А до Я.

3. Чтобы открыть разметку просмотра для узла или строки, нажмите на кнопку .

4. Вернуться на предыдущий уровень вложенности можно нажатием на ссылки в верхней части окна конструктора справочников.

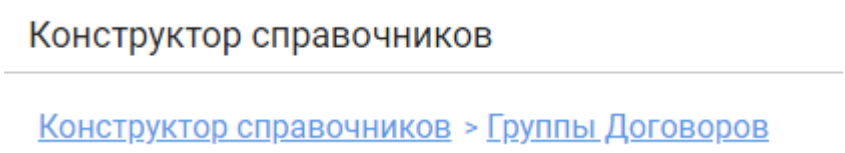


Рисунок 405. Навигационные ссылки

Если ЭУ используется в разметке с режимом *Создание* или *Редактирование*, и разметка не была сохранена явным образом, при обновлении страницы выбранное в элементе значение будет сброшено, даже если ЭУ находится в режиме редактирования по месту.

Поиск по справочнику

Чтобы воспользоваться быстрым поиском, введите часть искомой записи справочника. Поиск может выполняться по записям и узлам в зависимости от настроек ЭУ "Строка конструктора справочников".

Нажмите кнопку ⊗, чтобы удалить введённое в поисковую строку значение.

Редактирование конструктора справочников

Конструктор справочников можно редактировать. Возможность редактирования зависит от настроек элемента управления "Строка конструктора справочников".

Редактирование конструктора справочников выполняется при помощи следующих кнопок:

Информация ⓘ

Открывает окно с информацией об узле или строке конструктора справочников.

Карандаш ✎

Позволяет редактировать название и описание узлов и строк справочника.

Корзина 🗑

Позволяет удалить узел или строку конструктора справочников.

Удаление необходимо будет подтвердить в отдельном появившемся окне.

Кнопка

Позволяет создать новый узел конструктора справочников.

Кнопка →

По нажатию на эту кнопку можно перейти внутрь узла конструктора справочника.

Кнопка

Нажатие на кнопку создаёт новую строку конструктора справочников.

Таблица

Разметка карточки может содержать таблицу, предназначенную для работы с набором однотипных значение. Состав доступных для ввода данных определяется настройками таблицы.

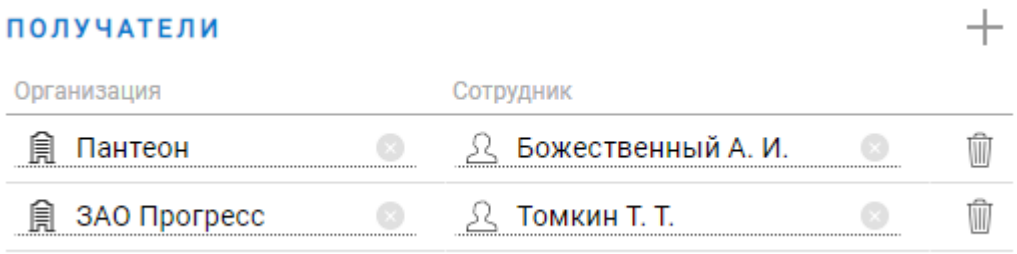


Рисунок 406. Пример таблицы с двумя столбцами

Добавление строки

Чтобы добавить новую строку в таблицу, нажмите кнопку +.

Удаление строки

Чтобы удалить строку, нажмите кнопку .

Изменение значения в строке

Способ ввода значения в ячейку таблицы зависит от типа данных, связанных с этой ячейкой. В самом простом случае, чтобы изменить значение, необходимо переключиться в режим редактирования, нажав на ячейку.

При необходимости пользователь может изменить порядок отображения строк в таблице. Откройте карточку в режиме редактирования и перетащите строку вверх или вниз. При этом форма курсора изменится на "Руку".

Комментарии

Компонент Комментарии используется для обмена информацией между сотрудниками при работе над заданиями.



Рисунок 407. Пример обсуждения в карточке с помощью компонента "Комментарии"

Комментарии сортируются по дате добавления, новые комментарии отображаются в конце списка с указанием автора и времени добавления. Голубым цветом выделяются комментарии текущего пользователя.

Старые комментарии могут быть скрыты. В этом случае будет показана ссылка *Показать предыдущие комментарии* для просмотра скрытых комментариев.

Добавление комментария

Чтобы добавить комментарий, введите его в поле Введите комментарий и нажмите клавишу `Enter` или кнопку ➤. Чтобы перейти на новую строку в многострочном комментарии, используйте сочетание клавиш `Ctrl` + `Enter` или `Shift` + `Enter`. Чтобы очистить содержимое поля ввода, нажмите клавишу `Esc`.

Изменение комментария

Пользователь может изменить комментарий при наличии достаточных прав доступа. Нажмите кнопку ✎ над строкой комментария, чтобы перейти к его редактированию.

Удаление комментария

Для удаления комментария требуются права на данную операцию. Нажмите кнопку 🗑 над строкой комментария, чтобы удалить его. Текст удалённого комментария меняется на `Комментарий был удалён. Восстановить`. До обновления страницы удалённый комментарий можно восстановить щелчком по ссылке *Восстановить*.

Функция комментариев с HTML-редактором

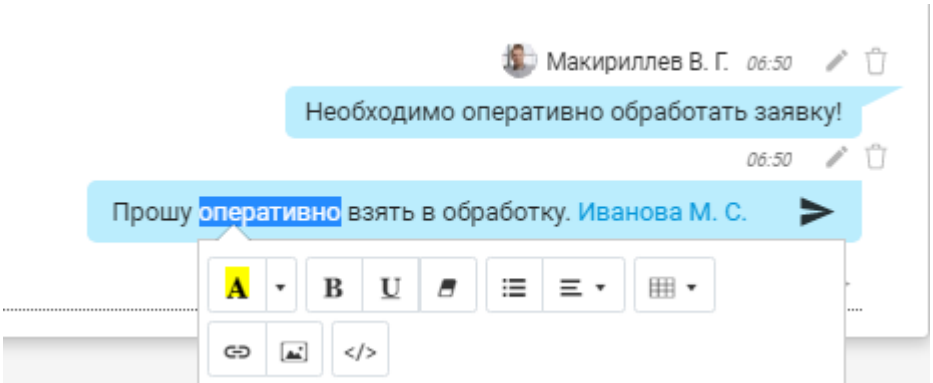
Функция `Использовать HTML редактор` открывает возможность форматирования текста и упоминания сотрудников.

> ⚠ Возможности форматирования комментариев и упоминания сотрудников доступны, только если активировано свойство в программе Конструктор Web-разметок.

Упоминания других сотрудников в комментарии

Чтобы упомянуть пользователя, введите символ `@` и начните вводить ФИО сотрудника. Чтобы упомянуть нескольких сотрудников, введите символ `@` несколько раз.

Комментарии с HTML-редактором



Сотруднику не будет уведомлен о том, что его упомянули по электронной почте, данная функциональность должна быть настроена администратором модуля при помощи бизнес-процесса или иных инструментов.

Форматирование комментариев

Функция добавляет возможности редактировать комментарии в процессе ввода. Если выделить текст комментария, появится всплывающая панель форматирования с набором инструментов.

	Устанавливает цвет выделения текста.
	Устанавливает стиль начертания.
	Сбрасывает настройки стиля для выбранного текста.
	Добавляет маркированный список.
	Добавляет нумерованный список.
	Устанавливает способ выравнивания текста; добавляет отступы.
	Добавляет таблицу.
	Добавляет ссылку на внешний ресурс или изображение с внешнего ресурса.
</>	Переключает редактор в режим ввода исходного (HTML) кода.

При работе с компонентом допускается вставлять изображения из буфера обмена.

При вставке изображения в браузере Internet Explorer, буфер обмена будет очищен сразу после первой вставки изображения. Для повторной вставки изображения потребуется снова его копировать.

Дерево исполнения заданий

Элемент Дерево исполнения предоставляет пользователю инструмент для ознакомления с ходом исполнения заданий и групп заданий, связанных с документом, заданием или группой заданий.



Рисунок 408. Дерево исполнения в карточке "Документ"

Корневым элементом дерева является текущая карточка, с которой связаны задания и группы заданий. Задания и группы заданий могут иметь подчиненные задания и группы заданий.

Если в документе нет связанных заданий, Дерево исполнения будет содержать только узел данного документа.

Если группа заданий содержит единственное задание, то вместо группы будет отображаться данное задание.

Работа с данными элемента

Все подчиненные задания и группы заданий будут скрыты, если количество исполнителей в них превышает установленный предел. В таком случае под элементом группы заданий будет отображаться кнопка , при нажатии которой группа будет развёрнута.

Чтобы свернуть все группы заданий, нажмите кнопку . Чтобы раскрыть все группы заданий, нажмите кнопку . При этом группы заданий, свёрнутые из-за

большого количества исполнителей, развёрнуты не будут.

Чтобы свернуть одну определённую группу заданий, нажмите кнопку , отображаемую под элементом. Чтобы развернуть, нажмите кнопку .

Чтобы обновить содержимое Дерева исполнения, нажмите кнопку .

Если дерево исполнения открыто в дочерней карточке задания, будут доступны кнопки для отображения всего дерева заданий и возврата к просмотру с текущего задания: и .

Чтобы изменить масштаб Дерева исполнения, поверните колесо мыши вперед или назад.

Графическое оформление значка элемента

Таблица 17. Цветовое оформление основного значка символизирует состояние задания/группы заданий:

Синий	Новое, не выполнено.
Зелёный	Выполнено.
Красный	Отклонено.
Синий с красным контуром	Просрочено.
Жёлтый	На доработке.

Таблица 18. Основной значок элемента определяет его тип:

	Элемент является заданием.
	Делегированное задание.
	Элемент является группой заданий с параллельным исполнением.
	Элемент является группой заданий с последовательным исполнением.

Таблица 19. Дополнительные индикаторы на значке указывают на особые настройки и состояния задания или группы заданий:

	Находится на контроле.
	Имеет высокую важность.

	Приложен отчёт.
	Приложена карточка отчёта.
	Задание ответственного исполнителя.

Получение информации по элементу дерева исполнения

Чтобы получить краткую информацию об элементе дерева, наведите на него курсор. Во всплывающем окне будут отображены:

Для задания:

- Исполнитель или исполнители,
- Состояние задания,
- Срок исполнения,
- Название.

Для группы заданий:

- Исполнители,
- Состояние задания,
- Срок исполнения,
- Вариант исполнения,
- Название.

Чтобы получить полную *справку* по элементу, выберите его в Дереве исполнения. Будет открыт дополнительный блок со следующей информацией:

Для заданий:

- Состояние задания,
- Исполнитель, его должность и фотография из Справочника сотрудников,
- Срок исполнения,
- Дата фактического завершения,
- Завершивший сотрудник,
- Автор задания,
- Дата создания,
- Название,
- Контролёр и дата контроля,

- Текстовый отчёт исполнителя или ссылка на карточку/файл приложенные в качестве отчёта,
- Содержание,
- Информация о делегировании (если задание было делегировано).

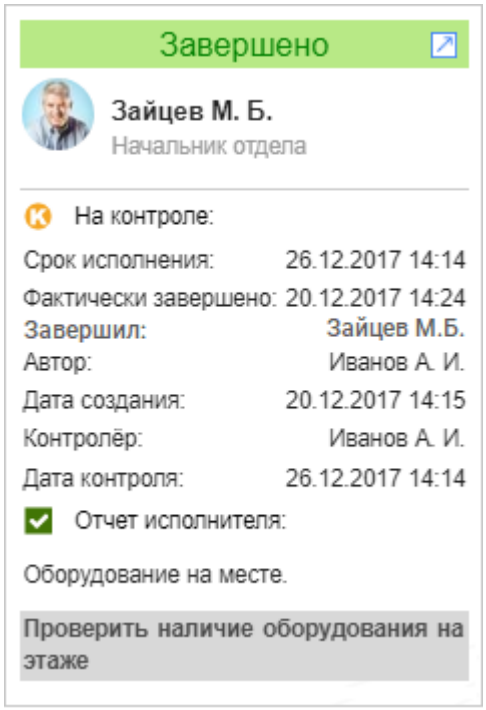


Рисунок 409. Пример справки для задания

Для группы заданий:

- Вариант исполнения с общим количеством исполнителей,
- Название,
- Состояние исполнения группы заданий,
- Список исполнителей,
- Количество выполненных заданий,
- Срок исполнения,
- Автор группы заданий,
- Контролёр и дата контроля,
- Содержание.

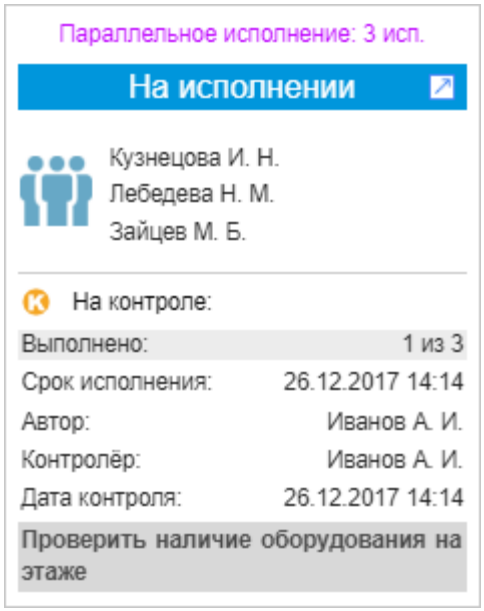


Рисунок 410. Пример справки для группы заданий

Чтобы перейти к карточке задания или группы заданий, нажмите на отображаемый вверху *справки* значок .

Если у пользователя нет прав на карточку задания или группы заданий, то просмотр *справки* и переход к карточке будут недоступны.

Таблица исполнения заданий

Элемент Таблица исполнения заданий предоставляет пользователю аналогичные элементу Дерево исполнения заданий возможности: ознакомление с ходом исполнения заданий и групп заданий, связанных с документом, заданием или группой заданий.

Основным отличием данного элемента является формат представления данных: в Таблице исполнения заданий зависимости заданий и групп заданий представлены в виде иерархического списка.

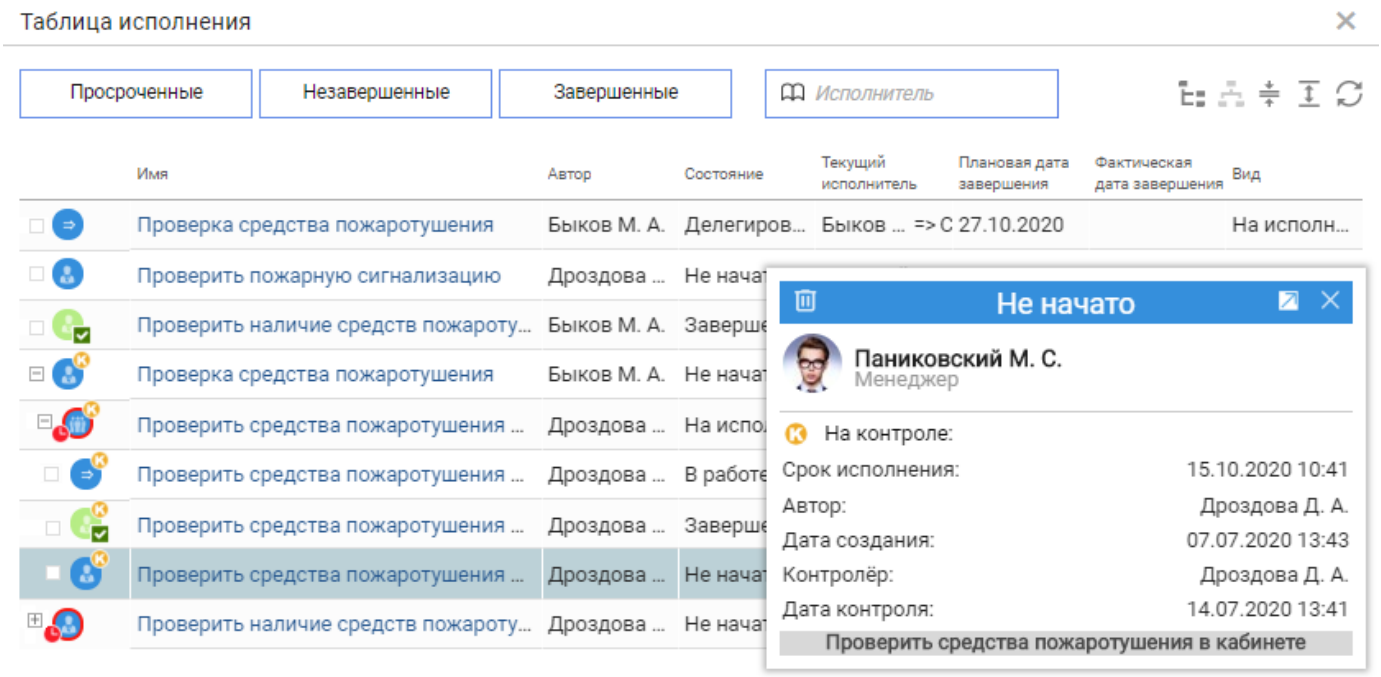


Рисунок 411. Таблица исполнения в карточке Документ

В первых двух колонках таблицы всегда отображаются значок элемента и имя элемента, другие колонки могут быть изменены при настройке Web-клиента.

Графическое оформление значка элемента

Таблица 20. Цветовое оформление основного значка символизирует состояние задания/группы заданий:

Синий	Новое, не выполнено.
Зелёный	Выполнено.
Красный	Отклонено.
Синий с красным контуром	Просрочено.
Жёлтый	На доработке.

Таблица 21. Основной значок элемента определяет его тип:

	Элемент является заданием.
	Делегированное задание.
	Элемент является группой заданий с параллельным исполнением.
	Элемент является группой заданий с последовательным исполнением.

Таблица 22. Дополнительные индикаторы на значке указывают на особые настройки и состояния задания или группы заданий:

	Находится на контроле.
	Имеет высокую важность.
	Приложен отчёт.
	Приложена карточка отчёта.
	Задание ответственного исполнителя.



Значение в колонке *Текущий исполнитель* будет пустым, если задание не было отправлено исполнителю. Текущий исполнитель также не отображается для Группы заданий.

Данные в таблице исполнения

По умолчанию задания первого уровня в таблице исполнения отсортированы по столбцу *Дата отправки* на каждом уровне вложенности. Выше помещаются задания с более ранней датой.

Сортировку можно изменить нажатием на заголовок колонки в таблице. При этом слева от заголовка появляется иконка сортировки . Сортировку можно инвертировать повторным нажатием на иконку. Сортировка не сохраняется при закрытии таблицы. В остальном сортировка действует аналогично сортировке в гриде.

Работа с данными элемента

Свернуть/развернуть всё

Нажмите кнопку , чтобы скрыть все задания до первого уровня, или , чтобы показать все вложенные задания и группы заданий.

Просмотреть всё дерево заданий

Находясь в карточке подчинённого задания нажмите кнопку , чтобы в таблице были показаны также родительские задания или группы заданий. При этом текущее задание будет выделено жирным шрифтом. Для возврата к просмотру списка с текущего задания нажмите кнопку .

Обновить список исполнения

Нажмите кнопку , чтобы обновить список исполнения.

Применение фильтра заданий

Нажмите кнопку **Просроченные**, **Незавершённые**[4] или **завершённые**, чтобы применить фильтр по состоянию задания. При применении фильтра отображается вся иерархия заданий до задания, соответствующего фильтру.

Не соответствующие фильтру родительские задания или группы заданий будут осветлены. Фильтр сбрасывается при повторном выборе текущего фильтра, выборе другого фильтра и поиске по исполнителю.

Поиск по исполнителю

Чтобы найти задания по исполнителю, воспользуйтесь полем *Исполнитель*:

1. Нажмите кнопку для выбора сотрудника из справочника сотрудников, или воспользуйтесь быстрым поиском по справочнику.
2. Поиск происходит моментально. В таблице будут оставлены задания указанного исполнителя (текущего или назначенного) и их родительские задания и/или группы заданий.

Получение информации по заданию или группе заданий

Нажмите на основной значок элемента, например, , чтобы открыть диалоговое окно со справкой по заданию.

Справка может включать следующую информацию:

Для заданий:

- Состояние задания,
- Исполнитель, его должность и фотография из Справочника сотрудников,
- Срок исполнения,
- Дата фактического завершения,
- Завершивший сотрудник,
- Автор задания,
- Дата создания,
- Название,
- Контролёр и дата контроля,
- Текстовый отчёт исполнителя или ссылка на карточку/файл приложенные в качестве отчёта,

- Содержание,
- Информация о делегировании (если задание было делегировано).

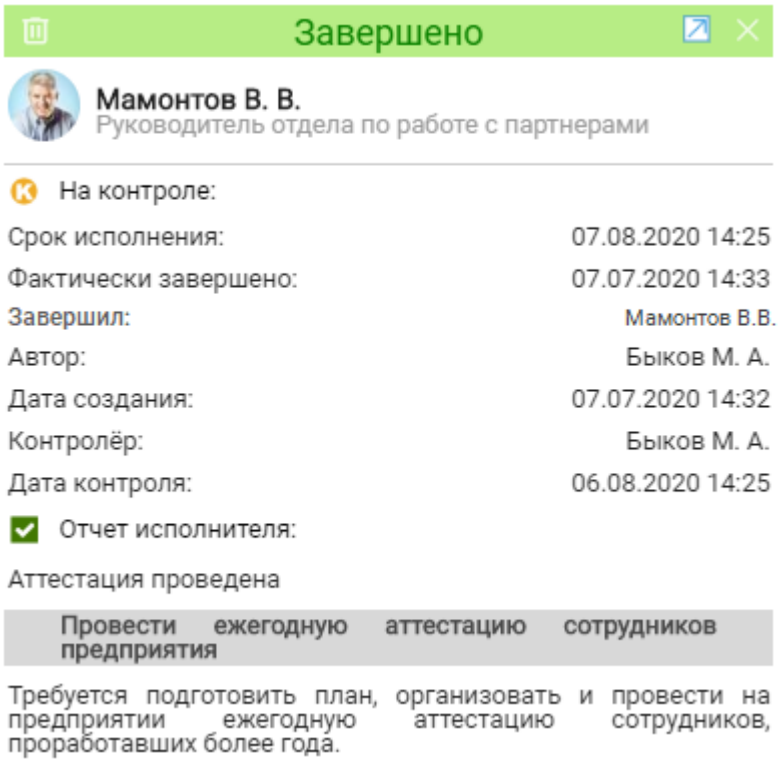


Рисунок 412. Пример справки для задания

Для группы заданий:

- Вариант исполнения с общим количеством исполнителей,
- Название,
- Состояние исполнения группы заданий,
- Список исполнителей,
- Количество выполненных заданий,
- Срок исполнения,
- Автор группы заданий,
- Контролёр и дата контроля,
- Содержание.

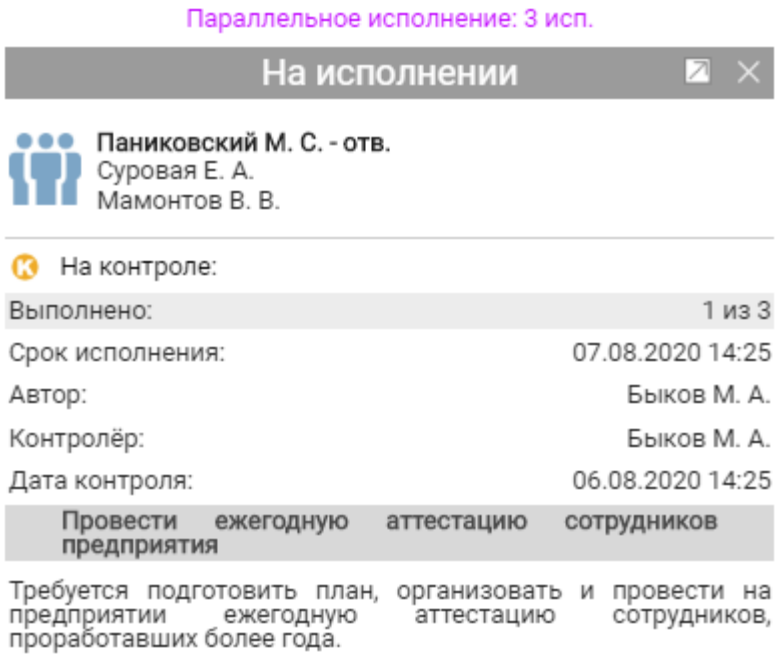


Рисунок 413. Пример справки для группы заданий

Из окна справки можно перейти к карточке данного задания или группы заданий. Чтобы перейти к карточке, нажмите кнопку .

Удаление задания или группы заданий

При наличии у пользователя прав на удаление задания или группы заданий в справке будет доступна кнопка . При нажатии кнопки данное задание или группа заданий будут удалены.

Предпросмотр файла

Компонент Предпросмотр файла позволяет просматривать содержимое файла карточки без его открытия во внешнем приложении и скачивания на устройство.

> По умолчанию предпросмотр документов в карточках отключен. Чтобы включить предпросмотр, необходимо настроить компоненты Р7-Офис Сервер документов/OnlyOffice.
>
> Максимальный размер файла для предпросмотра может быть ограничен настройками системы.

При предварительном просмотре некоторые файлы формата `.xlsm` могут не отображаться.

Если в карточке УД отображается предварительный просмотр файла, поля карточки автоматически становятся 5 px.

Область предварительного просмотра отображается слева от карточки.

Файлы сортируются по имени в алфавитном порядке. Предварительный просмотр будет отображаться для первого файла в списке.



Из режима предпросмотра карточку можно отправить на печать. При печати из браузеров Internet Explorer и Edge (до перехода на Chromium в версии 79) в конечном документе могут быть лишние листы и записи. Для корректной печати используйте более современные браузеры.

Web-клиент предоставляет два варианта предпросмотра файлов (определяется при настройке Web-клиента):

- В отдельном всплывающем окне:

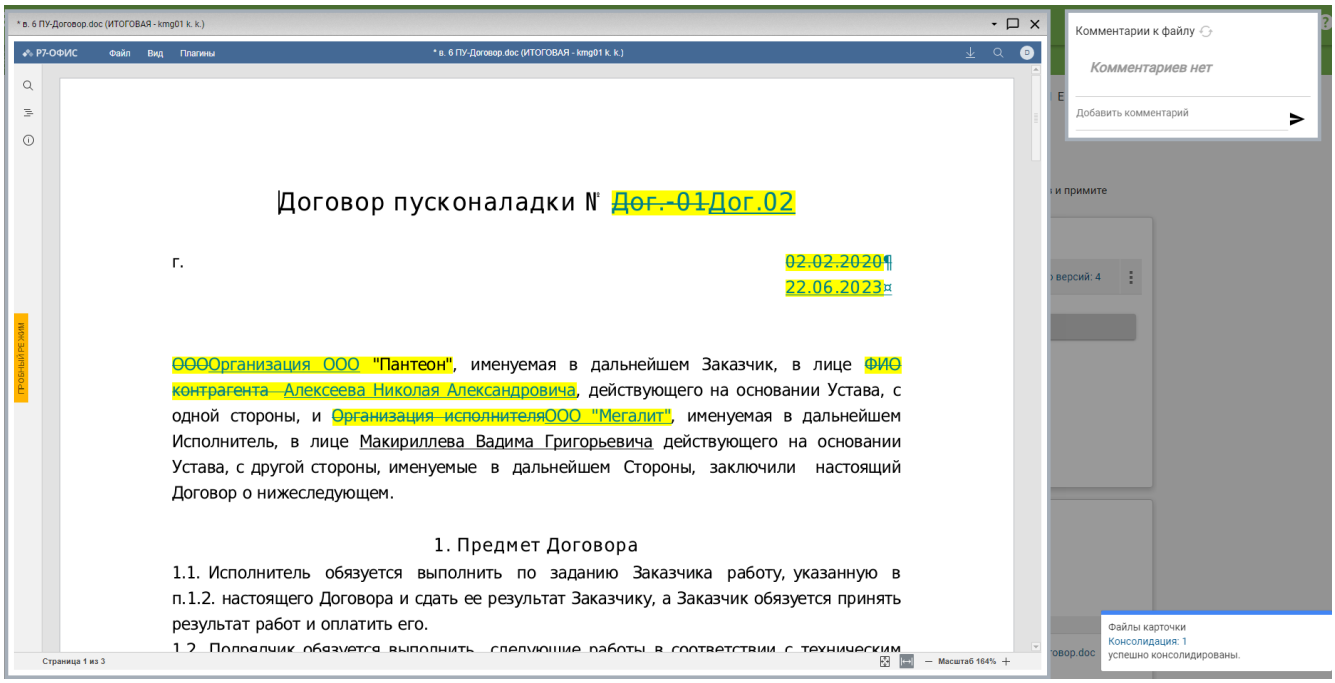


Рисунок 414. Предварительный просмотр файла в отдельном всплывающем окне

- Непосредственно в разметке карточки:

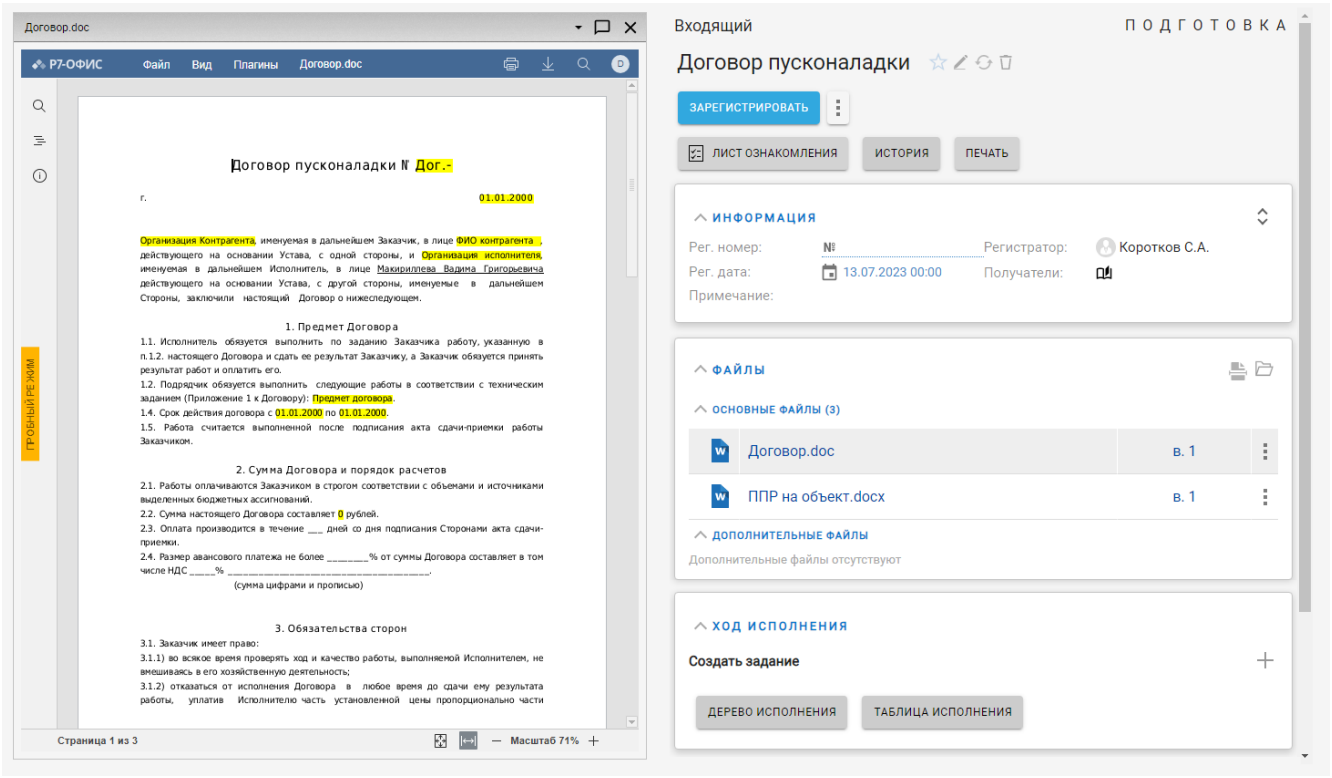


Рисунок 415. Предварительный просмотр в разметке карточки

Открытие предпросмотра файла

Чтобы открыть предварительный просмотр содержимого файла, выберите его в списке файлов карточки.

Для компонента Предпросмотр файла, встроенного в разметку, также доступна возможность выбора просматриваемого файла из списка в самом компоненте. При этом файл будет выделен в списке файлов карточки.

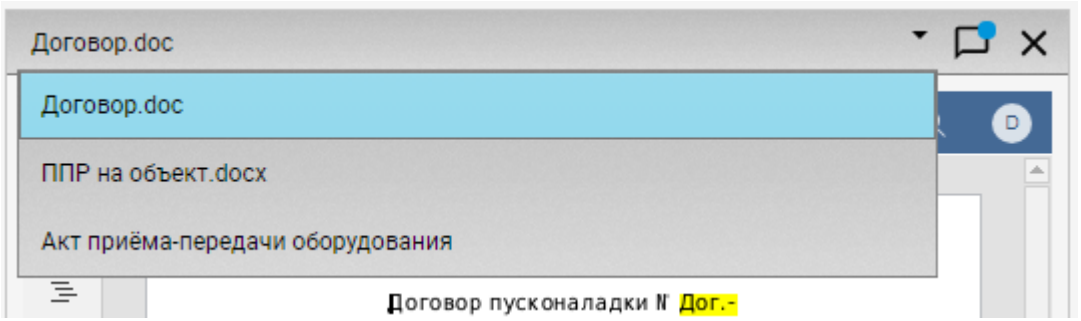


Рисунок 416. Выбор файла для предпросмотра из списка

Типы файлов, для которых поддерживается предварительный просмотр:

- .pdf (в т.ч PDF/A-1a).
- Офисные: .docx, .pptx, .odt, .doc, .rtf, .dot, .dotx, .xlt, .xls, .xltx, .xlsx, .ods;
- Изображения: .jpg, .jpeg, .gif, .png, .bmp, .tiff, .tif;

- Прочие: `.htm`, `.adoc`, `.js`, `.xslt`, `.xsl`, `.xml`, `.asmx`, `.ascx`, `.aspx`, `.cpp`, `.h`, `.css`, `.idl`, `.ini`, `.java`, `.rc`, `.rc2`, `.shtm`, `.shtml`, `.txt`, `.log`, `.cs`, `.sql`, `.csv`.



Предварительный просмотр файлов `.tiff` и `.tif` с разным размером страниц не поддерживается.

Если формат файла не поддерживается, будет предоставлена ссылка для его сохранения на компьютер пользователя.

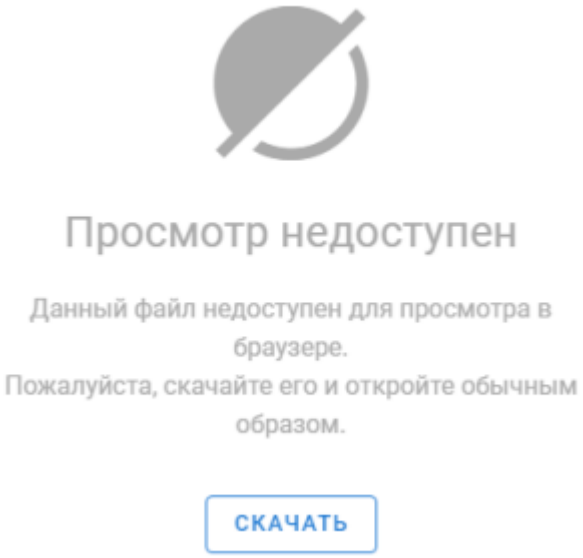


Рисунок 417. Предпросмотр неподдерживаемого типа файла

При предпросмотре текстовых файлов (с расширением `.txt`) отображаются только первые 100 000 символов. Если файл содержит большее количество символов, в предпросмотр будет добавлена строка `Полный предпросмотр данный файла недоступен по причине большого размера`. Чтобы посмотреть файл полностью, сохраните его на устройство и откройте в стандартном средстве просмотра.

Содержимое файла в области предпросмотра иногда может отображаться с некорректным форматированием. Например, отсутствующие на сервере Р7 шрифты могут незначительно отличаться от исходного документа, некоторые сложные элементы могут выглядеть иначе.

Комментирование

Если предпросмотр открывается в отдельном окне, область комментариев будет доступна сразу. Чтобы прокомментировать файл в компоненте `Предпросмотр файла`, встроенном в разметку, нажмите 💻, если комментариев нет и 💻 если комментарии есть. Закрыть блок комментирования можно повторным нажатием

кнопки или клавишей `Escape`.

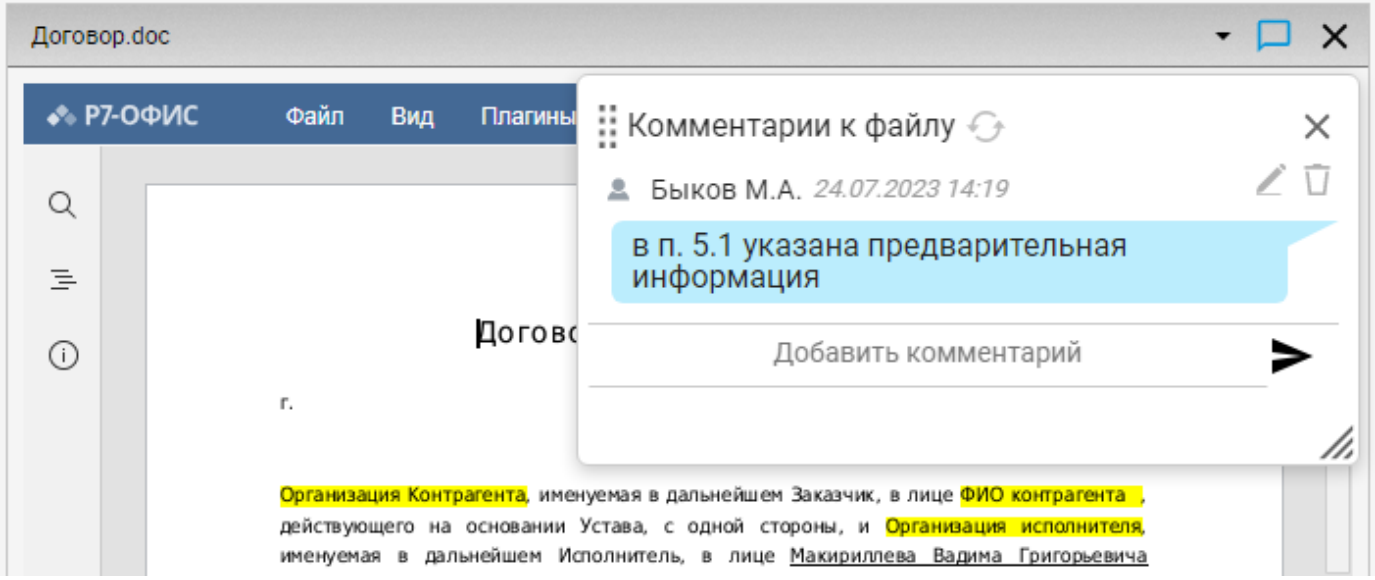


Рисунок 418. Комментирование файла

Данный блок комментариев можно перетаскивать с помощью мыши.

Информация по работе с компонентом комментариев приведена в пункте Комментарии.

Поиск по документу

Нажмите кнопку на панели инструментов и введите искомый текст в поле поиска.

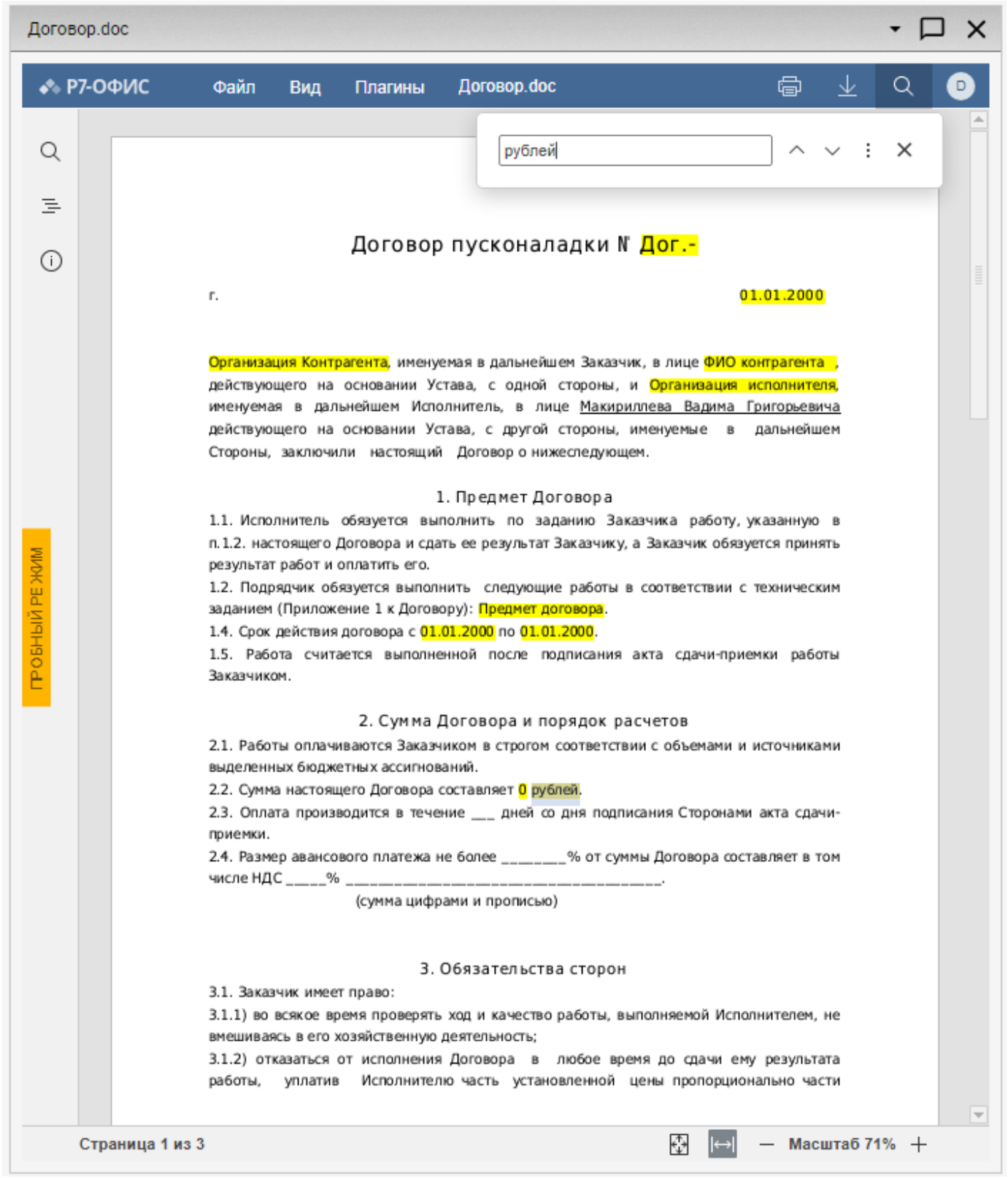


Рисунок 419. Поиск по документу в окне предпросмотра

Поиск возможен только при просмотре текстового документа.

Изменение масштаба

Для изменения масштаба документа используйте кнопки **-** / **+** или список справа от кнопок.

Для компонента Предпросмотр файла, встроенного в разметку, также доступна возможность расширения области предпросмотра. Чтобы расширить область предпросмотра, наведите курсор на границу предпросмотра и данных карточки так, чтобы изменилась форма курсора. Потяните разделитель вправо или влево, чтобы изменить ширину области предпросмотра.

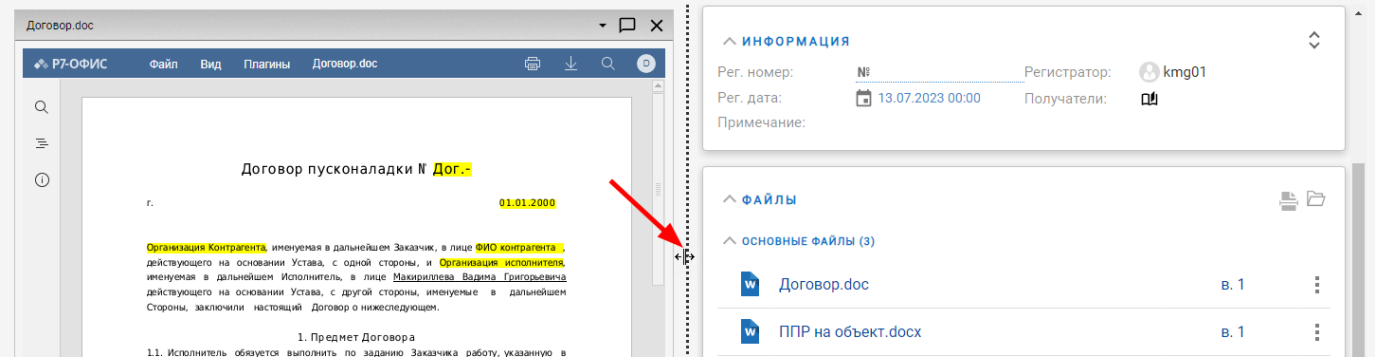


Рисунок 420. Изменение ширины области предпросмотра

Другие функции предпросмотра

Компонент предпросмотра предоставляет также другие функции: печать, сохранение документа и др.

Нажмите кнопку », чтобы открыть полный список доступных команд.



Из режима предпросмотра карточку можно отправить на печать. При печати из браузеров Internet Explorer и Edge (до перехода на Chromium в версии 79) в конечном документе могут быть лишние листы и записи. Для корректной печати используйте более современные браузеры.

Ссылка на карточку

Элемент управления Ссылка на карточку используется для добавления ссылки на карточку в другой карточке.

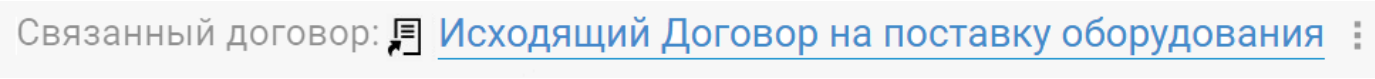


Рисунок 421. Ссылки

Строка ссылки содержит:

- Ссылку для открытия ссылочной карточки.
- Кнопку ⋮, открывающую меню операций со ссылкой.

При настройке Web-клиента может быть ограничен список типов карточек, на которые можно добавлять ссылки.

Добавление ссылки на карточку

1. Нажмите на текст *выберите карточку*, чтобы открыть окно выбора карточки для ссылки.

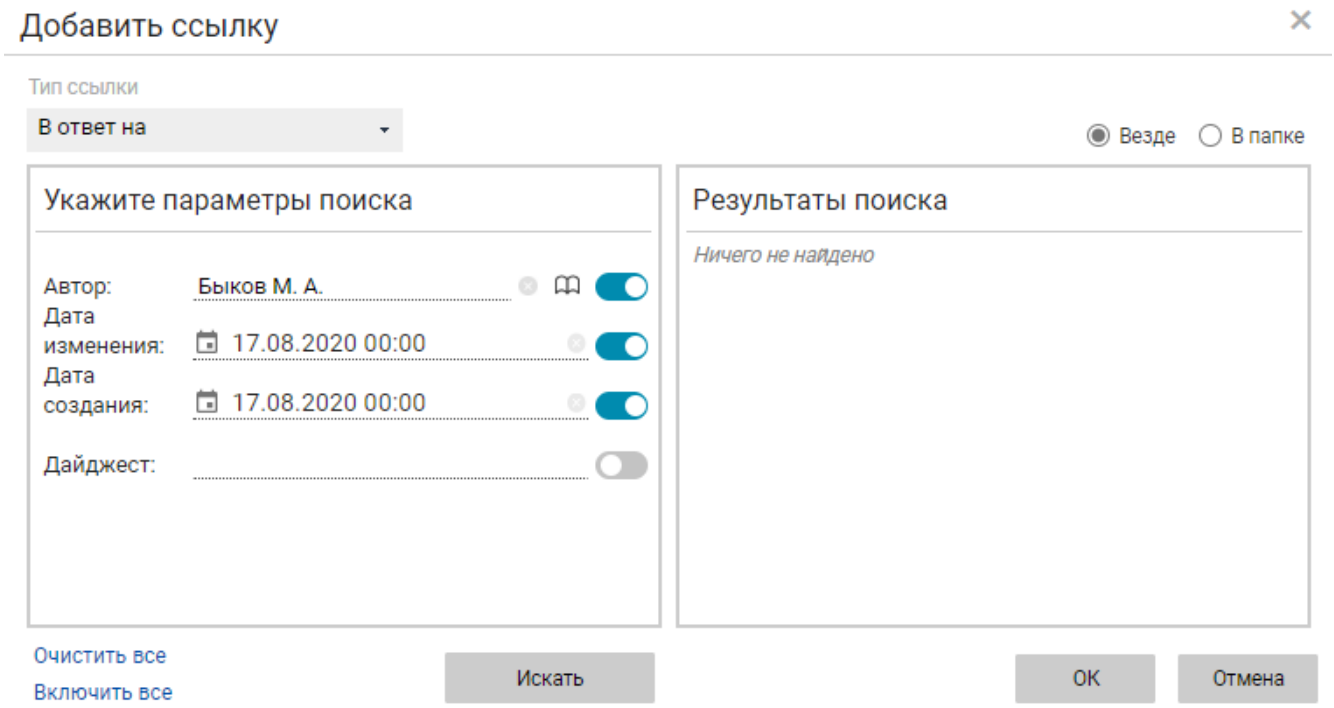


Рисунок 422. Выбор карточки для ссылки

2. Выберите карточку, на которую добавляется ссылка.
3. Нажмите **ОК**, чтобы добавить ссылку.

Выбрать карточку для ссылки

Карточку можно найти следующими способами:

- С помощью поиска.
- Выбрать из папки.
- Из быстрого поиска.

В мобильной версии Web-клиента доступна только возможность поиска карточки. Функция выбора из папки не предоставляется.

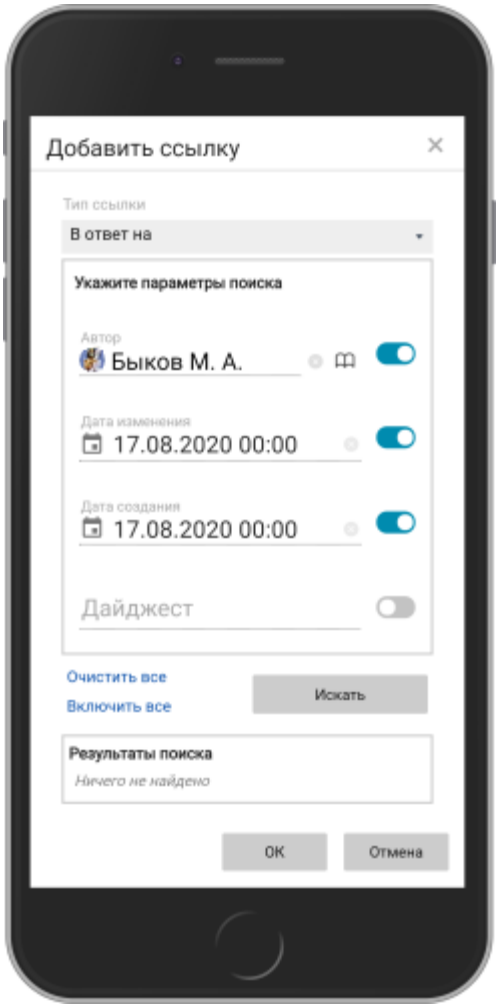


Рисунок 423. Добавление ссылки на существующую карточку в мобильной версии

С помощью поиска:

1. Установите переключатель в положение **Везде**.
2. Установите критерии поиска.

   > Список доступных критериев поиска соответствует поисковому запросу, установленному для элемента Ссылки при настройке Web-клиента.

3. Нажмите кнопку **Искать**, чтобы отобразить карточки, соответствующие параметрам запроса.

   > Если поисковый запрос не содержит параметров, список подходящих карточек будет выведен сразу при открытии окна выбора ссылки.

Выбрать из папки:

1. Установите переключатель в положение **В текущей папке**.
2. Выберите карточку из папки.



Пользователь может выбрать сразу несколько ссылок на карточки. Чтобы добавить несколько ссылок, удерживайте нажатой клавишу `Ctrl` на клавиатуре при выборе карточки. В мобильной версии используйте долгое касание (нажмите и удерживайте).

Список папок, из которых можно выбирать карточку для ссылки, устанавливается при настройке Web-клиента, а также может ограничиваться правами пользователя.

Папки с параметрическим поиском открываются без выполнения поискового запроса.

Выбранная карточка будет выделена жирным.



Ссылка может указывать на поисковый запрос. При поиске по карточке используется стандартный поисковый запрос *Ссылки*.

Из быстрого поиска:

1. Введите в поле имя карточки.
2. Выберите нужную карточку из раскрывающегося меню.

Открытие ссылки

Чтобы открыть карточку, на которую ведёт ссылка:

1. Нажмите на текст ссылки.
2. Нажмите ⋮ **> Открыть карточку**.

Просмотр файла карточки по ссылке

Нажмите ⋮ **> Просмотреть файл**, чтобы открыть основной файла карточки в режиме предпросмотра.

Изменение ссылки

Чтобы изменить ссылку на карточку, нажмите ⋮ **> Выбрать другую карточку**.

Удаление ссылки

Нажмите ⋮ **> Удалить ссылку**.



Удаляется только ссылка, но не карточка, на которую указывается ссылка.

Ссылки

Элемент управления Ссылки используется для добавления в карточку ссылок на другие карточки, файлы или веб-ресурсы.



Элемент управления всегда работает в режиме редактирования по месту — и в разметке редактирования, и в разметке просмотра. Это значит, что добавленная ссылка сохранится всегда, даже если изменения в карточке не были сохранены.

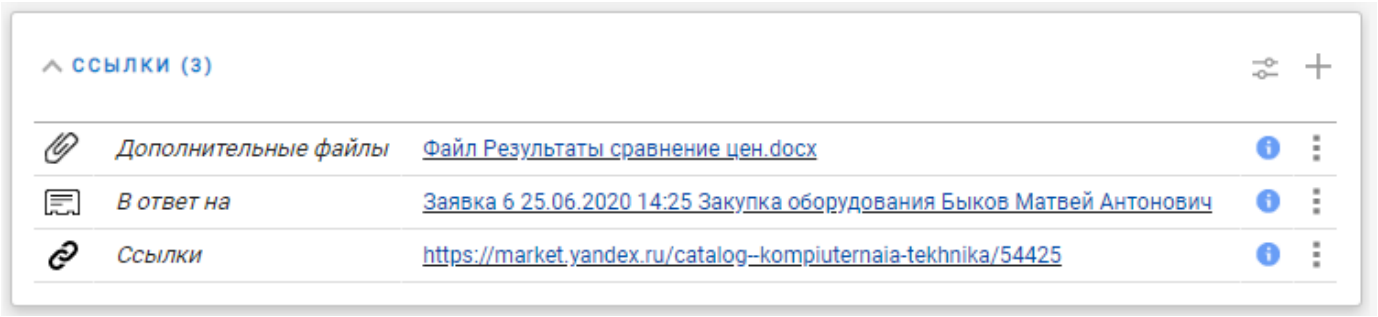


Рисунок 424. Ссылки

Строка ссылки содержит:

- Значок 📎/🗒/🔗.

  Обозначает назначение ссылки: ссылка на файл, карточку или веб-ресурс.

- Тип ссылки.
- Ссылка для открытия карточки, файла или веб-ресурса.
- Кнопка ⓘ.

  Показывает окно с информацией о ссылке.

- Кнопка ⋮.

  Показывает меню операций со ссылкой.



Возможность добавления ссылок на некоторые типы карточек может быть недоступна.

Добавление ссылки на существующую карточку

1. Нажмите + > , чтобы открыть окно выбора карточки для ссылки.

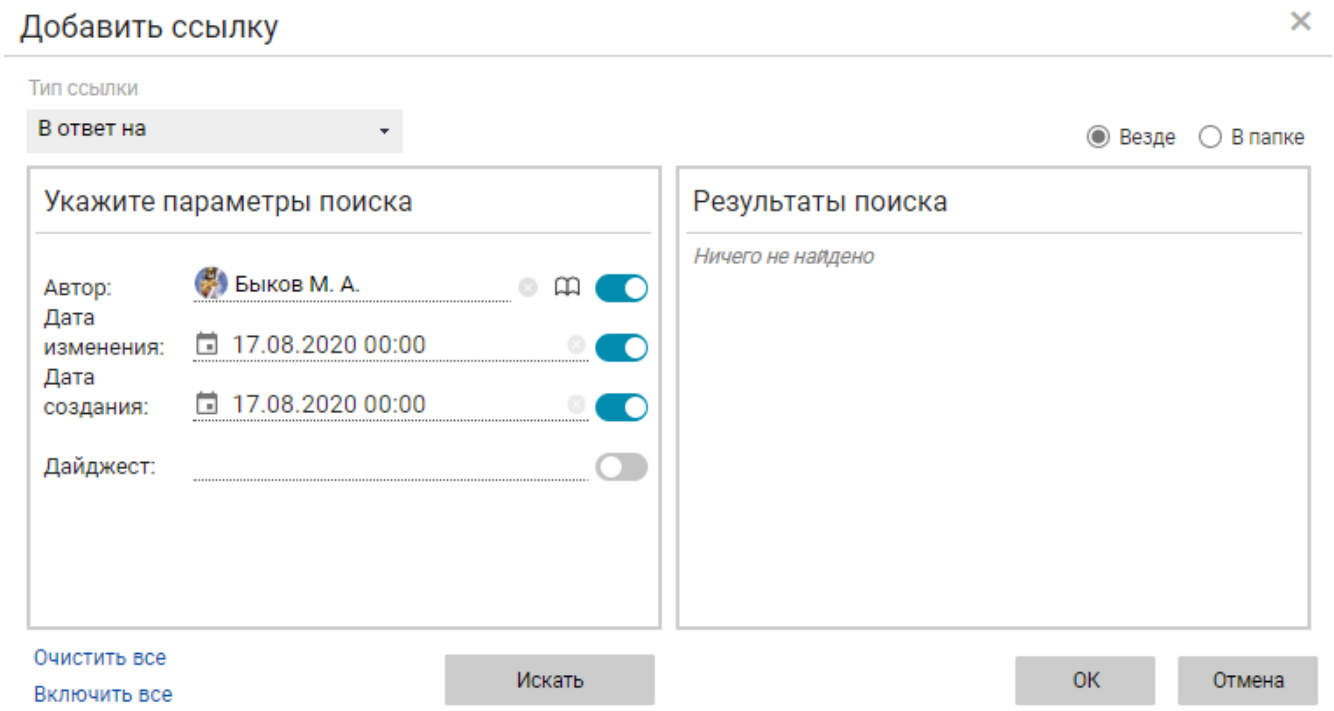


Рисунок 425. Добавление ссылки на существующую карточку

2. В списке *Тип ссылки* выберите тип для добавляемой ссылки.
3. Выберите карточку, на которую добавляется ссылка.

   Карточку можно найти следующими способами:

   - С помощью поиска.
   - Выбрать из папки.
   - Из быстрого поиска.

В мобильной версии Web-клиента доступна только возможность поиска карточки. Функция выбора из папки не предоставляется.

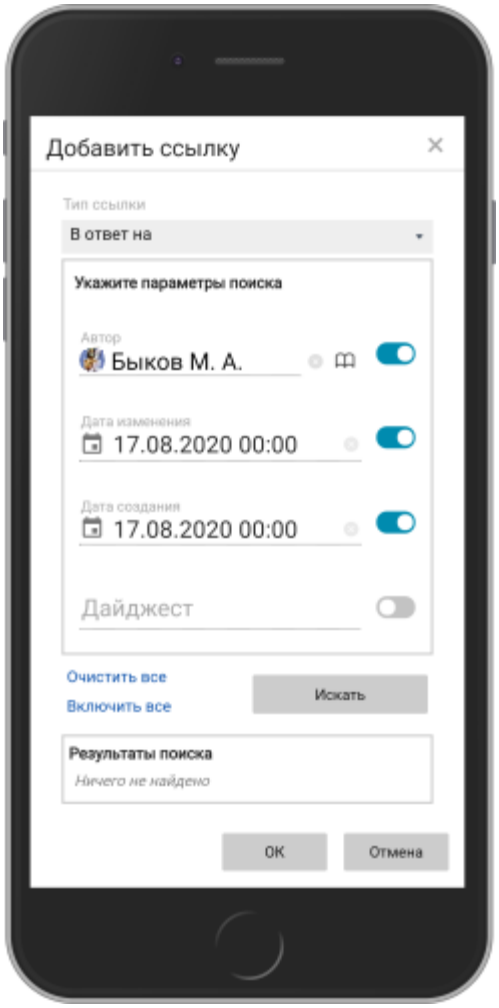


Рисунок 426. Добавление ссылки на существующую карточку в мобильной версии

С помощью поиска:

1. Установите переключатель в положение **Везде**.
2. Установите критерии поиска.

   > Список доступных критериев поиска соответствует поисковому запросу, установленному для элемента Ссылки при настройке Web-клиента.

3. Нажмите кнопку **Искать**, чтобы отобразить карточки, соответствующие параметрам запроса.

   > Если поисковый запрос не содержит параметров, список подходящих карточек будет выведен сразу при открытии окна выбора ссылки.

Выбрать из папки:

1. Установите переключатель в положение **В текущей папке**.
2. Выберите карточку из папки.



Пользователь может выбрать сразу несколько ссылок на карточки. Чтобы добавить несколько ссылок, удерживайте нажатой клавишу `Ctrl` на клавиатуре при выборе карточки. В мобильной версии используйте долгое касание (нажмите и удерживайте).

Список папок, из которых можно выбирать карточку для ссылки, устанавливается при настройке Web-клиента, а также может ограничиваться правами пользователя.

Папки с параметрическим поиском открываются без выполнения поискового запроса.

Выбранная карточка будет выделена жирным.

Добавление ссылки на новую карточку

1. Нажмите + > , чтобы открыть окно настройки ссылки.

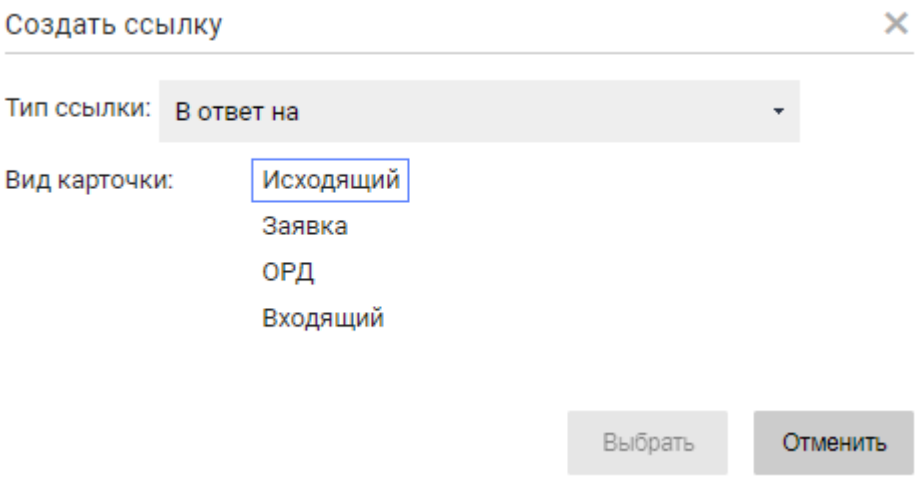


Рисунок 427. Добавления ссылки с созданием карточки

2. Выберите тип ссылки и вид создаваемой карточки, создаваемой по ссылке.
3. Нажмите кнопку **Выбрать**. Будет открыта страница создания карточки, на которую добавляется ссылка. В основную карточку ссылка добавится после сохранения новой карточки.

При работе с элементом Ссылки в новой карточке и при редактировании существующей карточки функция добавления ссылки на новую карточку недоступна.

Добавление ссылки на файл

1. Нажмите + > 📎, чтобы открыть окно выбора файла.
2. Выберите файл из файловой системе.
3. Нажмите кнопку **Открыть**, чтобы добавить ссылку на карточку выбранного файла.

Добавленные через ЭУ ссылки на файлы сразу сохраняются на сервер. При отказе от сохранения карточки (нажатии кнопки **Отменить**) ссылка на файл остаётся в карточке, так как уже была сохранена на сервере. Отменяются только те данные, которые не были отправлены на сервер.

Добавление ссылки на веб-ресурс

1. Нажмите + > 🔗, чтобы открыть окно добавления ссылки.



Рисунок 428. Добавление ссылки на веб-ресурс

2. Выберите тип ссылки и введите URL-адрес ресурса, например, http://www.Docsvision.com.
3. Нажмите кнопку **Подтвердить**, чтобы сохранить ссылку.

Открытие ссылки

Для перехода по ссылке щелкните по её названию. В зависимости от вида ссылки будет открыта карточка файл в режиме предпросмотра или веб-ресурс.

Просмотр информации о ссылке

Для просмотра информации о ссылке нажмите кнопку ℹ.

Удаление ссылки

Нажмите ⋮ > **Удалить** в строке удаляемой ссылки. При удалении ссылки на файл можно удалить только ссылку или ссылку и файл.

Настройка столбцов таблицы ссылок

Для настройки таблицы ссылок нажмите кнопку ⚙, чтобы открыть окно с настройками столбцов.

> Кнопка настройки будет недоступна, если возможность настройки отключена.

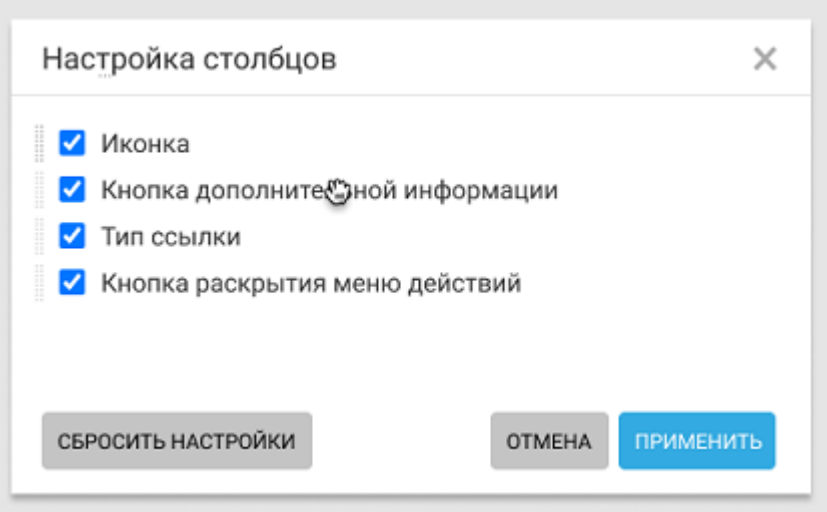


Рисунок 429. Настройки столбцов таблицы ссылок

Снимите флаги с колонок, которые нужно скрыть. Чтобы изменить порядок столбцов, перетащите строки при помощи курсора в полной версии или долгого нажатия в мобильной версии. Чтобы отменить собственные настройки, нажмите кнопку **Сбросить настройки**.

Выполнение групповой операции

Чтобы выполнить одну операцию в нескольких карточках:

1. Нажмите кнопку ☑ в заголовке раздела *Ссылки*.

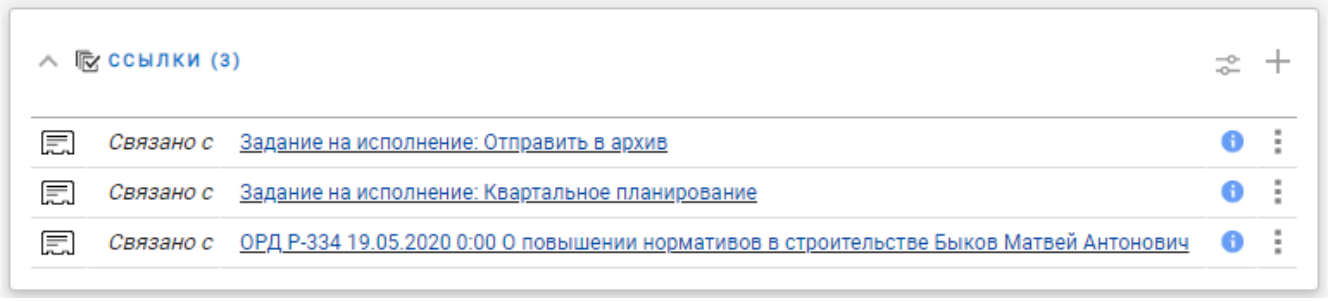


Рисунок 430. Список ссылок с доступным режимом групповых операций

Кнопка будет недоступна, если возможность выполнения групповых операций в ссылках отключена.

Таблица ссылок переключится в режим групповых операций.

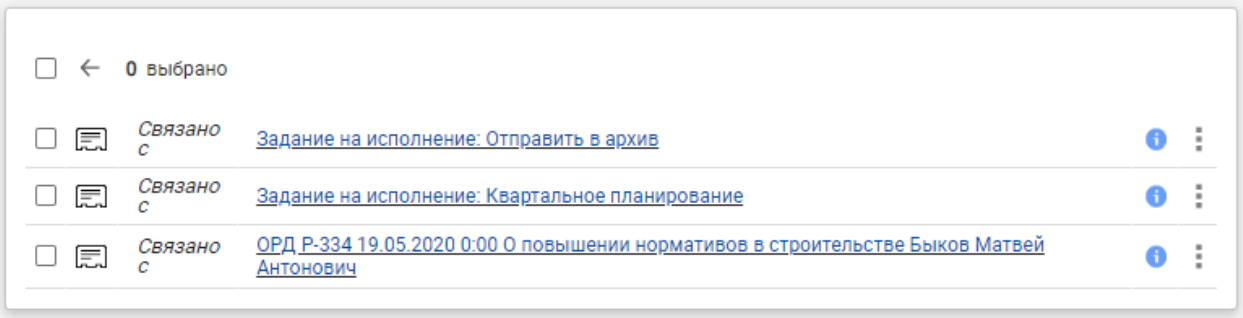


Рисунок 431. Список ссылок в режиме групповых операций

Чтобы выйти из режима групповых операций, нажмите ←.

2. Отметьте ссылки на карточки, в которых нужно выполнить групповую операцию.

   Если у выбранных карточек есть общие групповые операции, они отобразятся над таблицей ссылок. Если доступных групповых операций нет, появится сообщение Для выбранных карточек не предусмотрены групповые операции.

3. Нажмите на кнопку выполнения требуемой групповой операции и дождитесь её завершения.

Изображение

Элемент Изображение предназначен для отображения изображения в карточке, а также добавления изображения (должно быть разрешено при настройке Web-клиента).



Рисунок 432. Изображение

Следующие операции будут доступны, только если при настройке

> Web-клиента разрешено добавление изображений и пользователю разрешена операция редактирования.

Добавление изображения

Изображение может быть добавлено двумя способами:

- Нажмите на элемент Изображение и выберите файл формата `.jpg` или `.jpeg`.
- Перетащите файл формата `.jpg` или `.jpeg` из файловой системы в элемент Изображение.

Удаление изображения.

Чтобы удалить изображение, нажмите на кнопку ✕ в элементе управления. Кнопка отображается при наведении курсора мыши на элемент Изображение.

Файл

Элемент Файл предназначен для работы с файлами карточек. В отличие от элемента Список файлов данный элемент позволяет работать с файлами любых типов карточек, но не разделяет файлы на основные и дополнительные, а также не предоставляет функций для работы с электронными подписями.

Интерфейс элемента зависит от наличия загруженного файла, способа добавления файла и его настроек:

- Элемент содержит настроенную текстовую метку, если файл не добавлен:

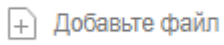


Рисунок 433. Текстовая метка "Добавьте файл"

- Когда файл добавляется методом перетаскивания:

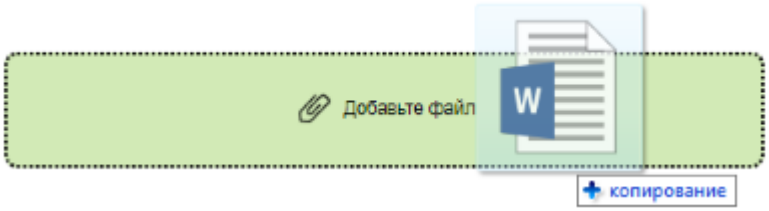


Рисунок 434. Файл добавляется перетаскиванием

- Когда файл добавлен:

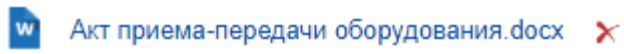


Рисунок 435. Файл в списке

После сохранения карточки, возможна работа с версиями файла. В этом случае в интерфейс добавляется номер версии.

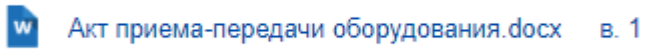


Рисунок 436. Файл с номером версии

Далее перечислены функции, предоставляемые элементом Файл. Некоторые из указанных функций будут недоступны, если в настройках элемента отключена поддержка работы с версиями файла или у пользователя отсутствуют права на выполнения операции.

Добавление файлов

Добавить файл можно двумя способами:

- Перетащите файл из файловой системы в область элемента Файл.
- Когда файл не добавлен, нажмите на элемент и выберите файл с помощью Проводника.

Условия добавления файлов:

- Типы файлов, которые можно добавлять: .docx, .pdf (в т.ч PDF/A-1a), .rtf, .dot, .doc, .dotm, .docm, .dotx, .ods, .odt, .csv, .xlt, .xls, .xltm, .xlsm, .xltx, .xlsx, .jpg, .jpeg, .gif, .png, .bmp, .tiff, .tif, .emf, .wmf, .xslt, .xsl, .xml, .htm, .html, .txt, .log, .cs, .asmx, .ascx, .aspx, .js, .cpp, .h, .css, .idl, .ini, .java, .rc, .rc2, .shtm, .shtml, .sql, .zip, .rar, .7z, .vsd, .vsdx, .msg, .one, .pptx, .ppt, .avi, .wav, .pfx.

  Некоторые файлы формата .xlsm могут не отображаться в области предварительного просмотра.



Хранение букв ё и й в операционных системах MacOS и Windows различается. Если вы планируете работать с файлами в обеих системах, рекомендуется не использовать эти буквы в названиях файлов.

Длина имени файла в любой ОС не должна превышать 255 байт.

Удаление файлов

Чтобы удалить файл:

- Из создаваемой карточки, нажмите кнопку ✕.
- Из существующей карточки, нажмите на название файла и выберите **Удалить**. Удаляемый файл, а также его карточка должны быть разблокированы.



При удалении будут удалены все версии данного файла.

Редактирование файла

Чтобы открыть файл в связанной программе, нажмите на название файла и нажмите **Открыть**. Данная возможность доступна для следующих файлов: `.doc`, `.docx`, `.docm`, `.xls`, `.xlsx`, `.xlsm`, `.ppt`, `.pptx`, `.odt` и `.ods`. Функциональность доступна только при установленном соответствующем ПО и DVWebTool/DVSupService. С добавленными дополнительными форматами функциональность может работать некорректно.

Открытый для редактирования файл будет заблокирован от изменений со стороны других пользователей.

Если в настройках элемента управления активно свойство `Включить версионинг`, при сохранении файла с изменениями будет создана новая версия.

Условия редактирования файла:

До завершения редактирования файла не закрывайте карточку, из которой он был открыт.

Предварительный просмотр



По умолчанию предпросмотр документов в карточках отключен. Чтобы включить предпросмотр, необходимо настроить компоненты Р7-Офис Сервер документов/OnlyOffice.

Для открытия файла в режиме предпросмотра щелкните по названию файла и нажмите **Предварительный просмотр**.

Типы файлов, для которых поддерживается предварительный просмотр:

- `.pdf` (в т.ч `PDF/A-1a`).

- Офисные: .docx, .pptx, .odt, .doc, .rtf, .dot, .dotx, .xlt, .xls, .xltx, .xlsx, .ods;
- Изображения: .jpg, .jpeg, .gif, .png, .bmp, .tiff, .tif;
- Прочие: .htm, .adoc, .js, .xslt, .xsl, .xml, .asmx, .ascx, .aspx, .cpp, .h, .css, .idl, .ini, .java, .rc, .rc2, .shtm, .shtml, .txt, .log, .cs, .sql, .csv.

Инструкция по работе с компонентом предпросмотра приведена в пункте Предпросмотр файла.

Скачивание файла

Чтобы скачать файл на устройство, нажмите на название файла и выберите **Скачать**.

Замена файла

Текущий файл может быть заменён на другой, соответствующий требованиям для добавления файла. Если в настройках элемента управления включено свойство версионинг, будет заменена текущая версия. Чтобы заменить файл, нажмите на название файла, затем **Заменить** и выберите новый файл из хранилища устройства.

Загрузка новой версии файла

Чтобы загрузить новую версию файла, нажмите на название файла, затем **Загрузить версию** и выберите новую версию файла из хранилища устройства.



Тип файла должен совпадать с типом файла, версия которого добавляется.

Возможность создания новой версии доступна, только если в настройках элемента управления Файл включено свойство версионинг.

После загрузки версии номер текущей версии увеличится.

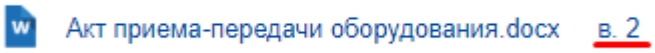


Рисунок 437. Файл с новой версией

Просмотр версий файла

Текущая версия файла отображается справа от его названия. Чтобы ознакомиться с другими версиями файла, нажмите на номер текущей версии (на рисунке выше — в. 2).

Будет открыто диалоговое окно со списком версий.

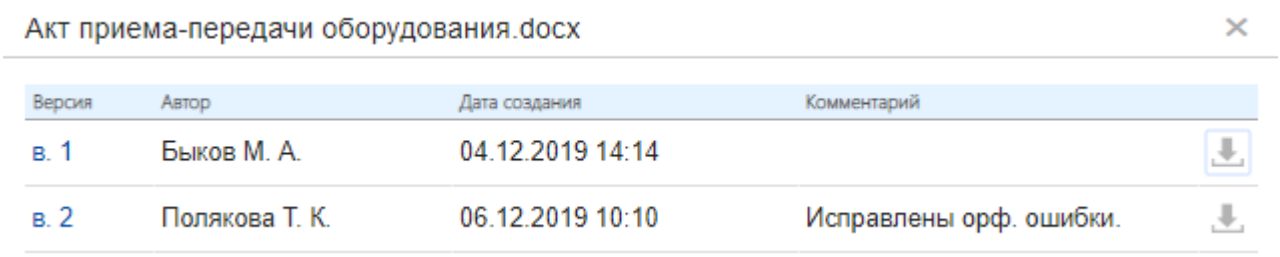


Рисунок 438. Список версий файла

Чтобы открыть версии файла в режиме предварительного просмотра, нажмите на номер версии.

Чтобы сохранить версию на устройство, нажмите кнопку .

Добавление комментария к версии

Для добавления комментария к текущей версии щелкните по названию файла и нажмите **Добавить комментарий**. Будет открыто окно комментирования версии.

В окне отображаются комментарии пользователей для текущей версии.

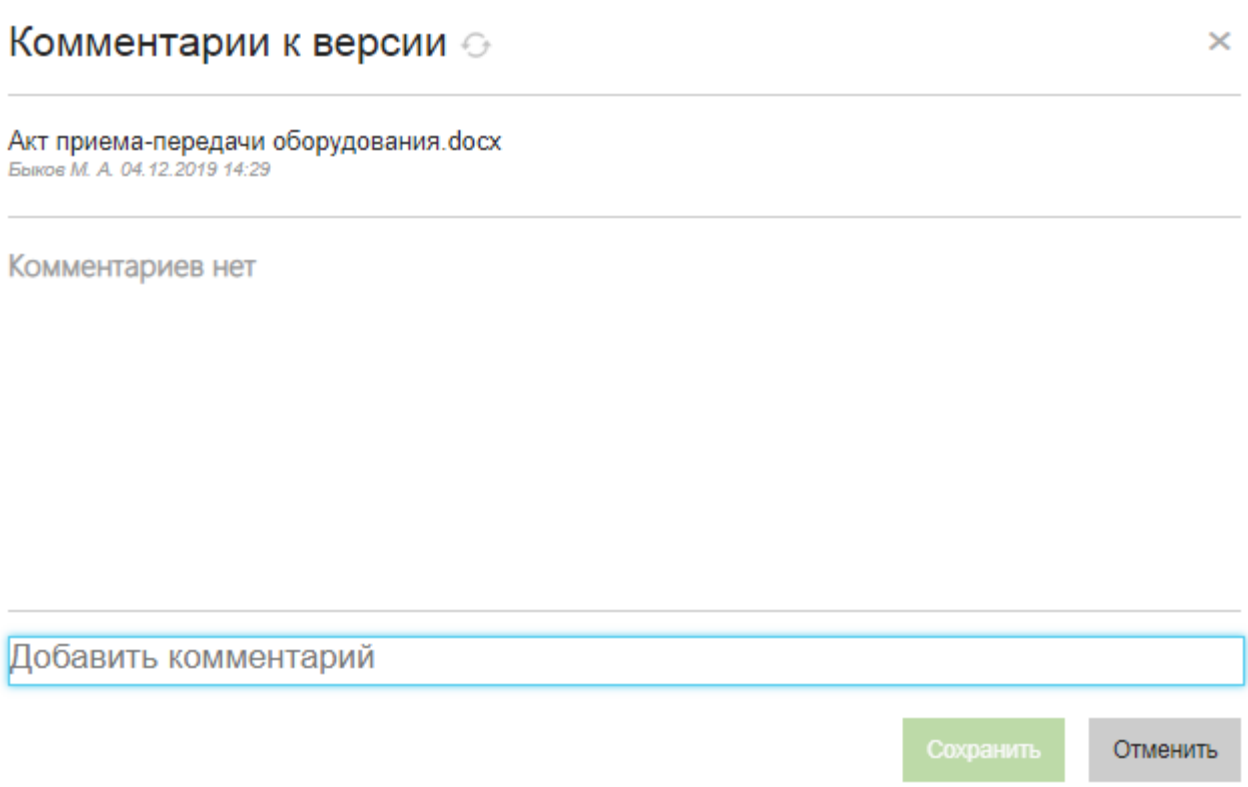


Рисунок 439. Комментарии к версии

Введите комментарий в строку ввода и нажмите кнопку **Сохранить**.

Добавленные комментарии отображаются в окне просмотра версий файла.

Список файлов

Для работы с коллекцией файлов карточки используется элемент Файл, встроенный в элемент Таблица.

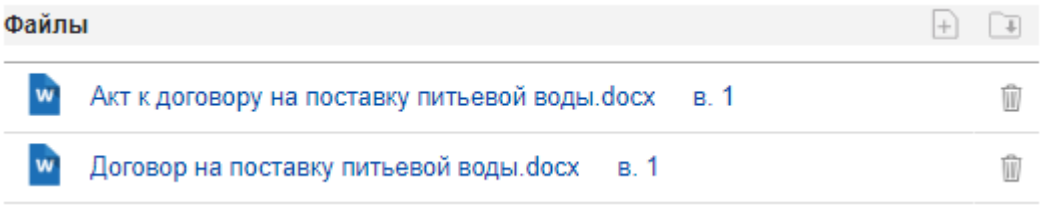


Рисунок 440. Таблица для работы с файлами

Добавление файлов

Добавить файл можно двумя способами:

- Перетащите файлы из файловой системы в область элемента Таблица.
- Нажмите кнопку и выберите файлы с помощью Проводника.

Условия добавления файлов аналогичны условиям добавления одиночного

файла, приведенным выше.

Скачивание всех файлов

Чтобы скачать все файлы, которые отображаются в списке, нажмите кнопку .

Удаление строки с файлом из списка

Чтобы удалить строку из таблицы, нажмите кнопку  в строке с файлом.



При этом будет удалена вся строка и файл, который был указан в строке.

При удалении файла с помощью команды **Удалить** контекстного меню файла будет удалён только файл, строка из таблицы не удаляется.

Операции с файлами выполняются аналогично рассмотренным выше для одиночных файлов.

Просмотр содержимого .zip архивов

Элемент управления позволяет просматривать содержимое добавленных .zip-архивов. Если в элемент управления приложен файл с расширением `.zip`, в контекстном меню появляется пункт *Просмотр состава архива*. При выборе данного пункта открывается окно, в котором на каждой строке будут перечислены входящие в состав архива файлы с расширением и папки.

Предпросмотр файлов из архива недоступен. Комментирование и скачивание архива работает по аналогии с обычным файлом.

Просмотр файлов в стороннем ПО

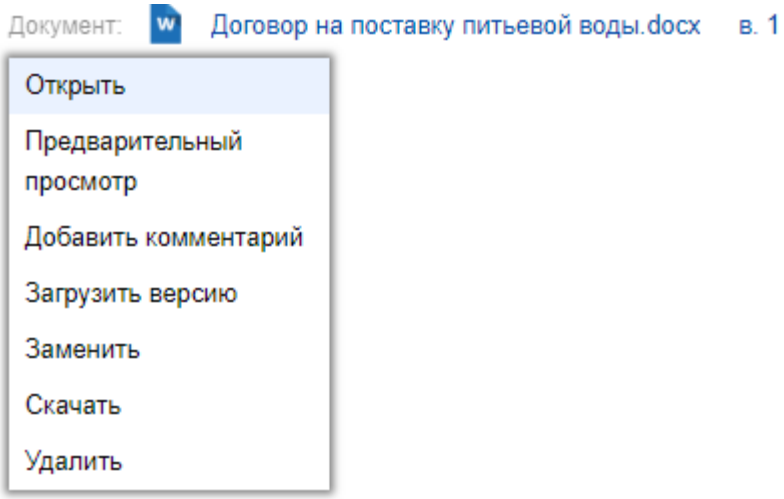


Рисунок 441. Открытие файла на просмотр

Чтобы открыть файл на просмотр в стороннем редакторе:

1. Убедитесь, что на машине сотрудника установлена программа DVWebTool или DVSupService в зависимости от используемой операционной системы, при отсутствии будет выведено сообщение о необходимости установить.
2. В строке файла выберите пункт *Открыть*.
3. Файл будет открыт в текстовом редакторе по умолчанию.
4. После сохранения и закрытия редактора новая версия файла будет добавлена в карточку.

Лист ознакомления

Элемент управления отображает лист ознакомления с документом.

Чтобы открыть лист ознакомления, нажмите на кнопку в ленте карточки:

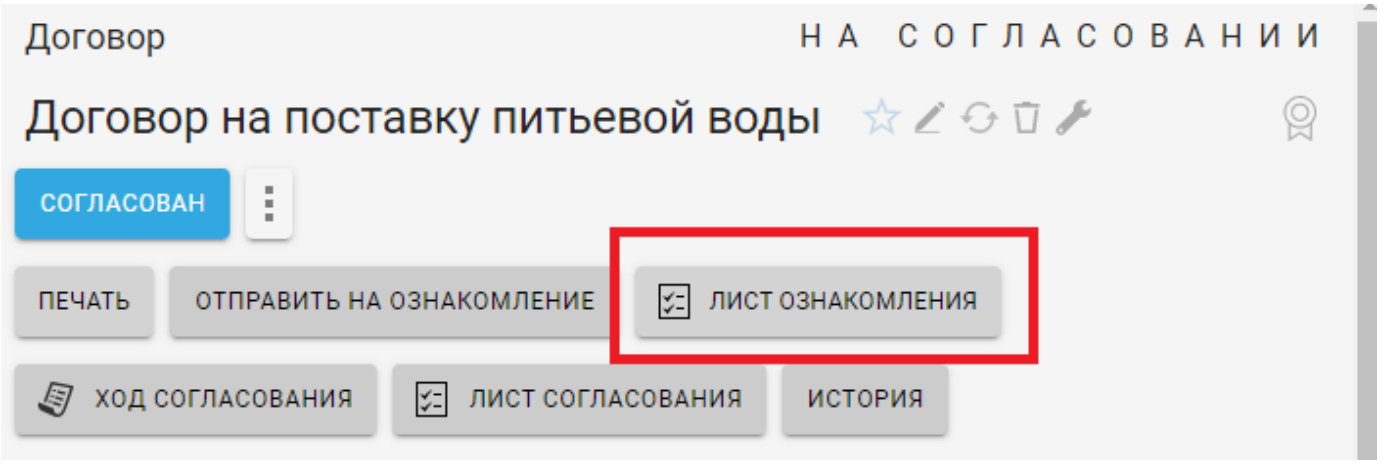
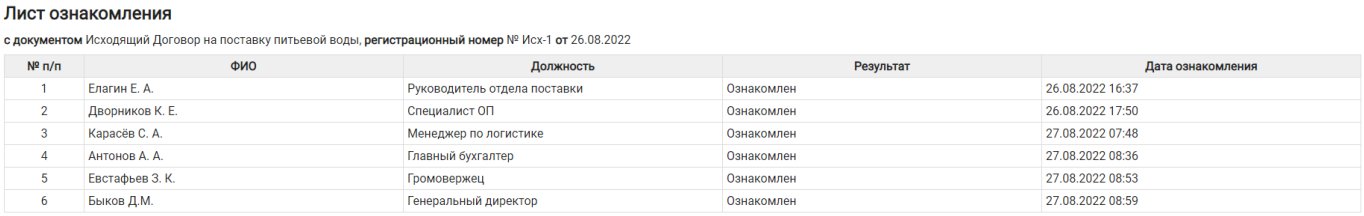


Рисунок 442. Лист ознакомления в разметке карточки

При работе с карточкой типа "Документ", содержащей этот ЭУ, пользователь получает возможность:

1. Настраивать выводимый лист ознакомления.
2. Выводить лист ознакомления с документом на отдельную вкладку в браузере.
3. Распечатывать лист ознакомления средствами браузера.



Лист ознакомления

с документом Исходящий Договор на поставку питьевой воды, **регистрационный номер** № Исх-1 **от** 26.08.2022

№ п/п	ФИО	Должность	Результат	Дата ознакомления
1	Елагин Е. А.	Руководитель отдела поставки	Ознакомлен	26.08.2022 16:37
2	Дворников К. Е.	Специалист ОП	Ознакомлен	26.08.2022 17:50
3	Карасёв С. А.	Менеджер по логистике	Ознакомлен	27.08.2022 07:48
4	Антонов А. А.	Главный бухгалтер	Ознакомлен	27.08.2022 08:36
5	Евстафьев З. К.	Громовержец	Ознакомлен	27.08.2022 08:53
6	Быков Д.М.	Генеральный директор	Ознакомлен	27.08.2022 08:59

Рисунок 443. Открытый лист ознакомления

Элемент может отображать только:

- Непосредственные подчинённые задания и их подзадания.
- Задания подчинённых групп заданий.

Документы

Работа в Web-клиенте

Документы

Дополнительные разделы документации, содержащие информацию, полезную при работе с Web-клиентом Docsvision.

- Нумератор
- Список файлов
- Дело
- Больше информации в навигационном меню слева…

Нумератор

Нумератор используется для получения регистрационного номера документа.



Рисунок 444. Нумератор

Получение регистрационного номера

Нажмите кнопку №, чтобы автоматически получить номер.

Самостоятельный ввод номер

Если при настройке Web-клиента было разрешено вводить номер вручную, нажмите на область, в которой отображается номер.

В зависимости от настроек элемент переключится в режим ввода номера или откроется блок для ввода номера.

Удаление номера

Чтобы освободить занятый номер, нажмите кнопку .

Номер станет доступен для следующего получения.

Список файлов

Элемент управления Список файлов предназначен для работы с файлами карточек *Документ*.

ЭУ "Список файлов" предоставляет следующие функции работы с файлами:

- Добавлять файлы,
- Добавлять файлы со сканера,
- Удалять файлы,
- Просматривать файлы (ограничено),
- Сортировать файлы,
- Редактировать файлы (ограничено),
- Загружать новые версии файлов,
- Переименовывать файлы,
- Подписывать файлы,
- Проверять подписи файлов,
- Импортировать и экспортировать электронные подписи,
- Экспортировать файлы с подписью.



Добавленные через ЭУ файлы сразу сохраняются на сервер. При отказе от сохранения карточки (нажатии кнопки **Отменить**) файл остаётся в карточке, так как уже был сохранён на сервере. Отменяются только те данные, которые не были отправлены на сервер.

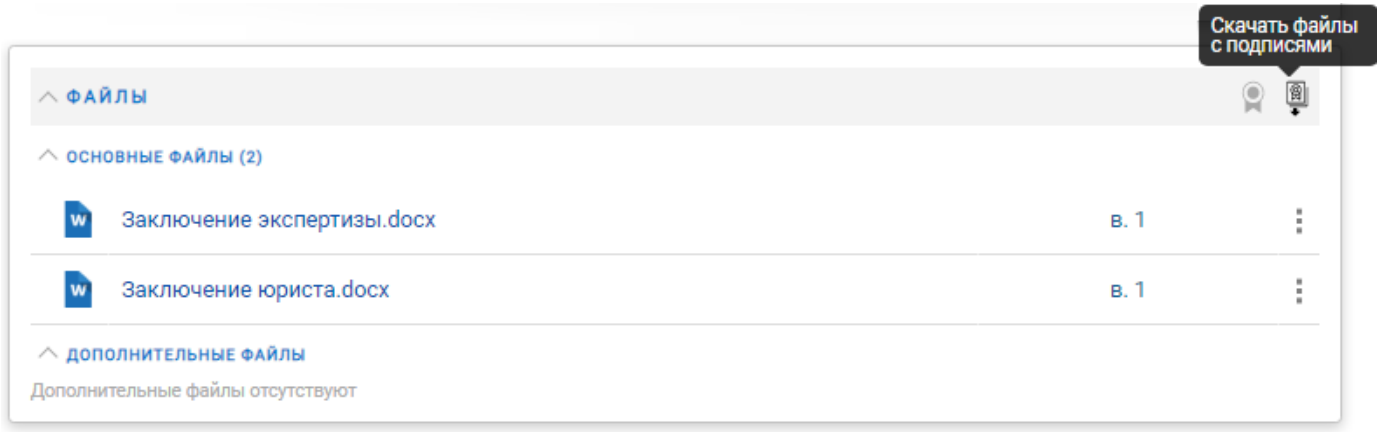


Рисунок 445. Список файлов

Условия для добавления файлов

Типы файлов, которые можно добавлять

.docx, .pdf (в т.ч PDF/A-1a), .rtf, .dot, .doc, .dotm, .docm, .dotx, .ods, .odt, .csv, .xlt, .xls, .xltm, .xlsm, .xltx, .xlsx, .jpg, .jpeg, .gif, .png, .bmp, .tiff, .tif, .emf, .wmf, .xslt, .xsl, .xml, .htm, .html, .txt, .log, .cs, .asmx, .ascx, .aspx, .js, .cpp, .h, .css, .idl, .ini, .java, .rc, .rc2, .shtm, .shtml, .sql, .zip, .rar, .7z, .vsd, .vsdx, .msg, .one, .pptx, .ppt, .avi, .wav, .p7s, .sig, .sgn, .sign, .pfx.

Список может быть изменён администратором Web-клиента.

Суммарный объем файлов, добавляемых в карточку *Документ* на этапе создания карточки, не должен превышать 128 Мб. Файл сверх этого размера можно добавить из режима чтения карточки (см. далее). Администратор может установить собственные ограничение на максимальный размер прикладываемых файлов.

Хранение букв ё и й в операционных системах MacOS и Windows различается. Если вы планируете работать с файлами в обеих системах, рекомендуется не использовать эти буквы в названиях файлов.

Длина имени файла в любой ОС не должна превышать 255 байт.

Добавление дополнительных файлов может быть запрещено при настройке Web-клиента.

Сканирование файлов доступно, только если функция разрешена

при настройке Web-клиента.

Добавление файлов из файловой системы

Элемент управления предоставляет несколько способов добавления файлов с файловой системы:

1. Щелкните по области *Прикрепить файлы*.

   Будет предложено выбрать файлы.

2. Нажмите кнопку и щелкните по типу добавляемого файла: *основной* или *дополнительный*.
3. Выберите файлы для добавления.

   Если добавление дополнительных файлов запрещено, вместо выбора типа файла будет сразу открыт диалог добавления.

   Файлы, содержащие в названии символы `#`, `%`, `+`, `&`, при добавлении будут автоматически переименованы, указанные символы будут заменены на `_`.

4. Перетяните файл(ы) из файловой системы в область *Прикрепить файлы*.

   Элемент `Список файлов` трансформируется:



Рисунок 446. Элемент "Список файлов" при разрешённом добавлении дополнительных файлов.

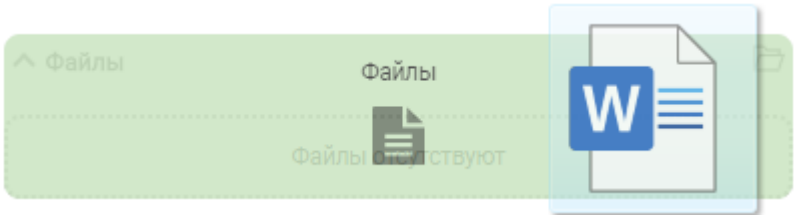


Рисунок 447. Элемент "Список файлов" при запрещённом добавлении дополнительных файлов

Перетащите файл(ы) в раздел Основные файлы или, если разрешено, в Дополнительные файлы.

После добавления файла или файлов элемент Список файлов примет вид списка файлов:

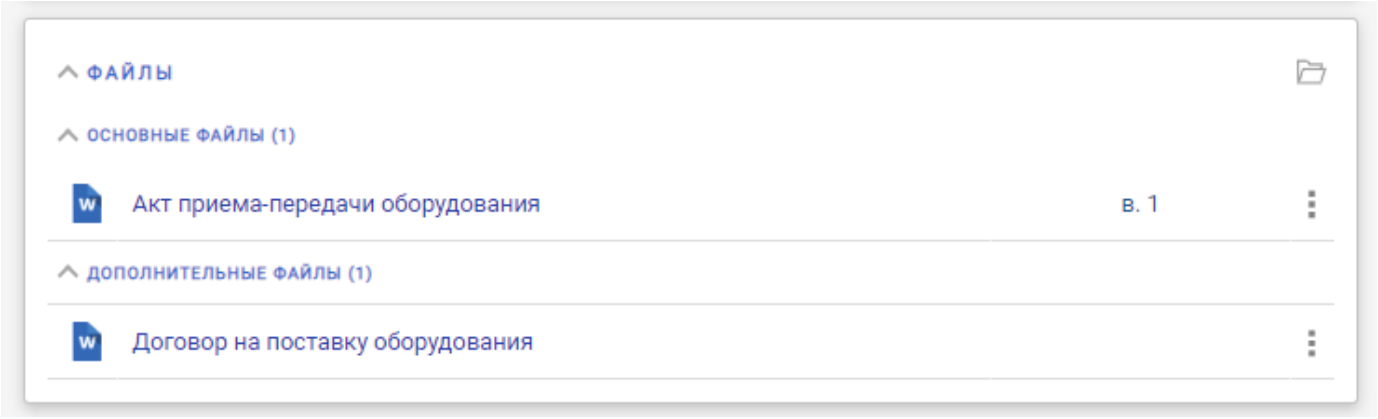


Рисунок 448. Элемент "Список файлов" с выбранными файлами

Если возможность добавления дополнительных файлов отключена, элемент Список файлов не разделяется на области основных и дополнительных файлов; в заголовке элемента отображается количество приложенных основных файлов.

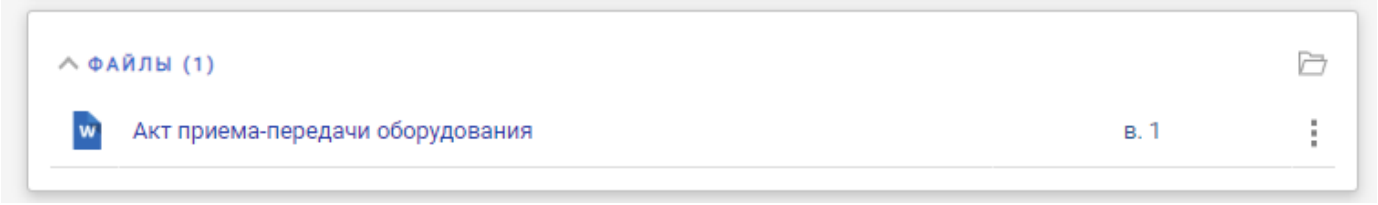


Рисунок 449. Элемент Список файлов с выбранными файлами с запрещённым добавлением дополнительных файлов

Добавление файлов со сканера

Условия для добавления файла со сканера

Элемент управления предоставляет возможность добавлять файлы со сканера, если:

1. Функция активирована при настройке Web-клиента.
2. Пользователь работает на персональном компьютере с операционной системой Windows или Linux.
3. DVWebTool или DVSupService установлены и запущены.
4. На компьютере с ОС Windows зарегистрирован совместимый WIA или TWAIN-сканер.

5. На компьютере с ОС Linux необходим сканер с поддержкой SANE.

Добавить файл со сканера



Добавление файла со сканера в ОС Linux доступно только для сканеров с поддержкой SANE.

Чтобы добавить файл со сканера:

1. Нажмите кнопку  и выберите тип добавляемого файла: *основной* или *дополнительный* (если разрешено).

   Будет открыто окно сканирования.

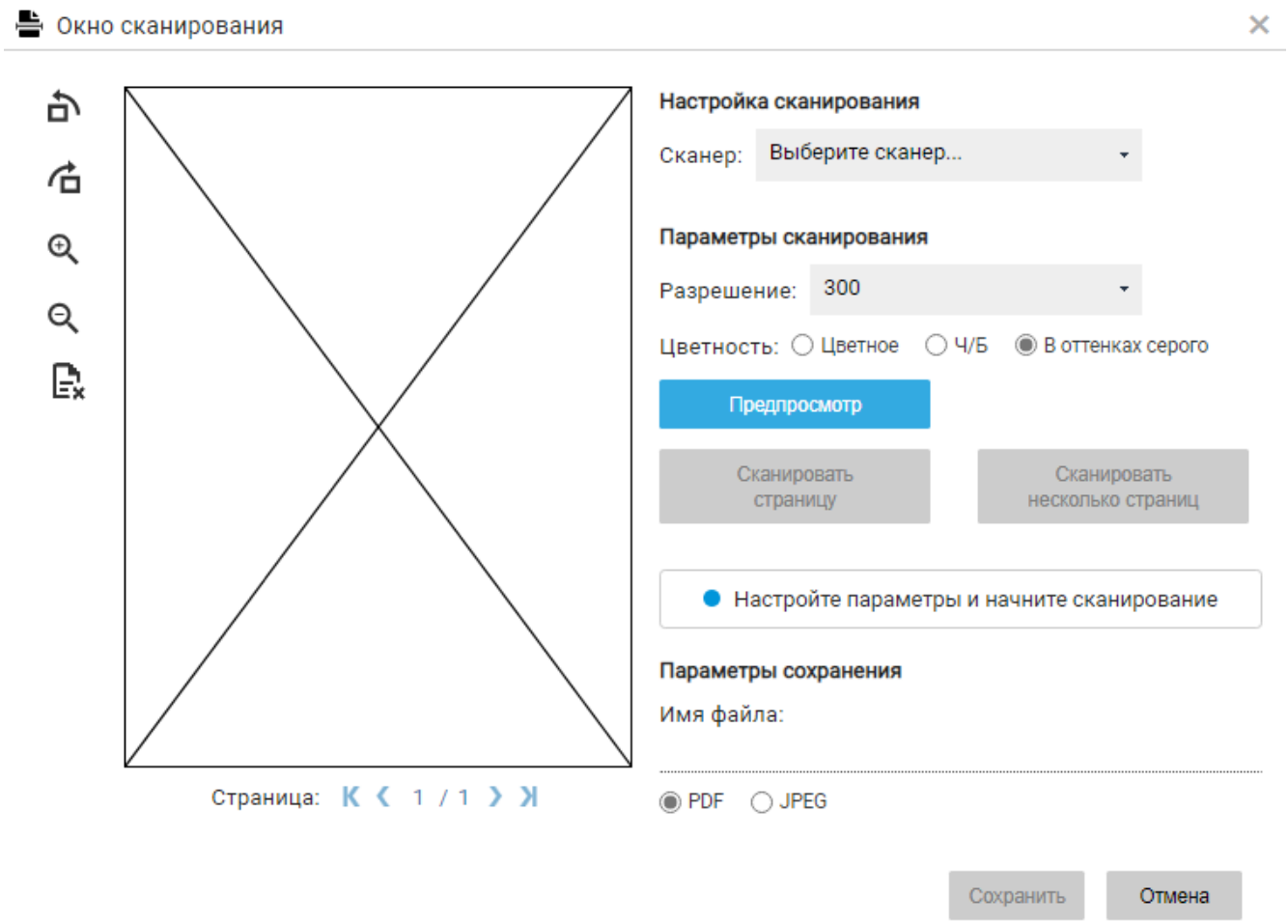


Рисунок 450. Окно сканирования

2. Выберите используемый сканер в списке *Сканер*.
3. Выберите разрешение сканера из списка доступных. Чем выше разрешение, тем лучше качество изображения, но больше размер файла.

4. Выберите цветность получаемого изображения: **цветное**, **черно-белое**, **оттенки серого**.

   Параметры сканирования сохраняются в браузере и устанавливаются автоматически при следующем открытии окна сканирования.

5. Если требуется предварительно оценить сканируемый документ, нажмите кнопку **Предпросмотр**.

   После завершения сканирования в области предварительного просмотра появится изображение.

   Инструкция по работе с компонентом предпросмотра приведена в пункте Предпросмотр файла.

6. В зависимости от количества сканируемых страниц нажмите кнопку:

   - **Сканировать страницу** — будет выполнено сканирование одного листа, каждое новое отсканированное изображение будет добавлено в конец документа.
   - **Сканировать несколько страниц** — будет выполнено сканирование нескольких листов, листы будут добавлены в конец файла в отсканированном порядке (сканер должен поддерживать функцию потокового сканирования).

     Документ будет отсканирован. После этого можно:

     - Повернуть изображение с помощью кнопок  и ,
     - Изменить масштаб и посмотреть другие страницы.
     - Удалить страницы кнопкой . Нажатие на кнопку удаляет выбранную страницу. Если отсканированных страниц не обнаружено, кнопка будет недоступна.

7. Укажите название файла, с которым он будет добавлен в карточку.
8. Выберите тип файла: `.pdf` или `.jpeg`. Если сканируется несколько страниц, будет использован формат `.pdf`.
9. Нажмите кнопку **Сохранить**. Файл будет добавлен в карточку.

Изменить тип файла

Чтобы изменить тип добавленного файла (*основной* или *дополнительный*), перетащите его в область файлов другого типа: из *основных* в *дополнительные*, или наоборот.

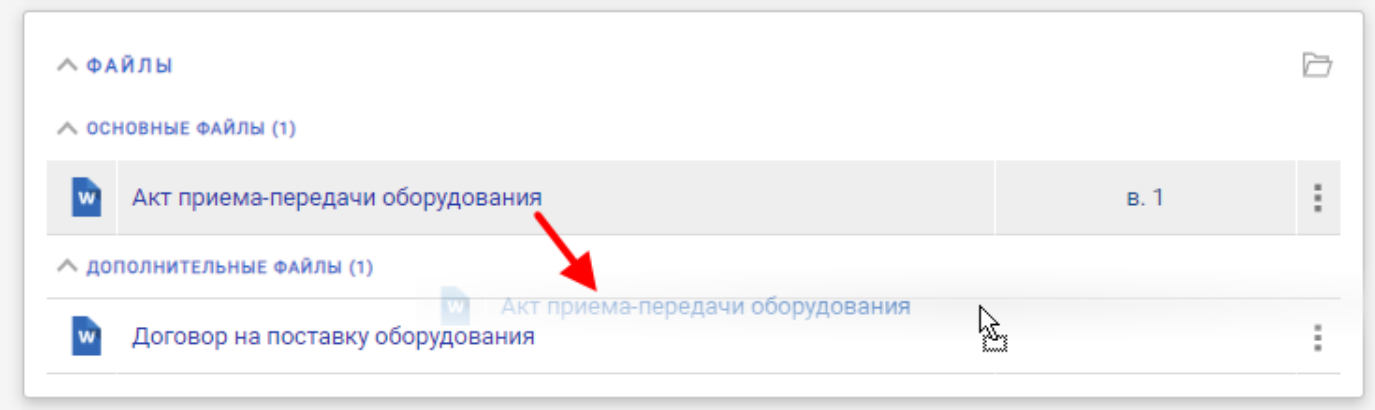


Рисунок 451. Изменить тип файла

Если возможность добавления дополнительных файлов отключена, изменить тип файла описанным образом нельзя.

Переименовать файл

Чтобы переименовать файл:

1. Нажмите ⋮ **> Переименовать в строке файла**.

   Переименование будет недоступно, если файл заблокирован или отсутствуют права на изменение имени файла (операция *Переименование файлов*).

2. Введите новое имя файла в открывшемся окне.

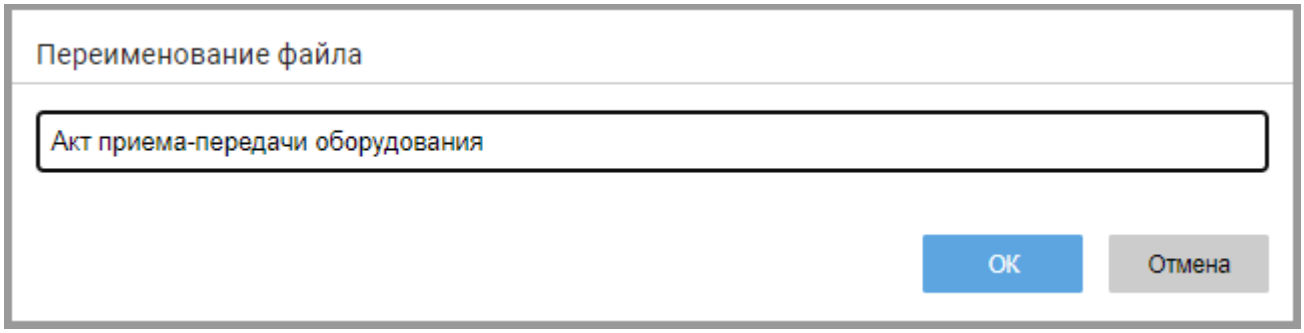


Рисунок 452. Переименование файла

3. Нажмите кнопку **OK**. Будет создана новая версия файла с указанным именем. Файлы в существующих версиях не переименовываются. Если имя файла не изменилось, новая версия не создаётся.

Ограничения для имени файла

- Название файла не может содержать символы: `*`, `/`, `\`, `|`, `<`, `>`, `"`, `?`, `:`, `#`, `%`, `+`, `&`. При попытке сохранить файл с запрещёнными символами будет выдана ошибка `Заданное имя не является допустимым именем файла`.
- В имени файла нельзя указать имя другого существующего файла карточки.



Функция *Переименовать* не может быть использована для изменения расширения файла.

Удалить файл

Чтобы удалить файл из карточки:

- Нажмите кнопку  в строке удаляемого файла при создании или редактировании карточки.

  Удаляемый файл, а также его карточка должны быть разблокированы.

- В строке файла нажмите кнопку ⋮ **> Удалить** при просмотре карточки.

  Удаление нужно подтвердить.



При удалении будут удалены все версии данного файла.

Другие операции с файлами

Следующие операции доступны, только если карточка открыта в режиме чтения.

Открыть файл

Чтобы открыть файл в связанной программе нажмите ⋮ **> Открыть** в строке файла.

Условия открытия файлов

Данная возможность доступна для следующих файлов: `.doc`, `.docx`, `.docm`, `.xls`, `.xlsx`, `.xlsm`, `.ppt`, `.pptx`, `.odt` и `.ods`. Функциональность доступна только

при установленном соответствующем ПО и DVWebTool/DVSupService. С добавленными дополнительными форматами функциональность может работать некорректно.

Дополнительные файлы открываются в режиме чтения. Основные файлы, доступные для редактирования в режиме редактирования. Открытый для редактирования файл будет заблокирован от изменений со стороны других пользователей. При сохранении файла с изменениями будет создана его новая версия.

До завершения редактирования файла не закрывайте карточку, из которой он был открыт.

Предварительный просмотр файла

Нажмите на название файла, чтобы открыть его в режиме предварительного просмотра. Инструкция по работе с компонентом предпросмотра приведена в пункте Предпросмотр файла.

Сортировать файлы

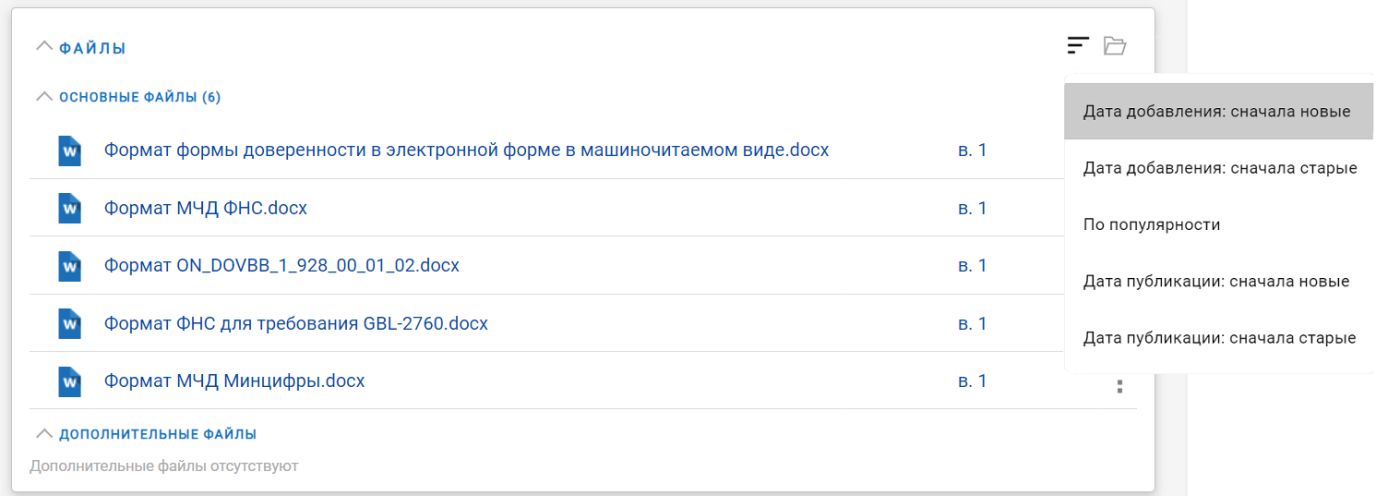


Рисунок 453. ЭУ "Список файлов" с применённой сортировкой

Если сортировка пользователем разрешена в настройках ЭУ, пользователь может выбрать вариант сортировки нажатием на иконку:

- *По имени (возр.)*
- *По имени (убыв.)*
- *По времени добавления (возр.)*

- *По времени добавления (убыв.)*

Сортировка работает одновременно для основных и дополнительных файлов. Сортировка по дате применяется по дате текущей версии. Вариант, выбранный пользователем, сохраняется для него для данного представления карточки и этого экземпляра элемента управления. Выбранный вариант сохраняется при обновлении страницы и при создании новой карточки того же вида.

Скачать файл

Чтобы сохранить файл карточки на компьютер, нажмите ⋮ **> Скачать** в строке файла.

Блокировать/разблокировать основной файл

Основой файл может быть временно заблокирован для защиты от изменений со стороны других пользователей.

Чтобы блокировать или разблокировать файл, нажмите ⋮ **> Заблокировать** или **Разблокировать** в строке файла. Заблокированный файлы выделяются розовой заливкой фона.

Просмотреть версии основного файла

Текущая версия файла отображается в основной области списка файлов. Номер текущей версии указывается справа от названия файла.

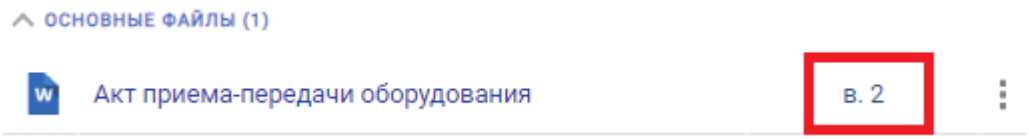


Рисунок 454. Текущая версия файла

Чтобы просмотреть другие версии файла, нажмите на номер текущей версии. Для версии указываются номер, автор и дата создания (добавления файла).

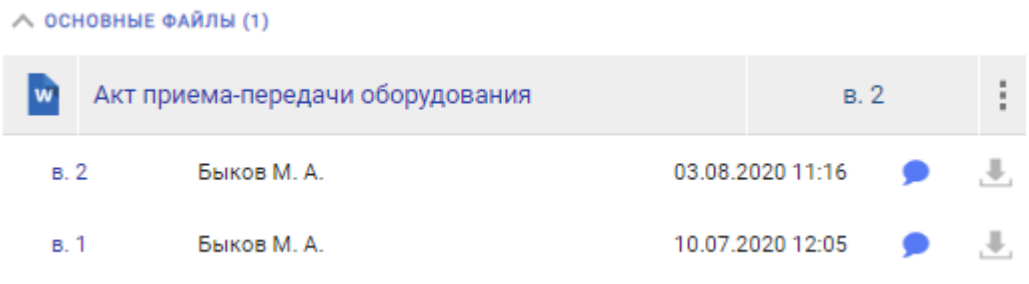


Рисунок 455. Список версий файла

Загрузить новую версию основного файла

Чтобы загрузить новую версию файла, нажмите ⋮ **> Загрузить версию** в строке файла и выберите новую версию файла из файловой системы. Название файла новой версии может отличаться от существующей. Если название файла новой версии не совпадает с названием существующей, версия будет переименована. Для переименования потребуются права на переименование файла. После загрузки версии номер текущей версии увеличится.



Тип файла должен совпадать с типом файла, версия которого добавляется.

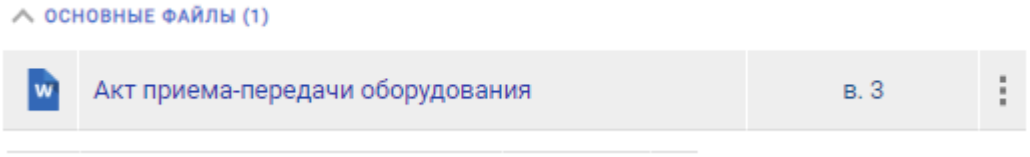


Рисунок 456. Новая версия файла

Скачать версию основного файла

Чтобы сохранить определённую версию файла на компьютер, откройте список версий и нажмите кнопку в строке скачиваемой версии.

Изменить файл

Способ изменения файла зависит от его типа:

- Чтобы изменить основной файл, нужно скачать его на компьютер, внести изменения в скачанный файл и загрузить его новую версию в карточку.
- Чтобы изменить основной файл поддерживаемого формата можно открыть его из меню, внести изменения и закрыть с сохранением.
- Чтобы изменить дополнительный файл, нужно скачать его на компьютер, внести изменения в скачанный файл, удалить данный файл из карточки и загрузить изменённый файл в карточку.

Подписать файл

Чтобы подписать файл, нажмите **> Подписать** и выберите вид подписи и сертификат электронной подписи (для квалифицированной подписи). См. подробности в пункте Произвольное подписание документов.

Выгрузить файлы с электронными подписями

Чтобы экспортировать основные подписанные файлы вместе с электронными подписями, нажмите . См. подробности в пункте Экспортировать файлы с

полписью.

Загрузить отсоединённую электронную подпись

Чтобы загрузить отсоединённую электронную подпись основного файла нажмите ⋮ **> Загрузить ЭП** в строке файла и выберите файл отсоединённой подписи. Поддерживаются форматы `.p7s`, `.sign`, `.sgn` или `.sig`, подробнее в пункте Загрузка отсоединённой электронной подписи файла.

Проверить электронные подписи

В элементе управления отображается список наложенных подписей. При нажатии на 🏅 будет открыто окно со списком подписей и результатами их проверки. Подробнее см. "Результаты проверки подписи".

Дело

> ⚠ Используйте только совместимые версии модулей Управление архивом и Web-клиент. Версия 6.2 и несовместима с версией 5.5.
>
> По вопросам совместимости Управление архивом версий 6.1 и 6.2 с Web-клиентом версии 6.1 и выше проконсультируйтесь со специалистом Docsvision.

См. описание ЭУ в документации модуля Управление архивом: 6.2: "Дело"

Кнопка печати

В карточках может находиться элемент управления Кнопка печати. По умолчанию элемент управления добавлен в разметки УД и Заданий. При желании ЭУ может быть также добавлен в другие разметки. Когда ЭУ находится в разметке, карточку можно отправить на печать с использованием одного из доступных шаблонов.

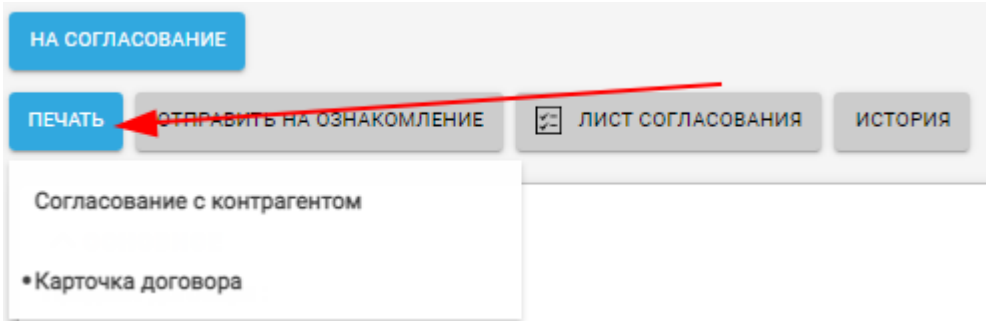


Рисунок 457. Элемент управления "Кнопка печати"

Число шаблонов может варьироваться в зависимости от разметки типа и вида

текущей карточки или настроек элемента управления.

> Доступные шаблоны передаются модулем *Платформа* и суммируются для типа и вида карточки.
>
> Например, если для типа карточки *Документ* в модуле *Платформа* создано 5 шаблонов, а для вида карточки *Исходящий* создано 2 шаблона, для карточки *Исходящего Документа* будет доступно 7 шаблонов = 5 шаблонов от типа + 2 шаблона от вида.

Если число доступных шаблонов больше одного, при нажатии на кнопку печати появится раскрывающийся список. Если доступен только один шаблон, он применяется сразу же после нажатия на кнопку.

Кнопка отправки письмом

Кнопка отправки карточки почтой позволяет отправить карточку по электронной почте. Файлы можно отправлять только из ЭУ Список файлов и Файловая панель.



В мобильной версии кнопка не отображается, функция отправки карточки по email недоступна.

Чтобы отправить карточку по email:

1. Откройте карточку, содержащую кнопку **Отправить письмом**.
2. Нажмите кнопку **Отправить письмом**, чтобы открыть диалог выбора файлов для отправки по почте.
3. Если выбор файлов разрешён в программе Конструктор Web-разметок, выберите основные и дополнительные файлы, которые должны быть приложены к письму. Выставите флаги напротив файлов или групп файлов и нажмите кнопку **ОК**.
4. Дождитесь формирования файла письма. Файл письма будет скачан на компьютер пользователя.
5. Откройте файл письма с карточкой в почтовом приложении.
   - В теме письма указывается дайджест карточки.
   - Тело письма формируется в зависимости от выбранного шаблона. Если выбран вариант ***Без шаблона***, тело письма останется пустым.
   - В списке вложений будут отображены выбранные файлы.

HTML-форматирование:

Если в карточке присутствуют поля с HTML-кодом и для печати используются стандартные шаблоны, текст в таких полях будет выводиться на печать в отформатированном виде.

По умолчанию HTML-форматирование учитывается только в БД с версией Базовые объекты 5.5.5 и выше. Для БД с Базовые объекты более ранних версий необходимо вручную отредактировать шаблоны для карточки *Документ*, привязанные на вкладке *Экспорт* в *Справочнике видов карточек*. Стандартные шаблоны хранятся в папке `C:\Program Files\Docsvision\Platform\Tools\Database\BackOffice\Xslt`.

В справочнике видов можно указать актуальные шаблоны или вручную изменить стандартные. В шаблоне необходимо обернуть HTML-текст в тег `<EscapeHTML>`.

Задания и Группы заданий

Работа в Web-клиенте

Задания и Группы заданий

Дополнительные разделы документации, содержащие информацию, полезную при работе с Web-клиентом Docsvision.

- Файловая панель
- Файловая панель согласования

Файловая панель

Для работы с файлами карточек *Задание* и *Группа заданий* предназначен элемент управления `Файловая панель`.

ЭУ `Файловая панель` *предоставляет следующие функции работы с файлами:*

- Добавлять файлы,
- Откреплять файлы,
- Просматривать файлы (ограничено),

- Редактировать файлы (ограничено),
- Загружать новые версии файлов,
- Просматривать файлы документов "Задания" или "Группы заданий".

Интерфейс элемента зависит от режима работы с карточкой и наличия загруженного файла:

- Элемент управления без добавленного файла при создании новой карточки:



Рисунок 458. Файловая панель при создании карточки Задание

- Элемент управления при просмотре карточки:

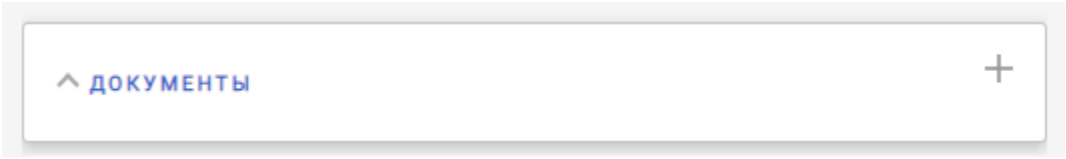


Рисунок 459. "Файловая панель" при просмотре карточки "Задание"

- Элемент управления с добавленными файлами при просмотре карточки:

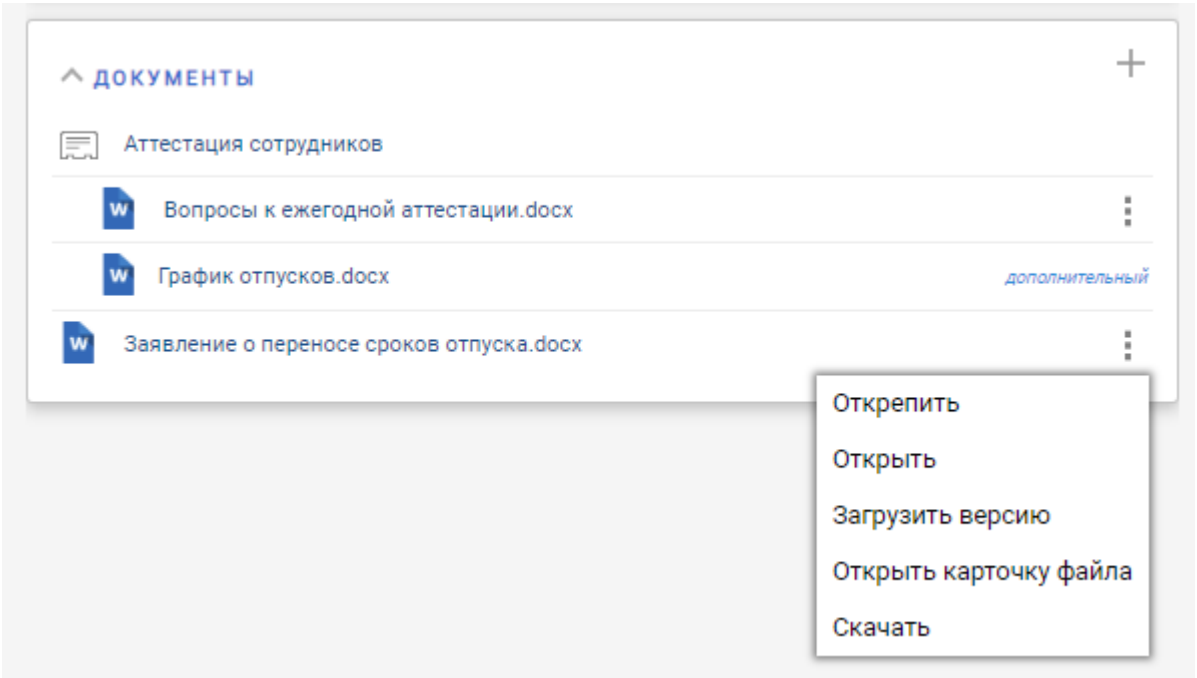


Рисунок 460. "Файловая панель" с добавленными файлами в карточке "Задание"

Добавление файлов

Добавить файл можно двумя способами:

- Перетащите файл из файловой системы в область элемента `Файловая панель`.
- Когда файл не добавлен, нажмите на элемент и выберите файл с помощью Проводника.

  При добавлении файла будет создана новая карточка *Файл* с выбранными файлом.

Условия добавления файлов:

- Типы файлов, которые можно добавлять: `.docx`, `.pdf` (в т.ч `PDF/A-1a`), `.rtf`, `.dot`, `.doc`, `.dotm`, `.docm`, `.dotx`, `.ods`, `.odt`, `.csv`, `.xlt`, `.xls`, `.xltm`, `.xlsm`, `.xltx`, `.xlsx`, `.jpg`, `.jpeg`, `.gif`, `.png`, `.bmp`, `.tiff`, `.tif`, `.emf`, `.wmf`, `.xslt`, `.xsl`, `.xml`, `.htm`, `.html`, `.txt`, `.log`, `.cs`, `.asmx`, `.ascx`, `.aspx`, `.js`, `.cpp`, `.h`, `.css`, `.idl`, `.ini`, `.java`, `.rc`, `.rc2`, `.shtm`, `.shtml`, `.sql`, `.zip`, `.rar`, `.7z`, `.vsd`, `.vsdx`, `.msg`, `.one`, `.pptx`, `.ppt`, `.avi`, `.wav`, `.pfx`.

  Некоторые файлы формата `.xlsm` могут не отображаться в области предварительного просмотра.



Хранение букв `ё` и `й` в операционных системах MacOS и Windows различается. Если вы планируете работать с файлами в обеих системах, рекомендуется не использовать эти буквы в названиях файлов.

Длина имени файла в любой ОС не должна превышать 255 байт.

Открепление файлов

Чтобы открепить файл:

- При создании или редактировании карточки нажмите кнопку  в строке файла.
- При просмотре карточки нажмите ⋮ **> Открепить** в строке файла.

> Карточка *Файл* при этом не удаляется.

Редактирование файла

Чтобы открыть файл в связанной программе в режиме просмотра карточки нажмите ⋮ **> Открыть** в строке файла. Открытый для редактирования файл будет заблокирован от изменений со стороны других пользователей. Элемент

управления поддерживает версионирование, это значит, что при сохранении файла с изменениями будет создана новая версия.

Условия редактирования файлов:

Данная возможность доступна для следующих файлов: `.doc`, `.docx`, `.docm`, `.xls`, `.xlsx`, `.xlsm`, `.ppt`, `.pptx`, `.odt` и `.ods`. Функциональность доступна только при установленном соответствующем ПО и DVWebTool/DVSupService. С добавленными дополнительными форматами функциональность может работать некорректно.

До завершения редактирования файла не закрывайте карточку, из которой он был открыт.

Предварительный просмотр

Щелкните по названию файла, чтобы открыть его в режиме предварительного просмотра. Инструкция по работе с компонентом предпросмотра приведена в пункте Предпросмотр файла.

Скачивание файла

Чтобы скачать файл на устройство, нажмите ⋮ **> Скачать**.

Загрузка новой версии файла

Чтобы добавить новую версию существующего файла:

1. Нажмите ⋮ **> Загрузить версию**.
2. Выберите новую версию файла с устройства. Если название файла новой версии не совпадает с названием существующей, версия будет переименована. Для переименования потребуются права на переименование файла.

Открытие карточки файла

Для открытия карточки *Файл*, содержащей файл *Задания* или *Группы задания*, нажмите ⋮ **> Открыть карточку файла**.

Файловая панель согласования

Для работы с файлами задания согласования предназначен компонент `Файловая панель согласования`. Данный элемент позволяет просматривать и комментировать существующие версии файлов, добавлять новые версии файлов.

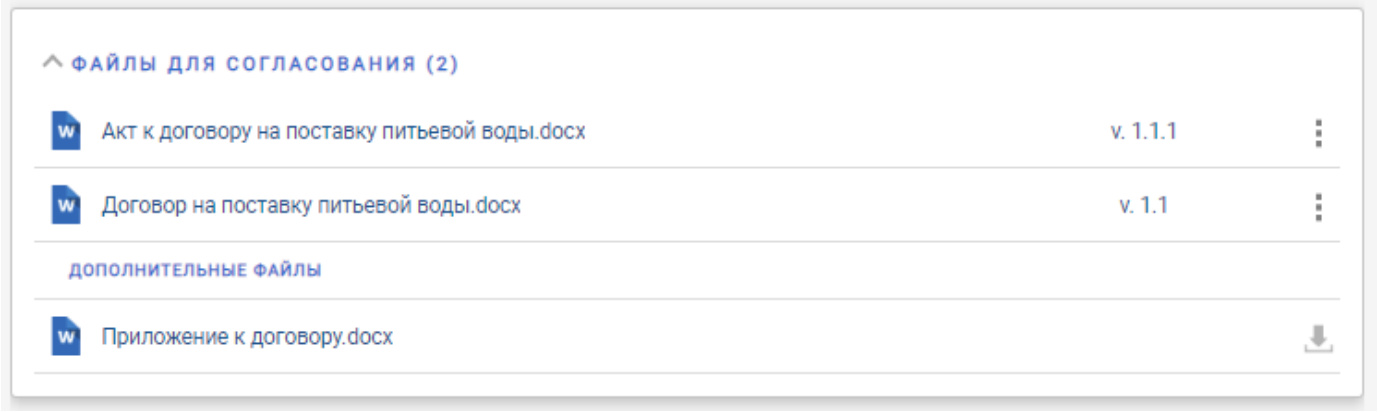


Рисунок 461. Файловая панель согласования

В компоненте отображается Список файлов для согласования. Список файлов для согласования включает основные файлы согласуемого документа, с номерами версий, а также дополнительные файлы согласуемого документа. Дополнительные файлы не участвуют в согласовании.

Просмотр версий файла

По умолчанию в компоненте Файловая панель согласования отображаются только текущие версии согласуемых файлов. Для открытия списка версий файла щелкните по номеру его текущей версии (например, v. 1.1.1).

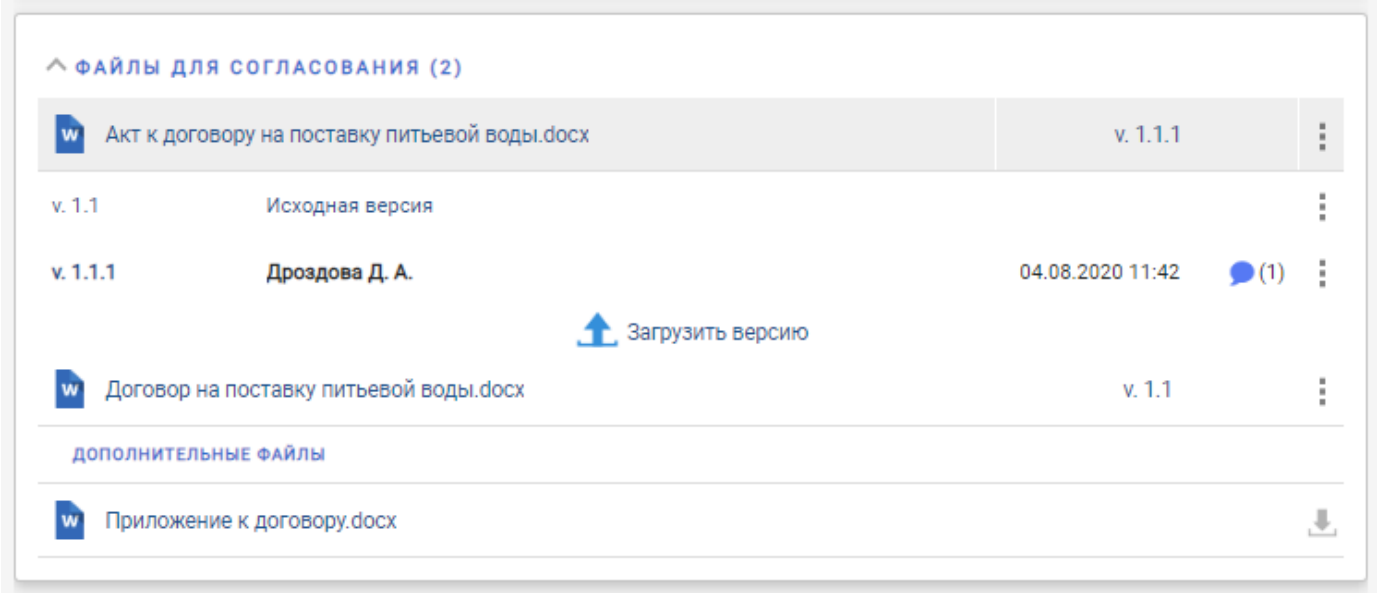


Рисунок 462. Файловая панель согласования

Для версий указываются: номер, автор (за исключением исходной версии) и дата создания.

Если к версии добавлен комментарий, отображается значок с указанием количества комментариев.

Редактирование версии

Чтобы открыть файл в связанной программе, нажмите ⋮ **> Открыть**. При сохранении открытого файла с изменениями будет создана новая версия.

Условия редактирования файла:

Данная возможность доступна для следующих файлов: `.doc`, `.docx`, `.docm`, `.xls`, `.xlsx`, `.xlsm`, `.ppt`, `.pptx`, `.odt` и `.ods`. Функциональность доступна только при установленном соответствующем ПО и DVWebTool/DVSupService. С добавленными дополнительными форматами функциональность может работать некорректно.

До завершения редактирования файла не закрывайте карточку, из которой он был открыт.

Предварительный просмотр



По умолчанию предпросмотр документов в карточках отключен. Чтобы включить предпросмотр, необходимо настроить компоненты Р7-Офис Сервер документов/OnlyOffice.

Щелкните по названию файла, чтобы открыть его в режиме предварительного просмотра. Инструкция по работе с компонентом предпросмотра приведена в пункте Предпросмотр файла.

Скачивание версии

Чтобы скачать версию файла на устройство, нажмите ⋮ **> Скачать**.

Загрузка новой версии

Чтобы добавить новую версию файла:

1. Нажмите ⋮ **> Загрузить версию**.
2. Выберите новую версию файла с устройства. Название файла должно совпадать с названием файла существующей версии.

[0] Например, корневой сертификат КриптоПро

[1] Правила доступа также распространяются на подпапки, если для них не отключено наследование прав доступа.

[2] Определяется настройками вида документа.

[3] Поддерживается только указанный справочник. Несмотря на метку "Данные устарели", информация из справочника актуальная. См. подробнее раздел "Актуализация справочников ЕСНСИ" на сайте Госуслуги

[4] Включает незавершённые и отклонённые задания.

Работа с программой Конструктор Web-разметок

Важная информация при работе с программой Конструктор Web-разметок

Web-клиент Конструктор Web-разметок

Важная информация при работе с программой Конструктор Web-разметок

Информация об установке, запуске, поддержке режима мультитенантности и ведении журнала для программы Конструктор Web-разметок.

- Установка программы Конструктор Web-разметок
- Запуск программы Конструктор Web-разметок
- Особенности работы в многопользовательском режиме
- Больше подробностей в навигационном меню слева…

Установка программы Конструктор Web-разметок

Программа Конструктор Web-разметок предназначена для настройки разметок модуля Web-клиент.

Конструктор Web-разметок может быть установлен отдельно или совместно с сервером Web-клиента.

Запуск программы Конструктор Web-разметок

Запустите Конструктор Web-разметок от имени администратора Docsvision из меню **Пуск** › **Docsvision** › **Конструктор Web-разметок**.

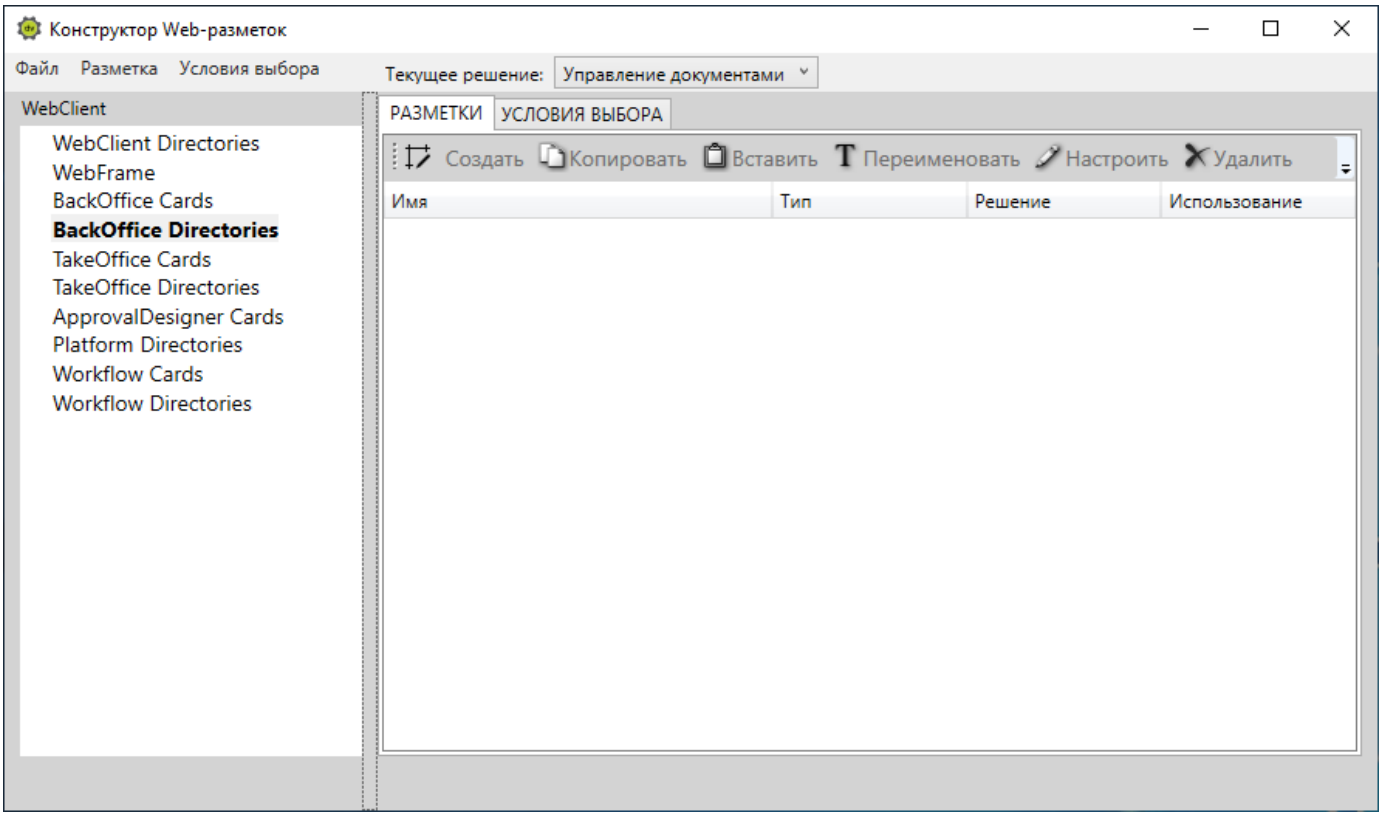


Рисунок 463. Главное окно программы Конструктор Web-разметок

При подключении программы Конструктор Web-разметок к серверу Docsvision используются следующие параметры:

- Подключение осуществляется к серверу Docsvision, который был указан при настройке Web-клиента.
- Подключение осуществляется к базе данных, выбранной по умолчанию. БД по умолчанию задаётся в *Консоль настройки Docsvision*.

Авторизация

При необходимости авторизации в программу Конструктор Web-разметок можно войти под учётной записью из ActiveDirectory или любого LDAP/LDAPS каталога, зарегистрированного на сервере Docsvision.

Чтобы вызвать окно авторизации, запустите конструктор с параметром `/login`, появится окно для ввода логина и пароля и флагом `Войти как текущий пользователь`. По умолчанию флаг снят, после его установки, поля ввода логина и пароля становятся недоступными, а вход будет выполнен от имени текущего пользователя, по аналогии с открытием конструктора без параметра `/login`.

При авторизации в конструкторе проверяется, состоит ли пользователь в группах безопасности **DocsVision Administrators** и **DocsVision Security**

Administrators в конфигурационном файле, справочнике сотрудников или LDAP/LDAPS каталоге, в т.ч. ActiveDirectory.

После успешной авторизации в заголовке окна программы Конструктор Web-разметок отображается логин сотрудника и имя рабочей БД. Если пользователь вводит некорректные или несуществующие данные, будет выведено сообщение `Некорректное имя пользователя или пароль`. Повторите авторизацию с корректными данными.



Для корректной работы авторизации убедитесь, что в конфигурационном файле программы указан корректный адрес сервера Docsvision.

Подключение к другому серверу или БД

Если необходимо подключаться к другому серверу или БД:

1. Откройте конфигурационный файл `C:/Program Files (x86)/Docsvision/WebClientToolkit/WebLayoutsDesigner.appsettings.json`.
2. Измените значение атрибутов параметра **Docsvision › WebClient › Server › SessionManagerConnection**:
   - ***ConnectionAddress*** — укажите строку подключения к требуемому серверу Docsvision;
   - ***BaseName*** — укажите псевдоним требуемой БД.

На сервере Docsvision, который указан в ***ConnectionAddress***, пользователь программы Конструктор Web-разметок должен являться *Администратором Docsvision*.

Особенности работы в многопользовательском режиме

Конструктор Web-разметок является многопользовательской программой: настраивать разметки могут одновременно несколько пользователей.

Параллельная настройка одной разметки несколькими пользователями **не поддерживается**. Разметка блокируется когда она открыта в редакторе разметок, блокировка будет снята после закрытия разметки.

При попытке открыть разметку, которую редактирует другой пользователь будет выдано предупреждение "Разметка редактируется другим пользователем". В данном случае пользователь может:

- открыть разметку для чтения
- снять блокировку нажатием кнопки **Редактировать**.



Сохранить разметку сможет только последний открывший её для редактирования пользователь.

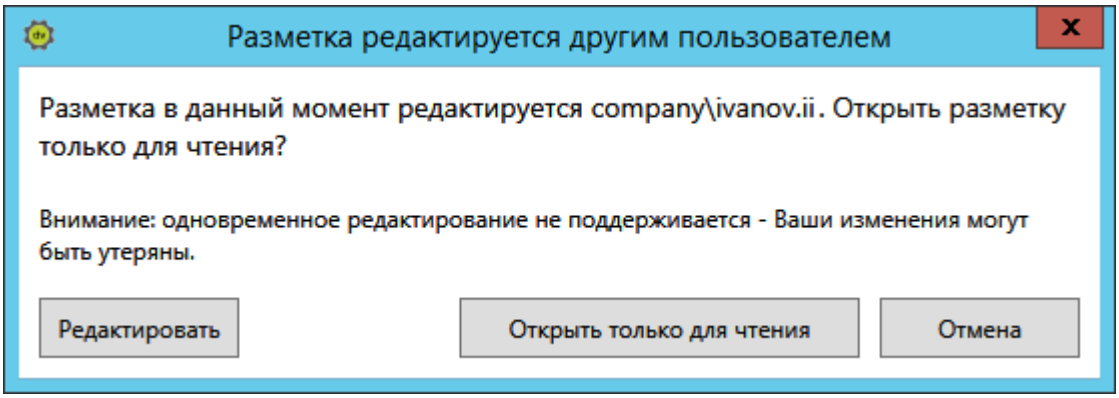


Рисунок 464. Предупреждение при открытии разметки, заблокированной другим пользователем

Конструктор Web-разметок периодически загружает актуальные настройки разметок, условий выбора и пр. из базы данных. Таким образом пользователь без перезапуска программы может увидеть изменения, сделанные другими пользователями. При получении новых данных будет показано всплывающее сообщение "Синхронизированы правки другого пользователя". Период обновления настроек составляет по умолчанию 30 секунд.

Чтобы изменить период обновления:

1. Откройте конфигурационный файл `C:/Program Files (x86)/Docsvision/WebClientToolkit/WebLayoutsDesigner.appsettings.json`.
2. Укажите таймаут обновления в секундах в настройке: **Docsvision › WebClient › SettingGroups › WebLayoutsDesigner › RefreshPeriod`**.

   Параметр ***RefreshPeriod*** нужно добавить самостоятельно.
3. Запустите Конструктор Web-разметок.

Особенности работы в мультитенантном режиме

Конструктор Web-разметок не поддерживает работу в мультитенантном режиме. В программе всегда открываются разметки БД, заданной по умолчанию через *Консоль настройки Docsvision*.

Чтобы открыть определённую базу данных в программе Конструктор Web-разметок:

1. Закройте программу, если она была открыта.
2. Найдите конфигурационный файл, расположенный в папке по адресу `C:/Program Files (x86)/Docsvision/WebClientToolkit/WebLayoutsDesigner.appsettings.json`.
3. Откройте файл для изменения и найдите настройку ***BaseName=""***.
4. Измените настройку, указав псевдоним БД между символов кавычек ("").
5. Сохраните файл и повторно откройте Конструктор Web-разметок.

   Программа откроется для указанной БД. Текущая БД будет указана в заголовке окна.

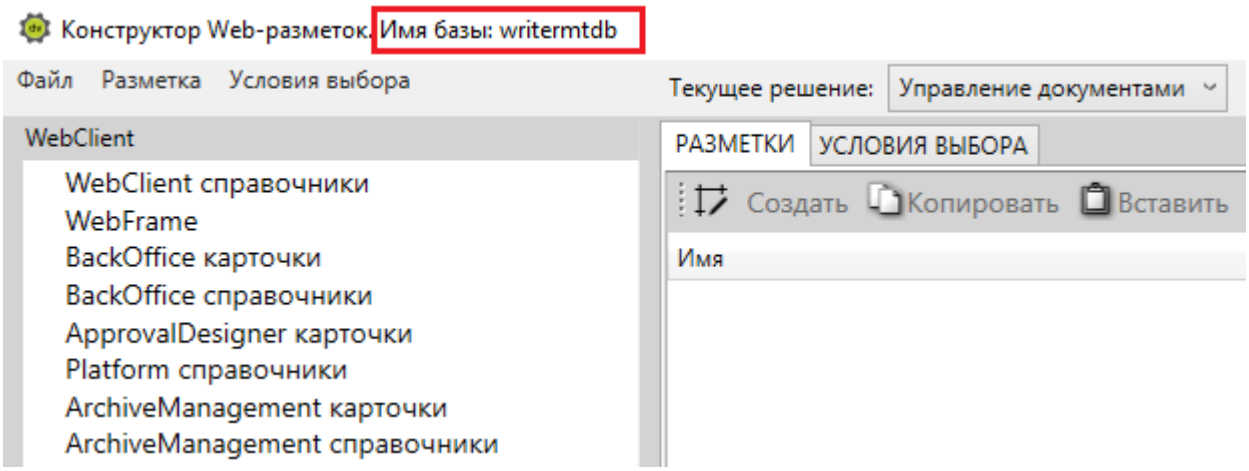


Рисунок 465. Псевдоним текущей рабочей БД в конструкторе Web-разметок

Журнал программы Конструктор Web-разметок

Для программы Конструктор Web-разметок предусмотрен журнал работы. Журнал хранится по пути `C:/Program Files (x86)/Docsvision/WebClientToolkit/Logs/`. Имя файла журнала имеет следующий формат: `WebLayoutsDesigner PID #### ГГГГ-ММ-ДД`, где `####` — номер процесса.

Журнал содержит следующую информацию:

1. Вход/выход пользователя.
2. Системная информация о работе программы.
3. Изменение условий выбора разметки.

Включить или отключить ведение журнала можно в конфигурационном файле программы.

Чтобы включить или отключить журнал программы Конструктор Web-разметок:

1. Найдите конфигурационный файл, расположенный в папке по адресу `C:/Program Files (x86)/Docsvision/WebClientToolkit/WebLayoutsDesigner.appsettings.json`.
2. Выполните настройку журналирования:

```
{
  "NLog": {
    "targets": {
        "async": true,
        "logFile": {
            "type": "File",
            "fileName": "${basedir}\\Logs\\WebLayoutsDesigner${event-
properties:item=Tenant} PID ${processId} ${shortdate}.log", ①
            "layout": "[${longdate}][${level}][${callsite}] ${message}
${onexception:${newline}${exception:maxInnerExceptionLevel=10:format=shortType,message
,stacktrace:separator=*:innerExceptionSeparator=&#xD;&#xA;&#x9;}}"
        }
    },
    "rules": [ ②
        {
            "logger": "Microsoft.*",
            "minLevel": "Trace",
            "maxLevel": "Info",
            "final": true,
            "writeTo": "logconsole" ③
        },
        {
            "logger": "*",
            "minLevel": "Error",
            "writeTo": "logFile" ④
        }
    ]
  }
}
}
```

① Измените значение параметра `fileName`, чтобы настроить путь для сохранения журнала работы:
В `fileName` можно использовать допустимые для NLog переменные, например: `\Logs\${level}\WebLayoutsDesigner_${shortdate}.log`.

② Измените значение параметров `minLevel` и `maxLevel`, например, на `trace`, чтобы включить протоколирование всех типов событий.
Допустимые уровни протоколирования приведены на странице NLog в

GitHub.

Для получения дополнительной информации о других настройках NLog обратитесь к документации по данной платформе.

③ Сообщения уровней от `Trace` до `Info` из библиотек, название которых начинается с `Microsoft.`, будут записываться в консоль.

④ Все сообщения уровней от `Error` и выше сообщения будут записываться в файл.

Пользовательский интерфейс программы Конструктор Web-разметок

На рисунке представлен вид главного окна программы после выбора библиотеки карточек и вида карточки.

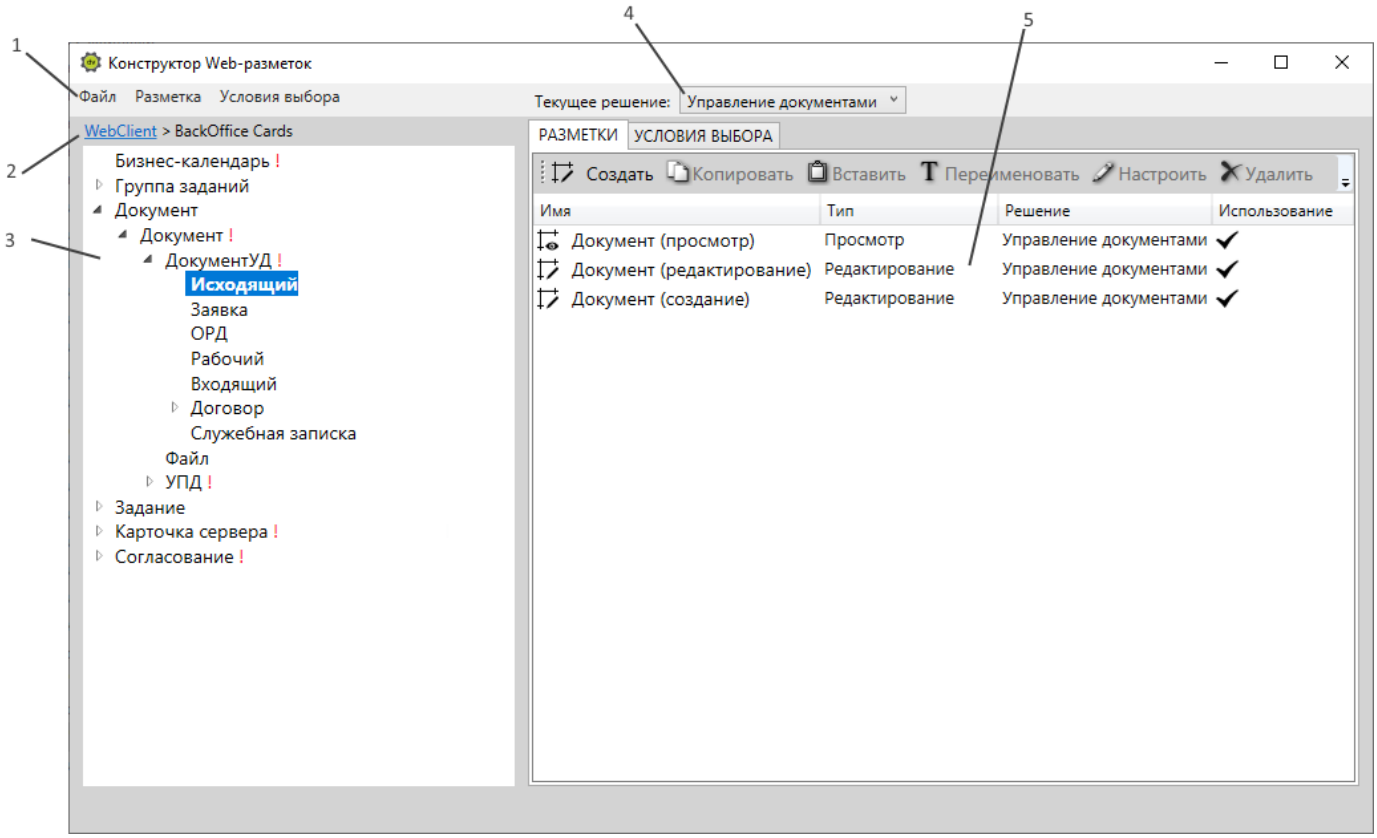


Рисунок 466. Главное окно программы Конструктор Web-разметок со списком разметок исходящего документа

Главное окно содержит:

1. Панель команд.
2. Текущая локация со ссылкой для перехода в корень *Дерева локаций*.
3. Дерево локаций — локации, для которой настраиваются разметки.



Локация — уникальный идентификатор комплекта разметок и условий их выбора, которые предназначены для формирования одной определённой части графического интерфейса Web-клиента: карточка, основное меню, дашборд и т.д.

4. Текущее настраиваемое Решение — список Решений содержит Решение "Мои настройки", в которое сохраняются настройки, не связанные с другими Решениям.
5. Рабочая область со списком разметок и условий выбора разметок локации, выбранной в дереве локаций.

Панель команд

Панель команд программы включает следующие команды:

Файл

Управление Решениями

Открывает Панель управления Решениями.

Локализация

Открывает Панель управления ресурсами локализации.

Локации

Открывает Панель управления локациями.

Выход

Закрывает Конструктор Web-разметок

Разметка

Создать

Открывает окно создания новой разметки.

Копировать из родительского вида

Копирует в текущий вид карточек все разметки из родительского вида. Функция недоступна для корневого вида, являющегося родительским для всех других видов.

Копировать, Вставить

Копирует выбранную разметку для добавления в другой или текущий вид карточек.

Переименовать

Открывает окно изменения отображаемого названия разметки.

Настроить

Открывает окно Редактора разметок для изменения выбранной разметки.

Удалить

Удаляет выбранную разметку.

Условия выбора

Добавить условие

Открывает Редактор условий для создания нового условия выбора разметки.

Дублировать

Создаёт копию выделенного условия выбора.

Редактировать

Открывает Редактор условий для изменения выделенного условия выбора.

Вверх, Вниз

Повышает или понижает приоритет выделенного условия выбора.

Удалить

Удаляет выделенное условие выбора.

Дерево локаций

Область **Дерево локаций** расположена в левой части главного окна программы Конструктор Web-разметок.

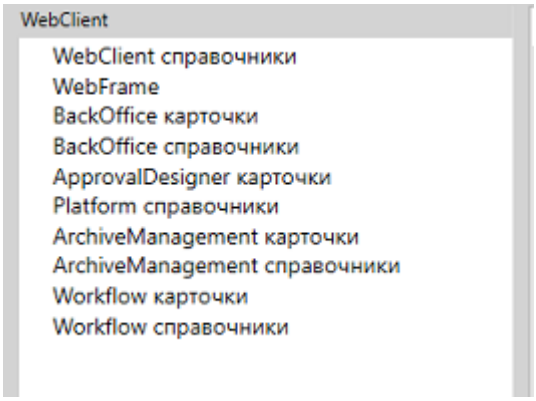


Рисунок 467. Дерево локаций с группами локаций

Дерево локаций содержит группы локаций:

WebClient справочники	Содержит разметки справочников для Web-клиента.
WebFrame	Содержит разметки общих элементов пользовательского интерфейса Web-клиента.
BackOffice карточки, TakeOffice карточки, ApprovalDesigner карточки и др.	Содержат разметки карточек указанных библиотек и решения, основанные на этих библиотеках.  В Web-клиенте отсутствуют элементы управления для работы с карточками Делопроизводство 4.5. Такие ЭУ могут быть реализованы самостоятельно.
BackOffice справочники, TakeOffice справочники, ApprovalDesigner справочники и др.	Содержат разметки справочников.  В Web-клиенте отсутствуют элементы управления для работы с карточками Делопроизводство 4.5. Такие ЭУ могут быть реализованы самостоятельно.

При выборе группы, будет открыт список входящих в неё локаций, а над списком появится ссылка для выхода из группы.

В группах локаций карточек используются специальные обозначения:

!	Означает, что разметка вида карточек не настроена и не наследуется от родительского вида. + Для открытия карточек вида, отмеченного флагом !, будут использованы разметки родительского типа карточек. Если для родительского типа разметки не настроены, открыть карточку невозможно. Дополнительная информация приведена в пункте Настройка разметки карточек.
Серый цвет шрифта	Означает, что разметки вида карточек наследуются от родительского вида. Возможность создания разметок для данной локации заблокирована.

?	Означает, что разметки вида карточек наследуются от родительского вида, но в программе Конструктор Web-разметок для вида настроены разметки. + Флаг ? свидетельствует о неоднозначности настроек разметок вида карточек. При открытии карточек данного вида будут использованы разметки родительского вида, а после отключения наследования — собственные.

Вкладка "Разметки"

Данная вкладка содержит список разметок локации, выбранной в Дереве локаций.

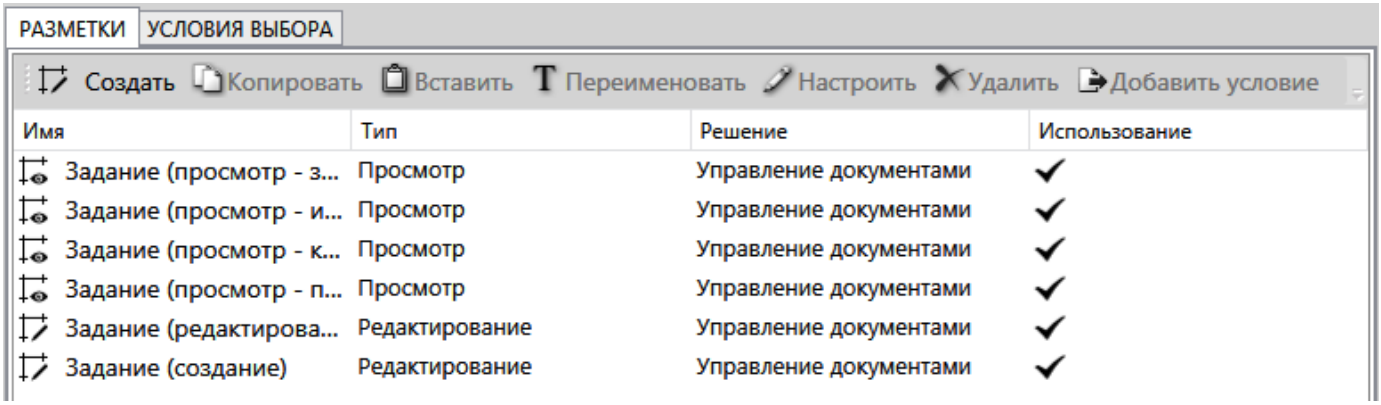


Рисунок 468. Конструктор Web-разметок, вкладка "Разметки"

В списке разметок в порядке следования столбцов отображаются:

- Имя разметки со значком назначения разметки.
- Назначение (тип) разметки:
  - "Просмотр" — разметка, используемая для просмотра данных;
  - "Редактирование" — разметка, используемая для изменения/создания данных.
- Название Решения, в которое включена разметка.
- Статус использования разметки:
  - Если для разметки определены условия использования, флаг активен.
  - Если разметка не используется, флаг неактивен.

Над списком разметок отображается лента команд:

Создать

Открывает окно создания новой разметки.

Копировать, Вставить

Копирует выбранную разметку для добавления в другой или текущий вид карточек.

Переименовать

Открывает окно изменения отображаемого названия разметки.

Настроить

Открывает окно Редактора разметок для изменения выбранной разметки.

Удалить

Удаляет выбранную разметку.

Добавить условие

Добавляет новое условие, при котором выбранная разметка будет использована для просмотра или редактирования карточки.

Вкладка "Условия выбора"

Данная вкладка содержит список условий использования разметок локации, выбранной в Дереве локаций.

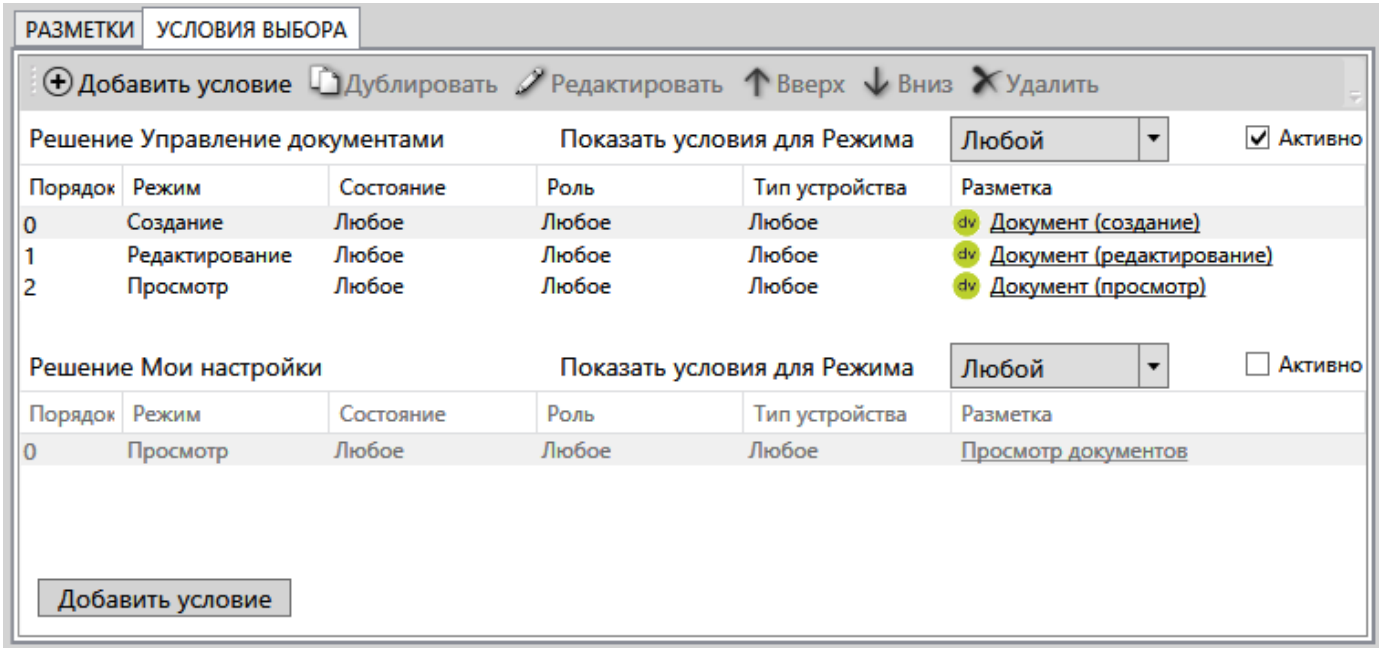


Рисунок 469. Условия выбора разметки

Список условий разделен на группы по Решениям, в которые входят условия.

Флагом **Активно** отмечается группа действующих условий — может быть установлен одновременно только у одной группы условий.

Фильтр *Показать условия для Режима* применяется к параметру **Режим** условия выбора. Для каждой группы условий используется собственный фильтр.

В списке условий выбора отображаются столбцы, соответствующие критериям, при которых действует данное условие. Набор критериев зависит от локации. Например, для локаций карточек библиотеки *Базовые объекты* в списке условий отображаются следующие столбцы:

- Порядковый номер условия в списке, определяющий приоритет данного условия над другими аналогичными условиями.
- Режим открытия карточки, при котором будет использоваться разметка.
- Состояние карточки.
- Роль пользователя.
- Тип устройства, на котором будет использоваться разметка.
- Используемая разметка.



Значение ***Любое*** означает, что критерий выполняется при любых значениях параметра.

Красным цветом выделяется условие, которое не может быть выполнено ни в каких случаях, из-за наличия более общего условия с более высоким приоритетом.

Таблица 23. Описание команд в ленте

Добавить условие	Открывает Редактор условий для создания нового условия выбора разметки.
Дублировать	Создаёт копию выделенного условия выбора.
Редактировать	Открывает Редактор условий для изменения выделенного условия выбора.
Вверх, Вниз	Повышает или понижает приоритет выделенного условия выбора.
Удалить	Удаляет выделенное условие выбора.

Кнопка **Добавить условие** дублирует функциональность одноименной команды

ленты.

Список "Текущее решение"

Раскрывающийся список Текущее решение предназначен для выбора Решения, в котором настраиваются разметки.

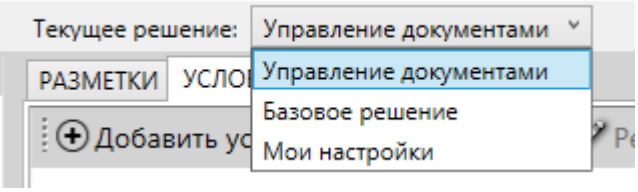


Рисунок 470. Раскрывающийся список "Текущее решение"

В списке представлены Решения, загруженные с помощью Панели управления Web-клиента, а также встроенное Решение *Мои настройки*.

Если в Конструкторе разметок отсутствуют Решения, доступные для изменения, новые разметки и условия выбора разметок можно будет добавлять только в Решение *Мои настройки*.

Работа с программой Конструктор Web-разметок

Перед началом работы с программой Конструктор Web-разметок обратите внимание на ограничения ниже.

- Запрещено изменять настройки стандартных Решений: *Базовые решения* и *Управление документами*.
- Не рекомендуется создавать/удалять/изменять разметки в группе локаций *WebFrame* за исключением следующих разрешённых изменений:
  - Создание/изменение/удаление разметок в локациях *Главная страница*, *Основное меню* и *Диалог профиля пользователя*.
  - Изменение настроек разрешённых элементов управления в разметках в локации *Корень WebFrame*.

Обратите внимание, на обновление справочников и разметок влияет заданный режим и период обновления справочников. См. подробнее "Функциональность "Общие справочники"".

Операции с локациями

Настройка локаций

Пользовательский интерфейс содержит несколько стандартных локаций, определяющих определённые части пользовательского интерфейса Web-клиента. К таким стандартным локациям относятся: *Главное меню*, *Стартовая страница*, *Панель фильтра папки*, *Окна карточек* и др.

Пользователь программы Конструктор Web-разметок может добавить в Web-клиент новую локацию с собственным набором разметок. Данная локация может быть встроена в разметку другой локации или быть доступна как отдельная страница.

Управление локациями осуществляется с помощью *Панели управления локациями*.

Свойства элемента локации "root"

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Данные

Расширенные источники данных

Выбор типа источников данных локации: выбор типов, доступных для элементов управления разметки.

Поведение

Стандартные css классы локации

Название CSS классов, определяющих стандартный стиль локации управления.

Дополнительные css классы локации

Названия дополнительных классов CSS для изменения стиля локации. Перечисляются через пробел.

Фон

Свойство позволяет задать фон и доступно только в следующих локациях: *Папка*, *Последние карточки*, *Результаты поиска*, *Главная страница*.

Доступны следующие варианты назначения фона:

- Без фона — используется стандартный серый фон (значение по умолчанию).
- Стандартный фон — используется фон, назначенный в ЭУ Содержимое.
- Специальный фон — появляется дополнительное свойство, в котором можно указать путь к файлу по аналогии с фоном дя ЭУ "Содержимое".
  - Если выбран специальный фон, обязательно требуется указать путь до файла с картинкой фона в свойстве `Картинка фона`. В противном случае будет невозможно сохранить изменения разметки.
  - При выборе специального фона и картинки фона также можно изменить свойство `Отображение фона` — выпадающий список с вариантами ***Растянуть*** или ***Замостить***.

События

Перед изменением разметки

Вызывается перед изменением разметки.

Перед изменением состояния

Вызывается перед изменением состояния.

Перед открытием карточки

Вызывается перед открытием карточки.

Перед сохранением карточки

Вызывается перед сохранением карточки.

Перед удалением карточки

Вызывается перед удалением карточки.

Подготовка к сохранению карточки

Вызывается подготовке к сохранению карточки.

После загрузки всех ЭУ

Вызывается после загрузки всех ЭУ.

После открытия карточки

Вызывается после открытия карточки.

После сохранения карточки

Вызывается после сохранения карточки.

После удаления карточки

Вызывается после удаления карточки.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Выбрать локацию

Выберите локацию в дереве локаций, чтобы изменить разметки.

Чтобы настроить разметки карточек:

1. Выберите группу локаций, соответствующей библиотеки карточек. Каждая локация в списке соответствует определённому виду карточек.
2. Выберите требуемую локацию.



Локация обычно содержит разметки для полной и мобильной версии Web-клиента. В локации также могут быть расположены разметки других решений или пользовательские разметки.

После выбора локации, в рабочей области программы Конструктор Web-разметок будут открыты разметки и условия выбора разметок выбранной локации.

Пример 1. Локации группы "WebFrame":

Группа локаций WebFrame содержит следующие локации:

Главная страница

Разметки дашборда.

Диалог профиля пользователя

Разметки диалогового окно *Профиль*, открываемый из меню пользователя.

Компоненты папки

Грид

Разметки области представления. Используются в Новом гриде.

Групповые операции с карточками

Разметки панели групповых операций.

Диалог поисковых параметров

Разметки диалогового окна атрибутивного поиска.

Избранное

Разметки папки *Избранное*.

Панель поиска по столбцам представления

Разметки панели настройки фильтра папки.

Папка

Разметки области представления.

Последние карточки

Разметки страницы *Последние карточки*. Используются при включении экспериментальной функции *Новый грид*.

Результаты поиска

Разметки страницы *Результаты поиска*. Используются при включении экспериментальной функции *Новый грид*.

Корень WebFrame

Содержит разметки основного окна Web-клиента (заголовок, в котором отображается текущая страница. Основное меню с деревом папок. и область, в которой отображается содержимое выбранной папки или карточки).

Локация *Корень WebFrame* содержит две разметки: для полной и мобильной версии Web-клиента. Создание и удаление разметок в данной локации запрещено. В разметках допускается только изменение элемента управления `barcodeSearchButton`

Основное меню

Разметки дерева папок.

Разметки для элементов управления

Диалоги операций карточек

Разметки боковой панели, отображаемой при выполнении операции с карточкой:

Делегировать

При делегировании задания.

Завершить

При завершении задания.

На доработку

При возврате задания на доработку.

Отклонить

При отклонении задания.

Перенос сроков исполнения

При переносе сроков исполнения задания.

Поиск карточек для ссылок

Разметки окна добавления ссылок карточки.

Создание дочернего задания согласования

Разметки панели создания дочернего согласования.

Панель управления локациями

Панель управления локациями предназначена для управления локациями: добавления, удаления, изменения.

Чтобы открыть *Панель управления локациями*, нажмите **Файл** › **Локации**. Будет открыто окно *Редактирование локаций*.

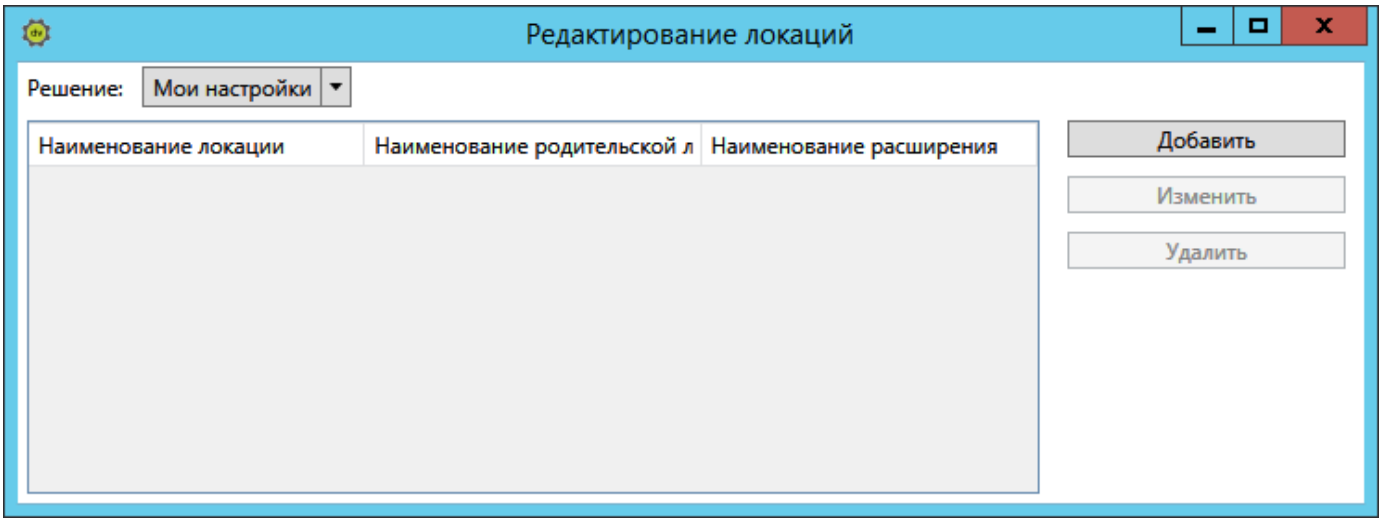


Рисунок 471. Панель управления локациями

Компоненты диалогового окна Редактирование локаций:

Раскрывающийся список Решение

Решение, в котором настраиваются локации.

Список добавленных локаций

Все доступные локации, кроме стандартных.

Кнопки управления

Добавить, **Изменить**, **Удалить**.

Создать новую локацию

1. Откройте Панель управления локациями.
2. Выберите Решение, в которое будет включена локация.
3. Нажмите кнопку **Добавить**. Будет открыто диалоговое окно настройки локации.

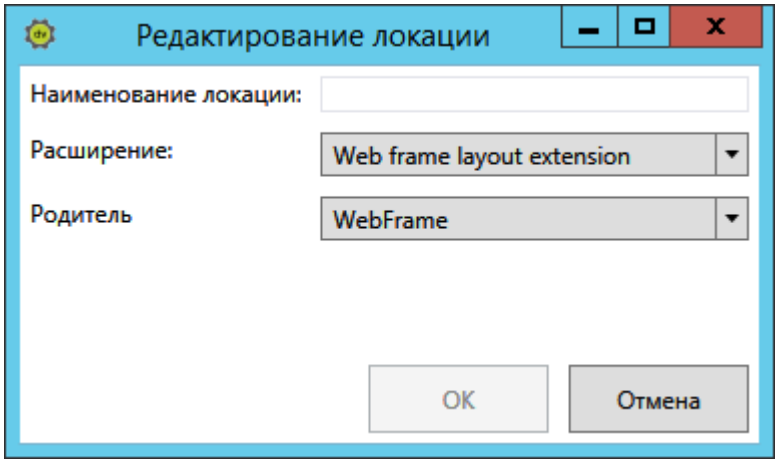


Рисунок 472. Диалоговое окно настройки локации

4. Введите название локации.
5. В списке Расширение оставьте значение по умолчанию `Web frame layout extension`.
6. В списке Родитель выберите локацию, для которой новая локация будет дочерней.

   Размещение локации в структуре дерева локаций не влияет на её возможности.

7. Нажмите **ОК**. Новая локация будет добавлена в список локаций выбранного Решения.



Для новой локации необходимо настроить разметки и условия выбора.

Встроить локацию в разметку

Локация может быть добавлена в разметку другой локации или предоставляться по ссылке.

Встроить локацию в существующую разметку можно при помощи ЭУ Контейнер локации.

1. Добавьте в разметку элемент управления `Контейнер локации`.
2. В свойстве `Локация` выберите добавляемую локацию.
3. Установите флаг `Часть родительской разметки`, чтобы элементы управления разметки локации были доступны, как часть родительской локации.

Чтобы сделать локацию доступной по ссылке:

1. Добавьте в разметку элемент управления Ссылка / Локация.
2. В свойстве `Локация` добавленного ЭУ выберите локацию, на которую добавляется ссылка.

Вы также можете настроить отображение связанной карточки в текущей карточке, см. Встраивание одной карточки в другую.

Операции с решениями

Настройка Решений

Конструктор Web-разметок позволяет объединять настройки разметок, условия выбора и данные локализации в *Решение*, которое может быть выгружено для последующего распространения. Поддержку распространяемого решения выполняет инженер Docsvision без привлечения разработчиков.

Полноценное *Решение* также может включать собственные элементы управления, скрипты и файлы стилей. Данные компоненты Конструктор Web-разметок не выгружает. Скрипты, файлы стилей и компоненты собственных элементов управления должны быть загружены на сервер Web-клиента самостоятельно.

Режим работы с Решением

Решение, реализованное с помощью *программы Конструктор Web-разметок*, может работать в одном из двух режимов: *Поставщик* или *Заказчик*. В зависимости от режима работы, вводятся определённые ограничения на изменение данного *Решения*:

Поставщик

Разрешено создавать и изменять любые объекты данного *Решения*: разметки, локализованные ресурсы и так далее.

Заказчик

Запрещено изменять любые объекты данного *Решения* кроме условий выбора разметок. Разрешено создавать новые разметки, запрещено включать их в *Решение*.

См. также пункт Изменение режима работы с Решением.

Панель управления Решениями

Панель управления Решениями предназначена для управления списком Решения, изменения режима Решения, выгрузки и загрузки Решений.

Чтобы открыть "Панель управления Решениями", нажмите **Файл › Управление Решениями**.

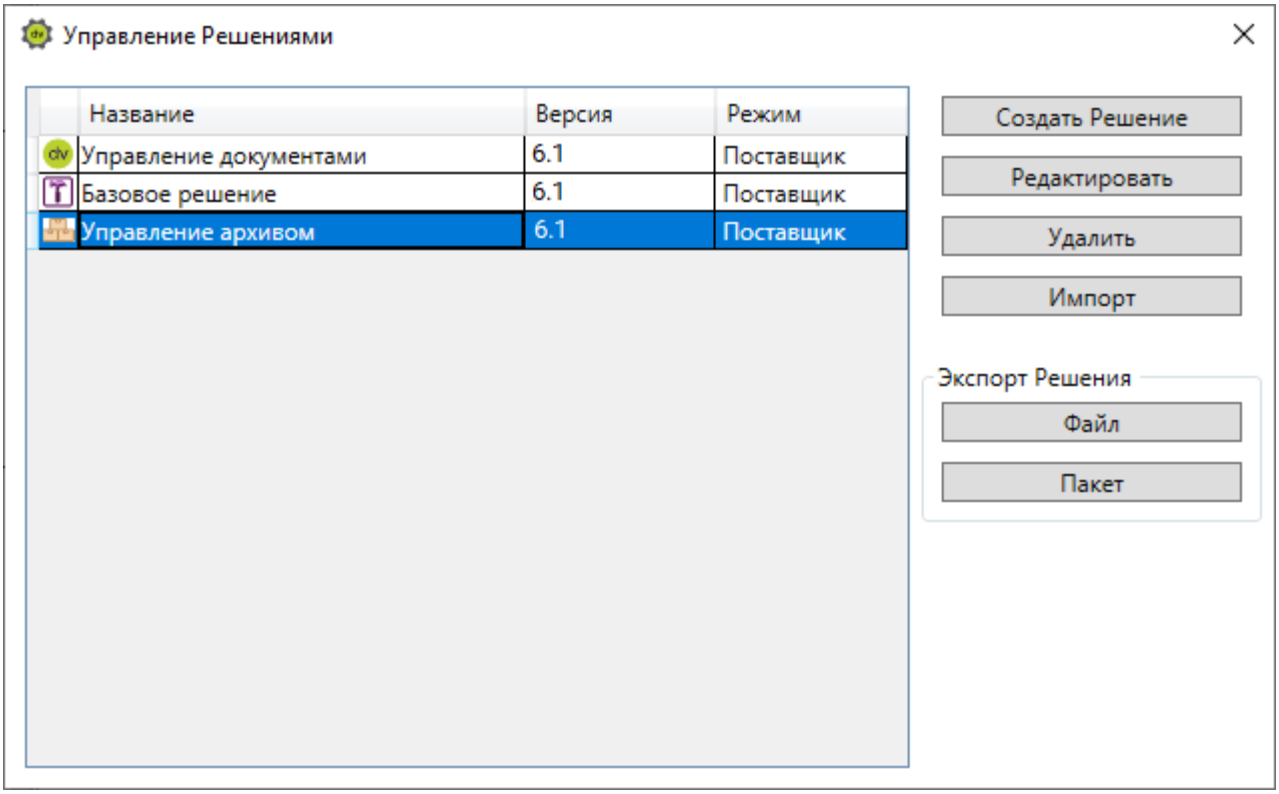


Рисунок 473. Панель управления Решениями

Компоненты диалогового окна "Управление Решениями":

Список решений

Для каждого решения указывается название, версия, режим работы.

Кнопки управления

Создать Решение

Открывает диалоговое окно создания нового Решения.

Переключить режим

Изменяет режим работы выбранного Решения с *Поставщик* на *Заказчик* и обратно. См. пункт Изменение режима работы с Решением.

Редактировать

Открывает диалоговое окно изменения Решения.

Удалить

Удаляет Решение вместе с разметками видов карточек, которые в него включены.

Экспорт Решения

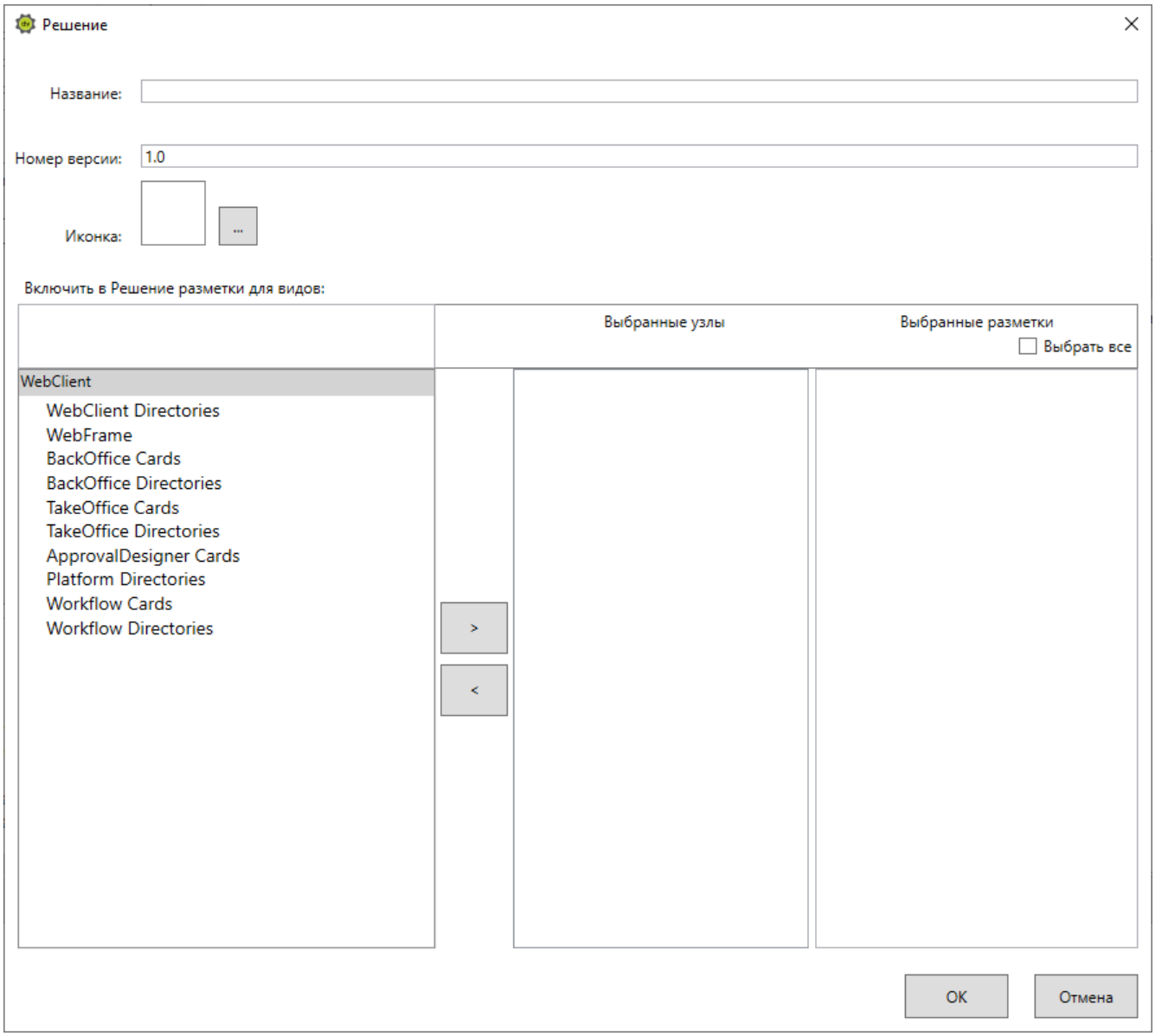
Выгружает выбранное Решение в файл или пакет.

Импорт

Загружает Решение из файла.

Создать новое Решение

1. Вызовите команду **Файл** › **Управление Решениями**.
2. Заполните поля в открывшемся окне Панели управления Решениями.



3. Нажмите **ОК** для сохранения. Решение будет добавлено в список Решений. Режим нового Решения — *Поставщик*.

Описание элементов окна см. в пункте Список компонентов диалогового окна "Решения".

Добавить разметки в Решение

При создании Разметка автоматически включается в Решение, которое выбрано в списке Текущее решение. После создания Разметку можно включить в Решение, только если она включена в Решение *Мои настройки*.

Чтобы включить разметку в Решение:

1. Вызовите команду **Файл › Управление Решениями**.

   Будет открыта Панель управления Решениями.

2. Выберите Решение, в которое следует включить вид карточек, и нажмите **Редактировать**.

   Будет открыто окно *Решение*.

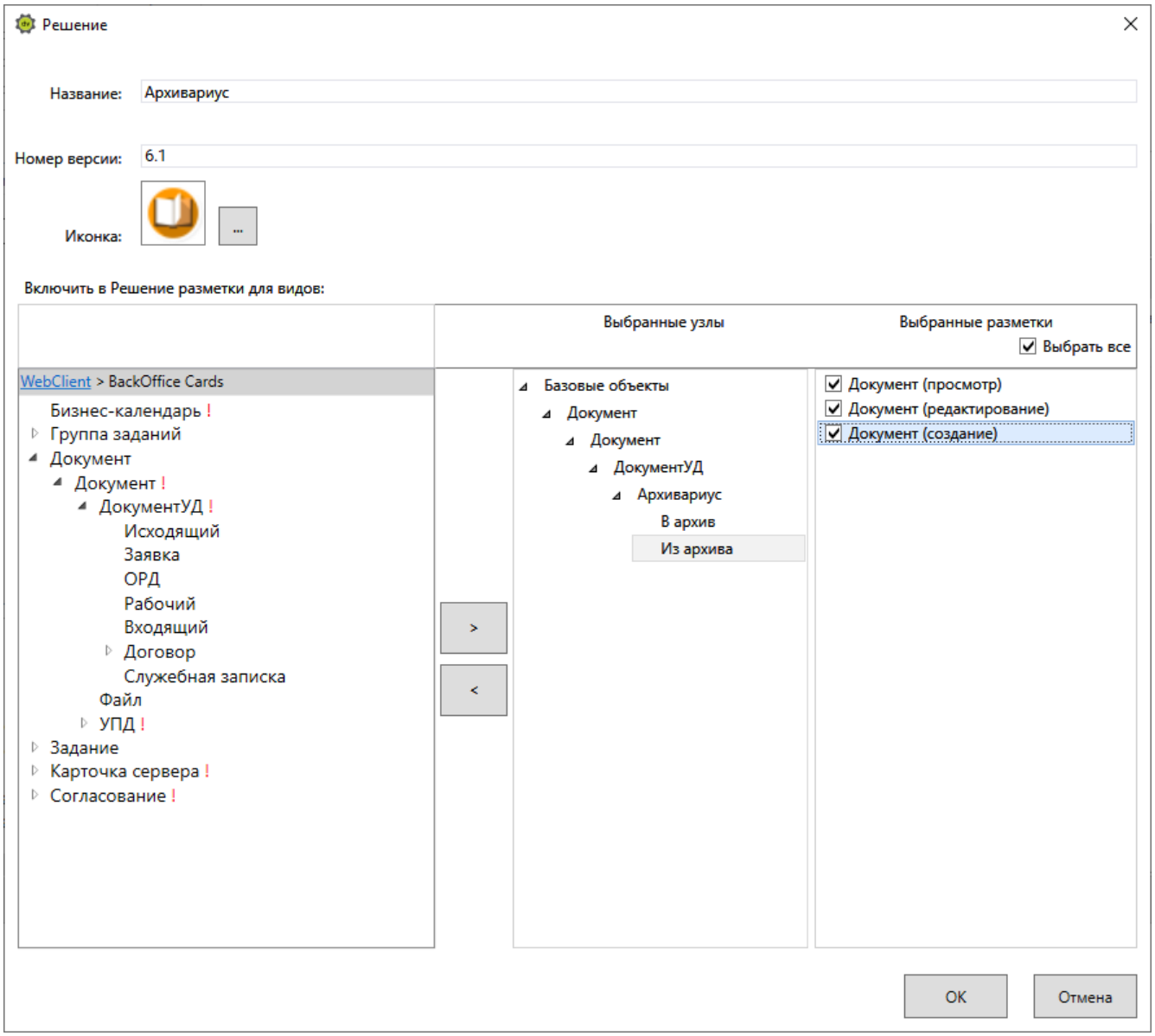


Рисунок 474. Окно управления Решением

Описание элементов окна см. в пункте Список компонентов диалогового окна "Решения".

3. В дереве локаций выберите группу локаций, а затем локацию, разметки которой выгружаются.
4. Нажмите кнопку [>]. Локация будет добавлена в список выбранных локаций.

   Чтобы убрать локацию из выбранных, выберите её в списке, и нажмите кнопку [<].
5. В списке выбранных локаций выберите локацию, разметки которой включаются в Решение. В списке разметок будут показаны все разметки, которые могут быть включены в Решение.

6. Установите флаги у разметок, которые должны быть включены в Решение.
7. Если требуется, аналогичным образом включите в Решение разметки других локаций.
8. Нажмите кнопку **ОК**. Выбранные разметки будут включены в Решение.

Список компонентов диалогового окна "Решения"

Компоненты диалогового окна "Решение":

Название

Название решения, которое будет отображаться в списке доступных Решений. Рекомендуется указывать уникальное название Решения.

Номер версии

Номер версии должен быть записан в формате: X.X, X.X.X или X.X.X.X.

Иконка Решения

Иконка отображается в столбце *Разметка* списка *Условия выбора*.

Дерево локаций

Список локаций, разметки которых могут быть выгружены.

Выбранные узлы

Список выбранных локаций.

Выбранные разметки

Список выгружаемых разметок.

Экспортировать Решение

В экспорт решения включаются настройки разметок, используемые в них локализованные ресурсы, условия выбора разметок, локации. Скрипты и стили, используемые в разметках, в выгрузку не включаются.

Чтобы экспортировать решение:

1. Нажмите **Файл** › **Управление Решениями**, чтобы открыть Панель управления Решениями.
2. Выберите выгружаемое Решение. Решение должно находиться в режиме Поставщик.
3. Выберите формат экспорта: **Файл** или **Пакет**.
4. Укажите каталог для экспорта и нажмите **Сохранить** (экспорт файлом) или

OK (экспорт пакетом).

Экспорт пакетом

Решение может быть экспортировано пакетом. Пакет представляет собой папку с набором файлов и имеет следующий вид:

Структура папки решения:

```
|-Управление документами
|--Layouts ①
|---Входящий УД (локация)_79EE95AB-80C3-4413-8CBD-61683CA0D0E9.xml ②
|---Исходящий УД (локация)_9D0C243B-B019-4A95-B4EB-64AE39A0352A.xml
|---...
|-- ConditionGroups.xml ③
|-- Locales.xml ④
|-- Solution.xml ⑤
```

① Папка с файлами разметок.

② Файл разметки.

③ Условия выбора разметок.

④ Локализации.

⑤ Главный файл решения.



Не переименовывайте файлы в пакете!

Экспорт файлом

Решение может быть экспортировано единым XML-файлом с данными.

Как и пакет, файл включает локализации, условия выбора разметок и разметки.

Импортировать Решение

Если после экспорта в Решение были внесены правки, импортированы будут только те разметки, в которые были внесены изменения.



Чтобы избежать нарушения условий выбора разметок, перед импортом решений убедитесь, что служба сервера Web-клиента запущена.

Чтобы импортировать решение:

1. Нажмите **Файл** › **Управление Решениями**. Будет открыта Панель

управления Решениями.

2. Нажмите кнопку **Импорт**. Будет открыто окно выбора импортируемого файла.

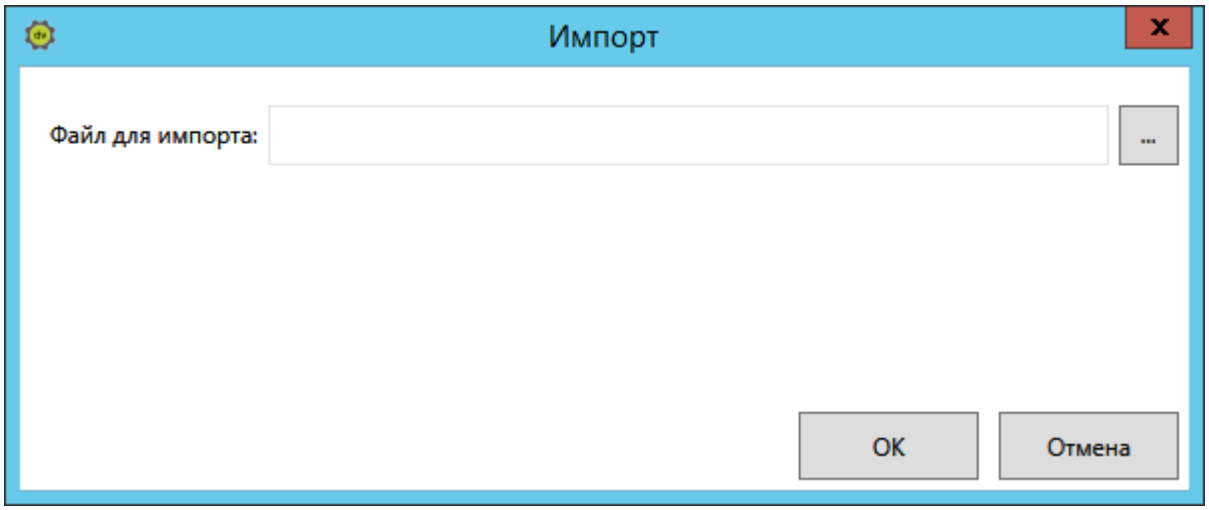


Рисунок 475. Окно "Импорт"

3. Выберите файл и нажмите кнопку **OK**.

   a. Если экспортированное решение в формате файла, выберите файл с нужным названием.

   b. Если экспортированное решение в формате пакета, выберите файл `Solution.xml` в папке решения.

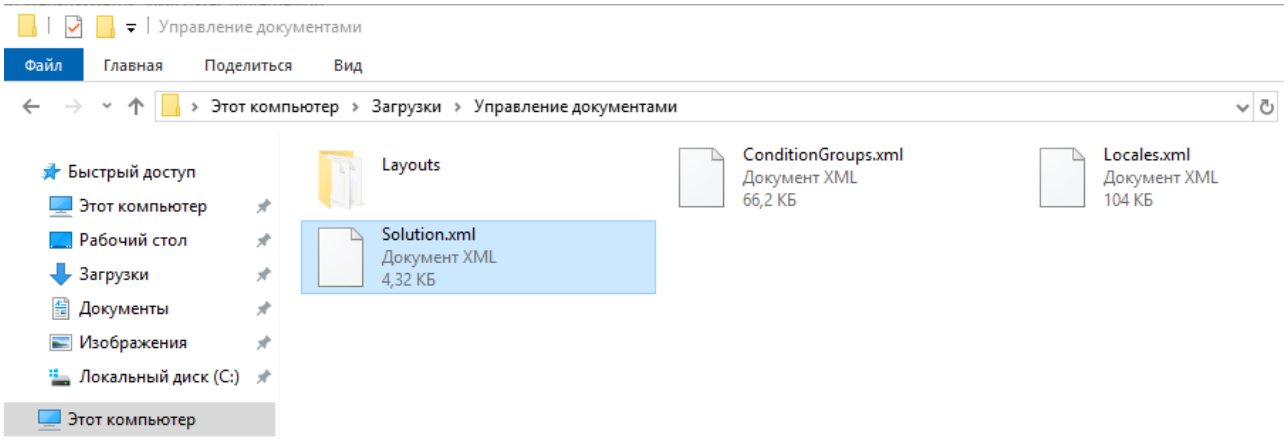


Рисунок 476. Файл "Solution.xml" в папке решения



Если из папки с разметками был удалён файл определённой разметки, при импорте в условиях выбора может отображаться условие с неназначенной разметкой.

4. В базу данных будут загружены разметки, условия выбора разметок и ресурсы локализации из XML файла.

В результате импорта будут загружены разметки для тех видов карточек, которые есть в базе-приёмнике.

Если в импортируемых данных содержатся локации с именами существующих в БД локаций, но другими идентификаторами, будет выдано предупреждение о дублировании локаций. Данные всё равно будут импортированы.



Скрипты и стили, которые используются в разметках Решения, нужно скопировать на сервер Web-клиента самостоятельно.

Изменить режим работы с Решением

Режим работы с Решением определяет возможность изменения разметок, условий выбора и других настроек Решения. Конструктор Web-разметок поддерживает два режима:

- *Заказчик*. Изменение Решения заблокировано.
- *Поставщик*. Изменение Решения разрешено.

Чтобы изменить режим работы с Решением:

1. Откройте Панель управления Решениями.
2. Выберите Решение, режим которого нужно изменить.
3. Нажмите кнопку **Переключить режим**. Режим изменится на противоположный.



- Запрещено изменять настройки стандартных Решений Web-клиента: *Базовое решение* и *Управления документами*. При необходимости выполните следующие шаги.
- После установки Решения в рабочую базу данных не рекомендуется изменять его режим работы на *Поставщик*, т. к. это может привести к ошибкам при обновлении Решения.

Отобразить кнопку "Переключить режим"

Кнопка **Переключить режим** по умолчанию недоступна. Чтобы её отобразить выполните следующие действия:

1. Откройте конфигурационный файл `C:/Program Files (x86)/Docsvision/WebClientToolkit/WebLayoutsDesigner.appsettings.json`.
2. Измените значение настройки **Docsvision** › **WebClient** › **SettingGroups** ›

WebLayoutsDesigner › **DesignerMode** на значение:

- `Vendor`, чтобы разрешить изменять режим Решения.
- `Customer`, чтобы запретить изменять режим Решения.

3. Запустите Конструктор Web-разметок и откройте Панель управления Решениями.
4. Выберите Решение в списке и нажмите кнопку **Переключить режим**. Значение режима будет изменено.

Изменить стандартные Решения

Изменять настройки стандартных Решений Web-клиента: *Базовое решение* и *Управления документами* запрещается. При необходимости модифицировать стандартные Решения, выполните следующие действия:

1. Создайте собственное Решение.
2. Скопируйте в него стандартные разметки.
3. Модифицируйте копии разметок требуемым образом.
4. Создайте правила для выбора данных разметок.

Изменить "Текущее Решение"

В списке "Текущее решение" выберите Решение, которое хотите изменить. Выберите вариант *Мои настройки*, если нет собственного разработанного Решения.

Если вы планируете распространять настройки разметок, рекомендуется создать Решение перед настройкой разметок и условий выбора. После создания вы сможете выбрать собственное решение из списка.

Возможность изменения настроек Решений *Базовые объекты*, *Управление документами*, *Управление архивом* и Решений сторонних производителей по умолчанию заблокирована. Заблокированные Решения не отображаются в списке. Как разблокировать изменение Решений, см. в пункте Изменить режим работы с Решением.



Вместо изменения стандартных Разметок можно скопировать данные разметки в собственное Решение или в *Мои настройки*. Копии Разметок могут быть изменены без ограничений.

Когда вы выбрали Решение:

- Новые разметки будут создаваться в выбранном Решении.
- Новые условия выбора будут создаваться в выбранном Решении.
- Возможность изменения условий выбора из других Решений будет заблокирована.
- Панель управления ресурсами локализации будет предоставлять доступ к ресурсам выбранного Решения.

Операции с разметками

Общие операции с разметками

Ограничения при работе с разметками

При работе со списком разметок обратите внимание на следующие особенности:

- Разметка может быть двух типов: *для редактирования* или *создания* и для *просмотра*.

  Для большинства видов карточек необходимо разрабатывать разметки обоих типов. Для общих локаций достаточно разработать только разметки просмотра.

- Разметка может входить только в одно Решение или в *Мои настройки*.
- К вида карточек могут быть собственные разметки или унаследованные от родителя в зависимости от настроек в *Справочнике видов карточек*.

  Если разметки наследуются, но у родительского вида не настроены собственные разметки, для открытия карточек будут использованы разметки типа карточек.

- Карточки, у которых нет собственных или унаследованных разметок, не могут быть открыты.

Элементы редактора разметок

Конструктор Web-разметок предоставляется с набором стандартных элементов управления, которые могут быть использованы для разработки разметок карточек и общих разметок. Данные стандартные элементы управления составляют *Библиотеку* стандартных элементов управления.

Все элементы управления Библиотеки стандартных элементов управления делятся на несколько групп, в зависимости от функционального назначения:

- Главная страница.
- Главное меню.
- Групповые операции.
- Компоненты папки.
- Параметры поискового запроса.
- Профиль пользователя.
- Системные.
- Согласование.
- Специальные.
- Справочники.
- Стандартные.
- Таблицы.
- Фильтр папки.
- Элементы разметки.

Данные группы можно, условно разделить, на:

1. Элементы разметки (далее — **ЭР**) — элементы управления для формирования блочной структуры разметки. Данные элементы управления следует рассматривать как контейнеры, в которые могут встраиваться другие элементы управления.
2. Элементы управления (далее — **ЭУ**) — основные элементы управления, которые, как правило, можно связать с данными или использовать для вызова определённой функции (например, изменить состояние карточки).

Элементы управления обладают набором определённых настраиваемых свойств (базовых и собственных) и событий, на которые можно подписаться для их обработки в скрипте карточки.

В зависимости от локации и типа разметки, пользователю может быть доступен ограниченный набор элементов управления, а также их настроек.



Полное описание стандартных элементов управления приведено в разделе Описание стандартных элементов управления.

Корневой блок разметки

Корневой блок разметки — частный случай элемента управления Блок, в который встраиваются все остальные элементы разметки. Корневой блок может называться root, documentRoot и т.п.

☑ В разметке может быть только один корневой блок.

☑ Корневой блок является предком других элементов разметки.

☑ Корневой блок не отображается в области *Компоновка*.

☑ Корневой блок не содержит настроек ЭУ Блок, связанных с оформлением.

☑ Корневой блок предоставляет возможность подписаться на события карточки (утверждение справедливо для разметок карточек).

☑ Корневой блок не может быть удалён или перемещен.

Создать новую разметку

1. Выберите локацию, для которой создаётся разметка, и Решение, в которое она будет включена.
2. В рабочей области откройте вкладку *Разметки*.
3. Нажмите кнопку **Создать** на ленте команд.

   Будет открыто диалоговое окно создания разметки.

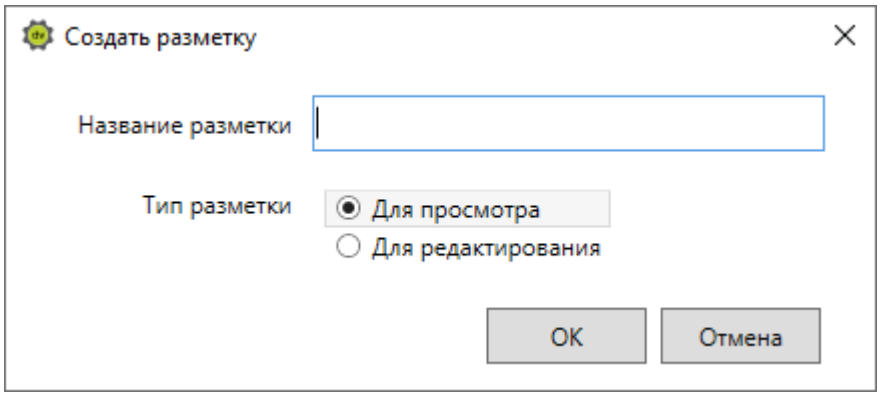


Рисунок 477. Диалоговое окно создания разметки



Возможность создания разметки карточки будет заблокирована, если выбрана локация вида, разметки которого наследуются (название локации серого цвета).

4. Введите название создаваемой разметки, выберите её тип и нажмите кнопку **ОК**.



Название разметки должно быть уникальным для данного вида. Если разметка с таким названием существует, будет выдано предупреждение: *Разметка с таким именем уже существует*; кнопка **ОК** при этом блокируется.

Тип разметки определяет режим работы с данными, в котором данная разметка будет использоваться. После создания разметки её тип изменить нельзя.

Для создания и изменения карточек используются разметки с типом *Для редактирования*.

Новая разметка будет добавлена в список разметок на странице *Разметки*. В столбце *Решение* будет указано Решение, выбранное в списке **Текущее решение**, флаг **Использование** будет снят.

После создания разметку необходимо настроить. Также необходимо настроить условия для выбора разметки.

Копировать разметку

1. Выберите копируемую разметку на вкладке Разметки.
2. Нажмите кнопку **Копировать** на ленте команд.
3. В дереве локаций выберите локацию, для которой копируется разметка.
4. На вкладке *Разметки* нажмите кнопку **Вставить** на ленте команд.

   Разметка будет добавлена в выбранную локацию. Если есть разметка с аналогичным названием, к названию копии будет добавлен постфикс *_Копия*.

При копировании может быть потеряна часть настроек элементов управления разметки, если они не поддерживаются локацией. Например, при копировании разметки одного типа карточек в другой, могут быть потеряны настройки связи с данными карточки, которых нет в целевом типе.

Переименовать разметку

1. Выберите переименовываемую разметку на вкладке Разметки.

2. Нажмите на кнопку **Переименовать** на ленте команд.
3. Введите новое название и нажмите кнопку **ОК**.



Название разметки должно быть уникальным для данного вида. Если разметка с таким названием существует, будет выдано предупреждение: `Разметка с таким именем уже существует`. Кнопка **ОК** при этом блокируется.

Настройка разметок

Настройка разметки — формирование набора настроенных элементов управления, необходимых для работы пользователя с данными. Разметки настраиваются с помощью Редактора разметок.

Стандартные этапы настройки разметки:

1. Сформируйте общую структуру разметки с помощью элементов разметки.
2. Сформируйте отдельные блоки для работы с данными при помощи элементов разметки `Блок`, `Страница вкладок`, `Таблица`.
3. Добавьте в разметку элементы управления для отображения или ввода данных.
4. Настройте элементы управления. Свяжите ЭУ с данными, настройте общие свойства, обработчики событий, локализации и т.д.

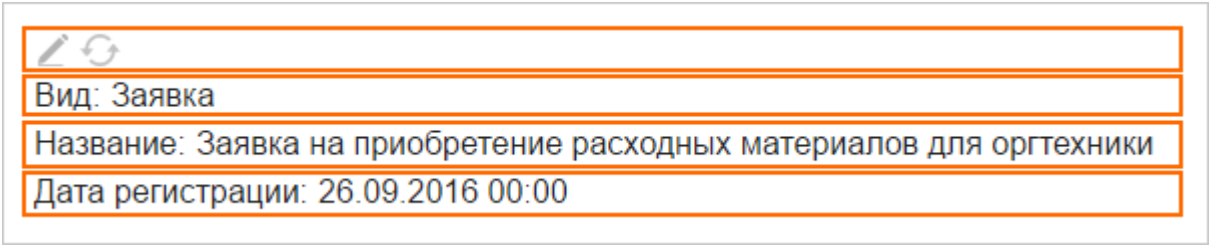


Рисунок 478. Пример простой разметки карточки

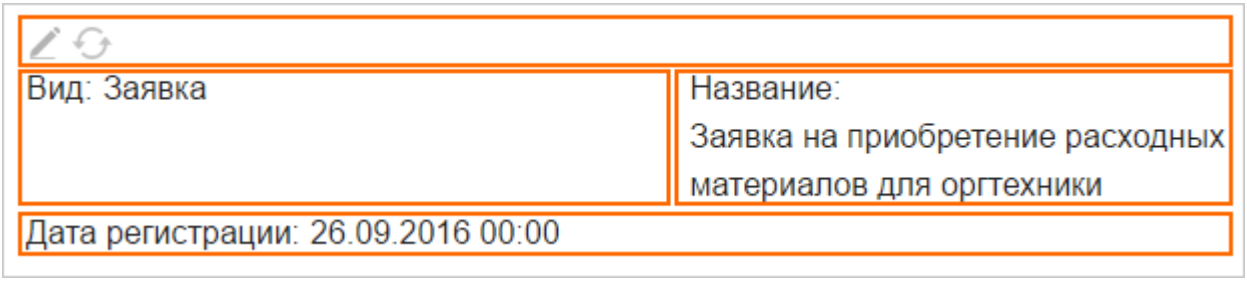


Рисунок 479. Пример разметки карточки с блоками для работы с данными

Адаптивность разметок

Конструктор Web-разметок предоставляет возможность разрабатывать разметки, зависимые от типа пользовательского устройства.

Есть два способа создания адаптивных разметок:

1. Разметка изначально разрабатывается для конкретного типа устройства с учётом размеров экрана и производительности. Затем настраивается условие выбора разметки, в котором учитывается тип устройства.
2. Для всех типов устройств разрабатывается общая разметка. При настройке элементов управления используется механизм адаптации, который позволяет устанавливать разные настройки ЭУ для разных устройств.

Подробнее про настройки адаптивности: Настроить адаптивные свойства разметки.

Локализация разметок

Конструктор Web-разметок предоставляет возможность локализовать разметки, определять для различных текстовых переменных значения в зависимости от языка интерфейса Web-клиента.

Редактор разметок

Редактор разметок предназначен для настройки разметок.

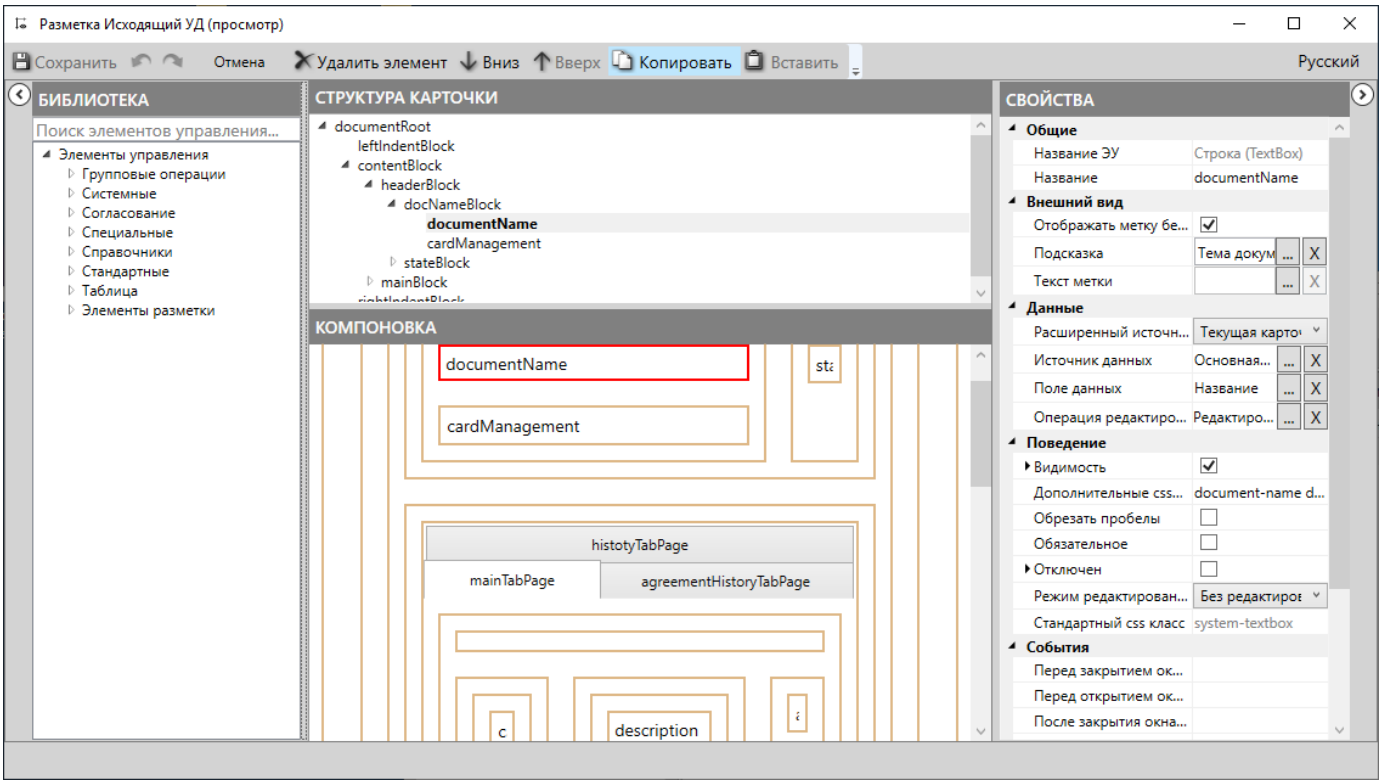


Рисунок 480. Разметка для просмотра карточек вида "Исходящий", открытая в редакторе разметок

Таблица 24. Области, доступные в окне редактора разметок

Библиотека	Коллекция зарегистрированных в Web-клиенте стандартных и разработанных самостоятельно элементов управления. ЭУ из списка могут быть добавлены в разметку. Элементы управления сгруппированы по назначению: *Системные*, *Согласование*, *Справочники* и др.
Структура карточки	Древовидное представление структуры разметки с корневым блоком на вершине.
Компоновка	Упрощенное представление полученной разметки. Блок, соответствующий элементу управления, выбранному в области *Структура карточки*, выделяется красной рамкой.
Свойства	Свойства элемента управления, выбранного в области *Структура карточки* или *Компоновка*.

Области в окне редактора разметок могут быть отображены или скрыты в зависимости от предпочтений.

Таблица 25. Команды на ленте редактора разметок

Сохранить	Сохраняет несохранённые изменения разметки.
	Отменяет / повторяет последнее изменение.
Отмена	Закрывает редактор с запросом на сохранение изменений.
Удалить элемент	Удаляет выбранный элемент управления с разметки.
Вниз / **Вверх**	Перемещает выбранный элемент управления по дереву разметки. Элементы управления могут перемещаться в пределах своего родительского блока.
Копировать и **Вставить**	Копировать или вставить в разметку элемент управления или блок со всем содержимым. При вставке скопированным элементам управления присваиваются новые имена. При копировании сохраняются настройки элементов управления, включая добавленные стили и обработчики событий.

Справа от панели команд отображается язык локализации по умолчанию, на котором ведется быстрый поиск по ресурсам локализации.

Если разметка недоступна для редактирования, команды ленты и компоненты в области *Библиотека* будут скрыты.

Добавить элемент управления в разметку

Чтобы добавить элемент управления в разметку:

1. В области *Библиотека* выберите нужный элемент управления.

   Используйте поиск по библиотеке элементов управления. Поиск осуществляется по названию типа элемента управления и по названию группы элементов управления.

   Поиск осуществляется на двух языках вне зависимости от текущей локализации. Если ввести название ЭУ на английском, поиск выдаст найденные ЭУ с названиями на русском и наоборот.

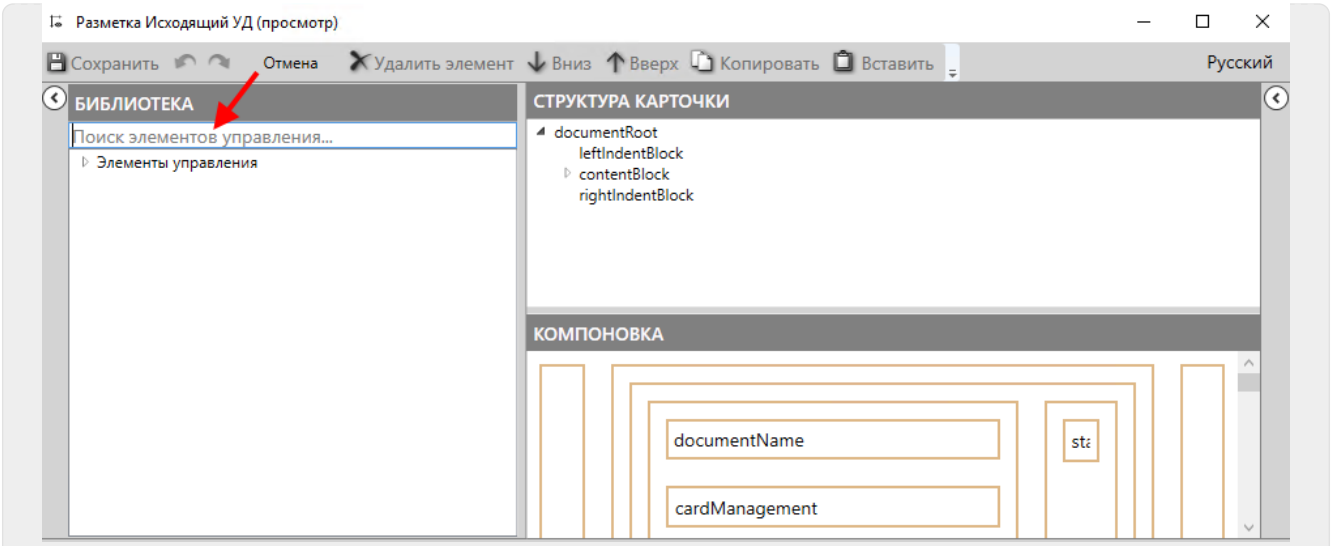


Рисунок 481. Строка поиска элементов управления

2. Перетащите ЭУ из области *Библиотека* в нужное место в области *Структура карточки* или в области *Компоновка*.

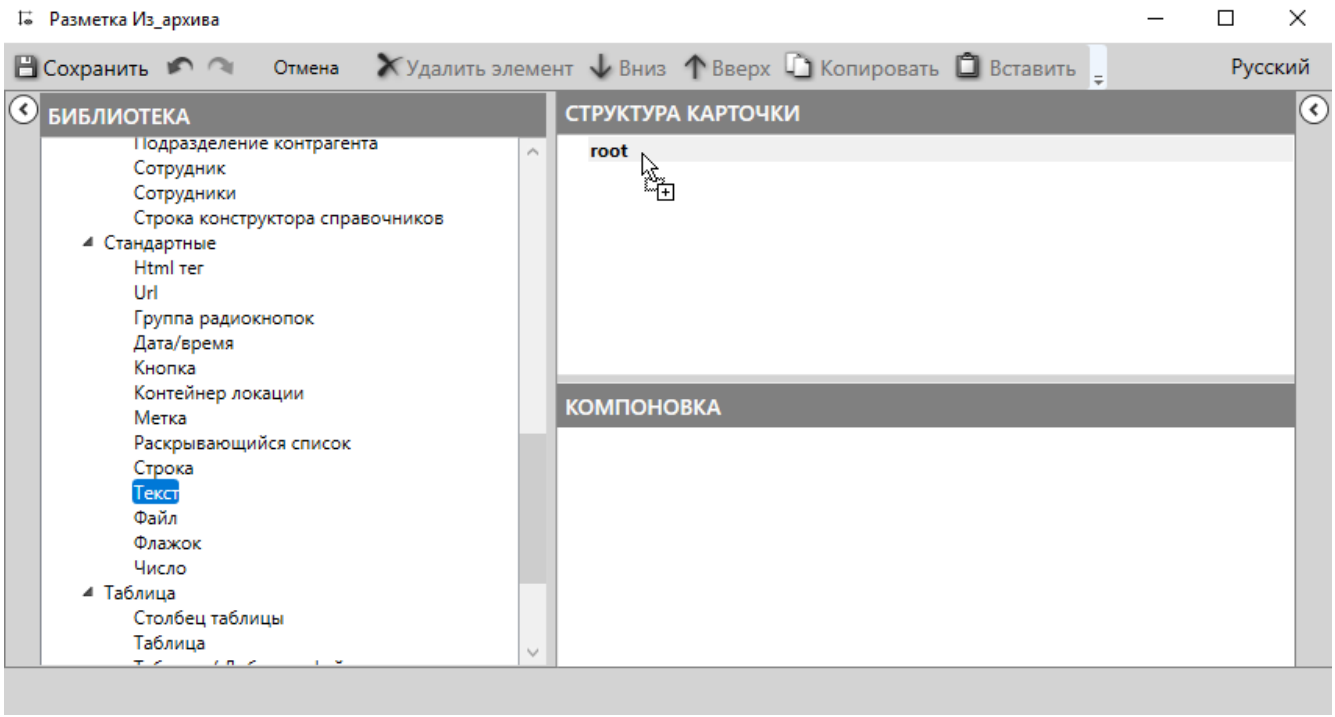


Рисунок 482. Перемещение элемента управления из "Библиотеки" на разметку в Редакторе разметок

3. Элемент управления будет добавлен в соответствующий узел.

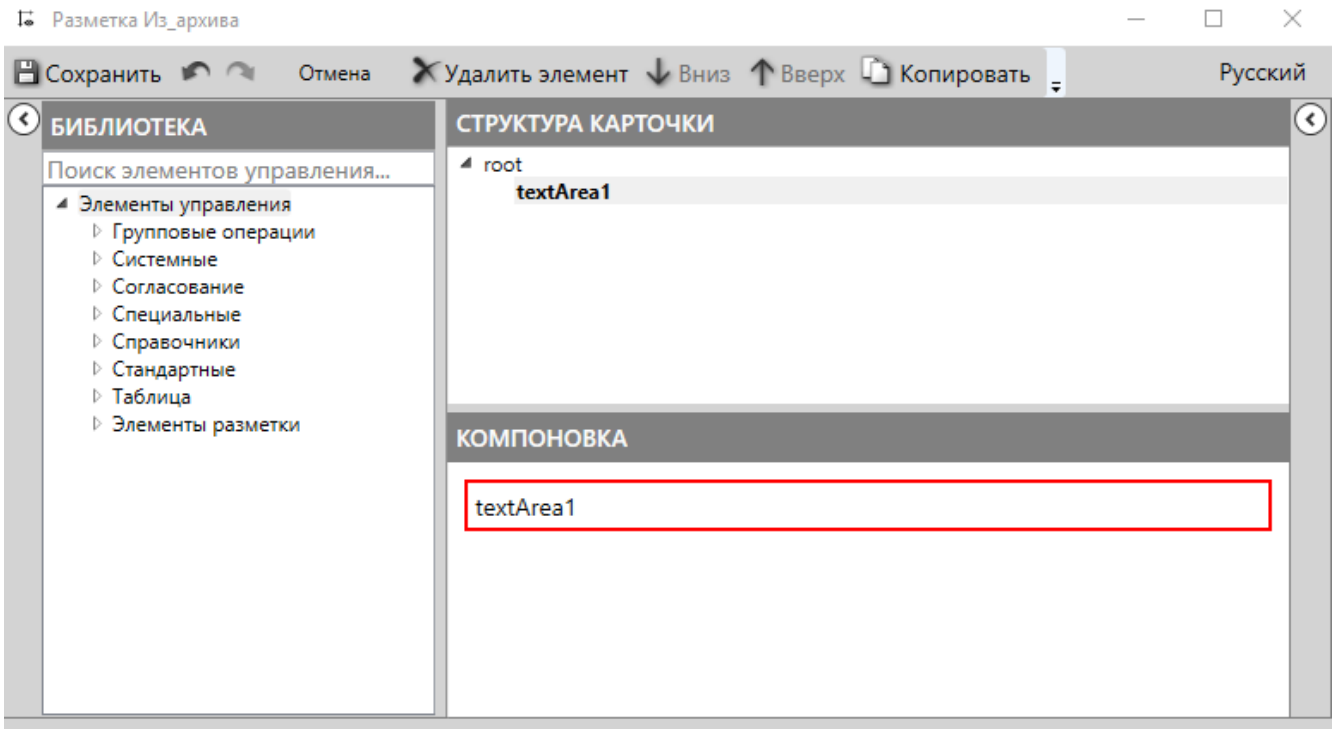


Рисунок 483. Разметка после добавления элемента управления textArea1

4. Настройте элемент управления.

Удалить элемент управления из разметки

Любой элемент разметки, за исключением Корневого блока, может быть удалён. Блока или другой элемент управления, будет удалён вместе с дочерними элементами, если таковые имеются.

Чтобы удалить элемент управления:

1. Выберите элемент управления в области *Структура карточки* или в области *Компоновка*.
2. Нажмите кнопку **Удалить элемент** или вызовите из контекстного меню удаляемого элемента команду **Удалить**.

   Элемент управления и вложенные элементы будут удалены.

Чтобы отменить удаление, нажмите кнопку на ленте.

Копировать элемент управления

Настроенный элемент управления может быть скопирован со всеми настройками в текущую или в другую разметку. Элементы управления будут скопированы вместе с дочерними ЭУ, если таковые имеются.

Чтобы скопировать элемент управления:

1. Выберите в текущей разметке копируемый элемент управления.
2. Нажмите кнопку **Копировать** или вызовите одноименную команду из контекстного меню копируемого элемента.
3. Если нужно, откройте другую разметку.
4. В области *Структура карточки* выберите Блок, в который необходимо вставить элемент управления, или Корневой блок.
5. Нажмите кнопку **Вставить** или вызовите из контекстного меню копируемого элемента команду **Вставить**.

> Элемент управления будет добавлен в выбранный Блок последним. Скопированному элементу управления будет присвоено новое название.

Переместить элемент управления

Чтобы переместить элемент между Блоками:

1. Выберите перемещаемый элемент управления.
2. Перетащите элемент в новый родительский блок.

Изменить порядок элементов управления в блоке

Чтобы изменить порядок отображения элементов управления в блоке:

1. В области *Структура карточки* выберите элемент управления, который нужно передвинуть.
2. Нажмите кнопку **Вверх** / **Вниз** на ленте *Редактора разметок*, или вызовите аналогичную команду из контекстного меню перемещаемого элемента управления. Элемент управления будет перемещен вверх или вниз в своём блоке.

   Если перемещение элемента управления в одном из направлений невозможно, то соответствующая кнопка блокируется.

С помощью свойства **Порядок** элемента управления Блок можно переопределить порядок следования элементов разметки. В этом случае фактическая последовательность отображения элементов управления может отличаться от отображаемой в Редакторе разметок в области *Компоновка*.

Сохранить изменения разметки

1. Чтобы сохранить изменения разметки, нажмите кнопку **Сохранить** на ленте

Редактора разметок.

Web-клиент кэширует разметки. После сохранения изменений разметка, расположенная в кэше, будет автоматически перезагружена. У пользователя новая разметка будет задействована после перезагрузки страницы.

Обработчик событий разметки

Для разметки может быть разработана функция-обработчик, которая будет вызываться при определённом событии разметки и выполнять требуемую задачу. См. Разработка расширения с обработчиками событий.



Название обработчика событий должно состоять из латинских букв в любом регистре и цифр, в качестве разделителей допускается использовать пробелы и следующие символы: `,` , `;` , `|` или `&`. При вводе некоренного названия скрипта появляется предупреждение: `Название события не подходит под заданный формат. Событие будет удалено.`.

Настроить адаптивные свойства разметки

Адаптивность разметок

Конструктор Web-разметок предоставляет возможность разрабатывать разметки, зависимые от типа пользовательского устройства.

Есть два способа создания адаптивных разметок:

1. Разметка изначально разрабатывается для конкретного типа устройства с учётом размеров экрана и производительности. Затем настраивается условие выбора разметки, в котором учитывается тип устройства.
2. Для всех типов устройств разрабатывается общая разметка. При настройке элементов управления используется механизм адаптации, который позволяет устанавливать разные настройки ЭУ для разных устройств.

Второй способ рассмотрен подробнее ниже.

Адаптивные свойства

Для некоторых свойств ЭУ может быть настроена зависимостью от типа клиентского устройства. Например, свойство `Видимость` может принимать значение ***TRUE*** (показывать) при работе на персональном компьютере, но

FALSE (скрывать) при работе на смартфоне или планшете. Такие свойства элементов управления называются **адаптивными**.



В разделе Стандартные элементы управления адаптивные свойства элементов управления отмечены тегом **Адаптивный**

Комбинируя возможности различных адаптивных свойств элементов управления можно добиться оптимального отображения разметки на разных типах устройств.

Если настроенная разметка не отображается для устройства, возможно неправильно определяется UserAgent устройства. Администратор может настроить UserAgent вручную.

Элемент управления "Блок"

Элемент управления "Блок" обладает наиболее полным набором адаптивных свойств, которые позволяют:

- Определять относительную ширину Блока (и его содержимого) для разных типов устройств.
- Скрывать содержимое Блока на определённых устройствах.
- Переопределять порядок отображения Блоков внутри родительского *Блока*.
- Определять способ выравнивания ЭУ на конкретном типе устройств и др.

Например, варьируя ширину блока относительно родительского блока можно установить значение 100% для мобильных устройств и 60% персонального компьютера.

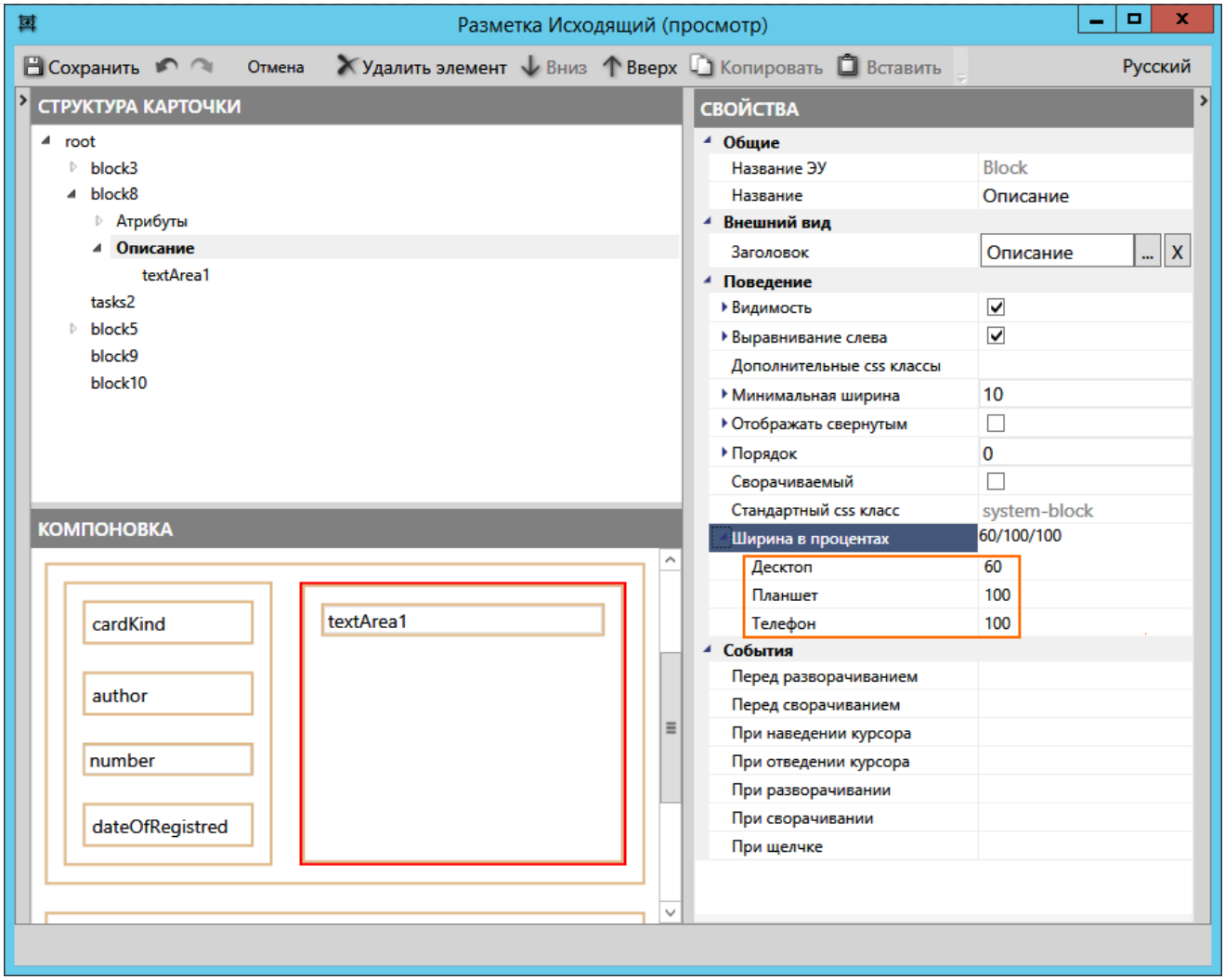


Рисунок 484. Пример настройки относительной ширины Блока для разных типов устройств

При такой настройке содержимое блока на мобильном устройстве займет всю строку разметки.

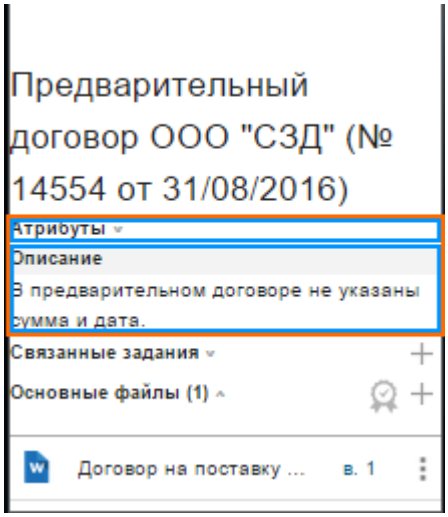


Рисунок 485. Пример размещения пары Блоков с относительной шириной 100% в карточке, открытой на телефоне

На персональном компьютере содержимое блока займет часть разметки. При этом следующей блок может быть размещен последовательно с первым, если их суммарная относительная ширина меньше или равна 100%.

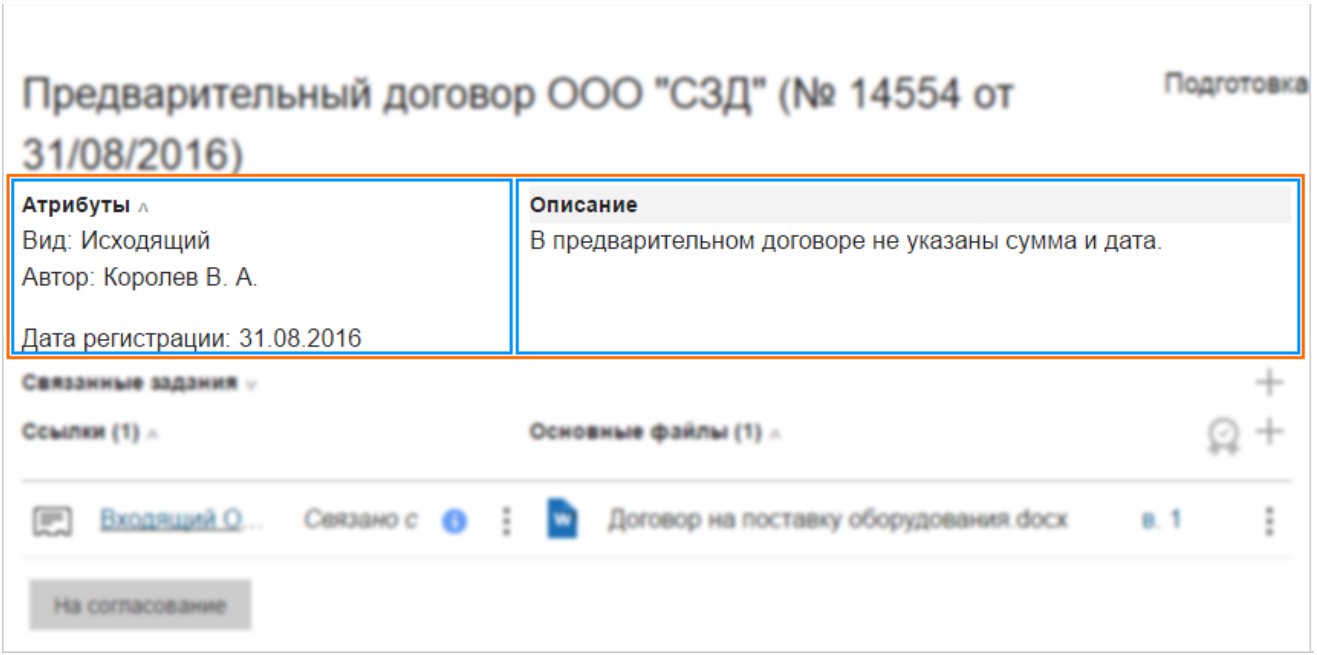


Рисунок 486. Пример размещения пары Блоков в карточке, открытой на персональном компьютере

Порядок отображения блоков в разметке также является адаптивным свойством, что позволяет настроить порядок отображения блоков на разметке с зависимостью от типа устройства.



Рисунок 487. Стандартная последовательность отображения блоков в карточке, открытой на персональном компьютере



Рисунок 488. Изменённая последовательность отображения блоков в карточке, открытой на телефоне

Настроить таблицу для работы с коллекционной секцией карточки

Элемент разметки Таблица предоставляет механизм доступа к данными коллекционной секции карточки.

Чтобы реализовать доступ к данным коллекционной секции карточки:

1. Добавьте в разметку элемент разметки "Таблица". В свойствах данного элемента выберите коллекционную секцию карточки, с которой связывается таблица.
2. Добавьте в таблицу один или несколько элементов разметки "Столбец таблицы". В свойствах столбцов нужно указать их заголовки.
3. Добавьте в каждый столбец элемент управления, свяжите ЭУ с полем секции. Тип добавляемого элемента управления должен соответствовать типу поля секции.
4. Каждый добавленный в столбец элемент управления свяжите с полем секции.

Пример 2. Использование Таблицы для отображения комментариев к документу (секция "Комментарии"):

1. Добавьте в разметку карточки *Документ* в режиме просмотра элемент разметки *Таблица*.

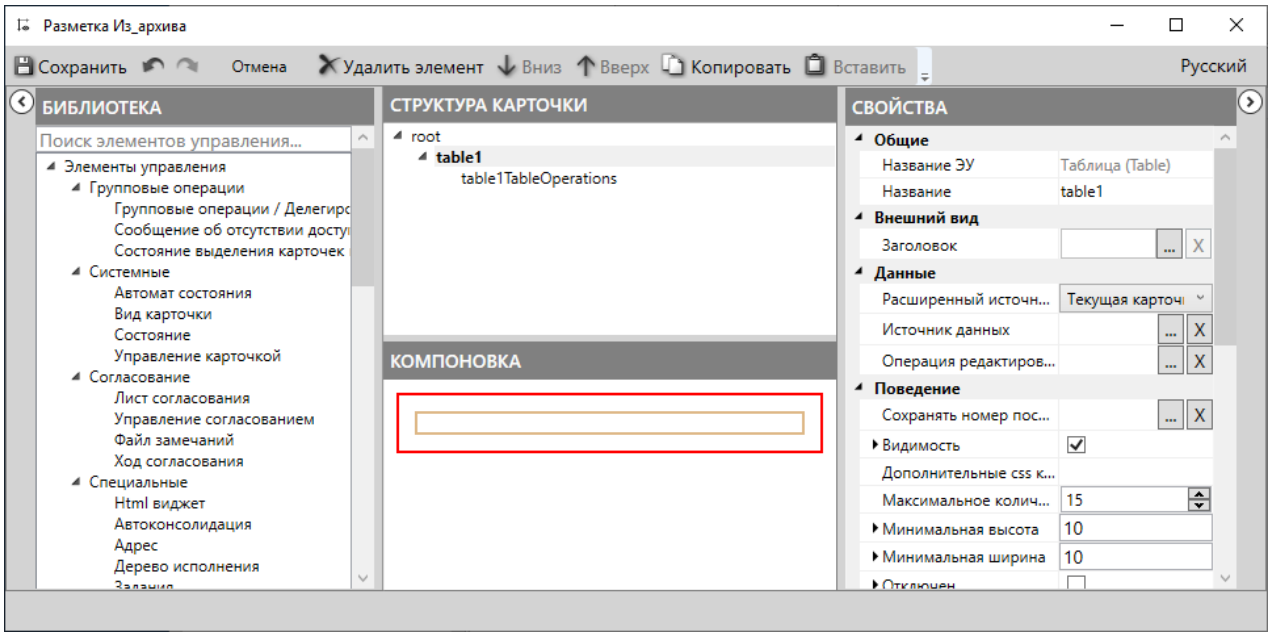


Рисунок 489. Добавление "Таблицы" в разметку

2. Укажите Заголовок.
3. Выберите Источник данных.

Если потребуется, измените другие настройки таблицы.

4. Добавьте в таблицу три (по числу полей в секции "Комментарии") элемента разметки Столбец таблицы.

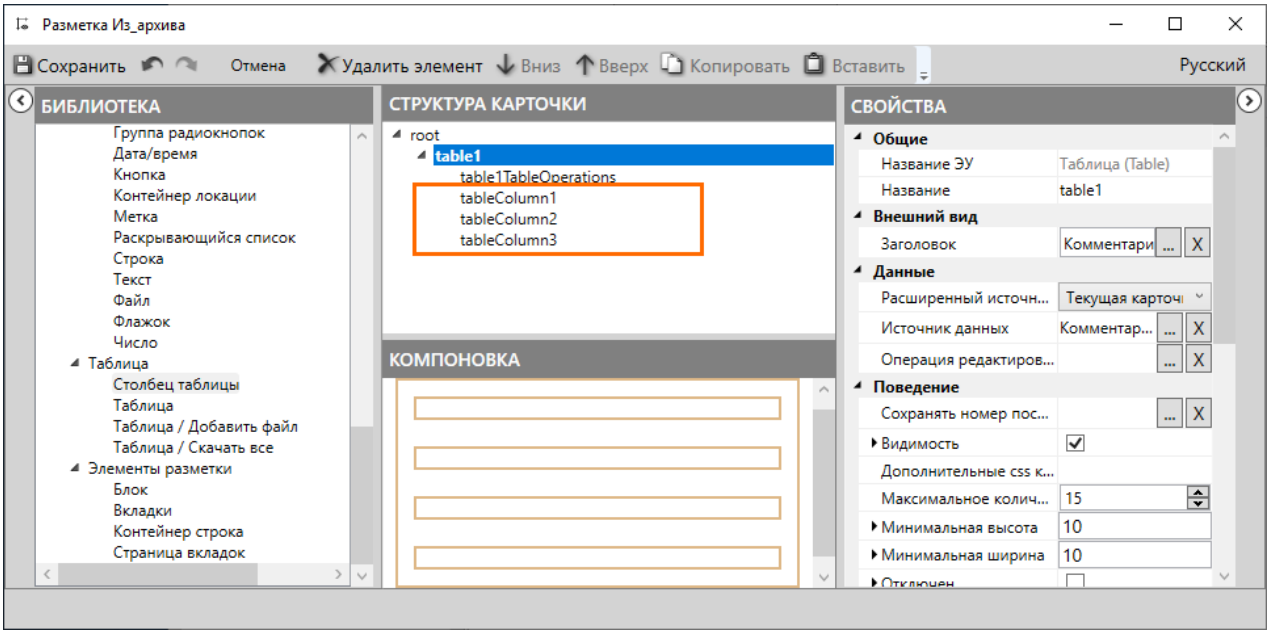


Рисунок 490. Добавление "Столбцов таблицы" в разметку

5. Для каждого столбца задайте Заголовок, соответствующий типу отображаемых данных: *Автор комментария*, *Дата* и *Текст*.
6. Добавьте в столбцы элементы управления, соответствующие типам полей:
   - "Сотрудник" в столбец *Автор комментария*.
   - "Дата/время" в столбец *Дата*.
   - "Строка" или "Текст" в столбец *Текст*.

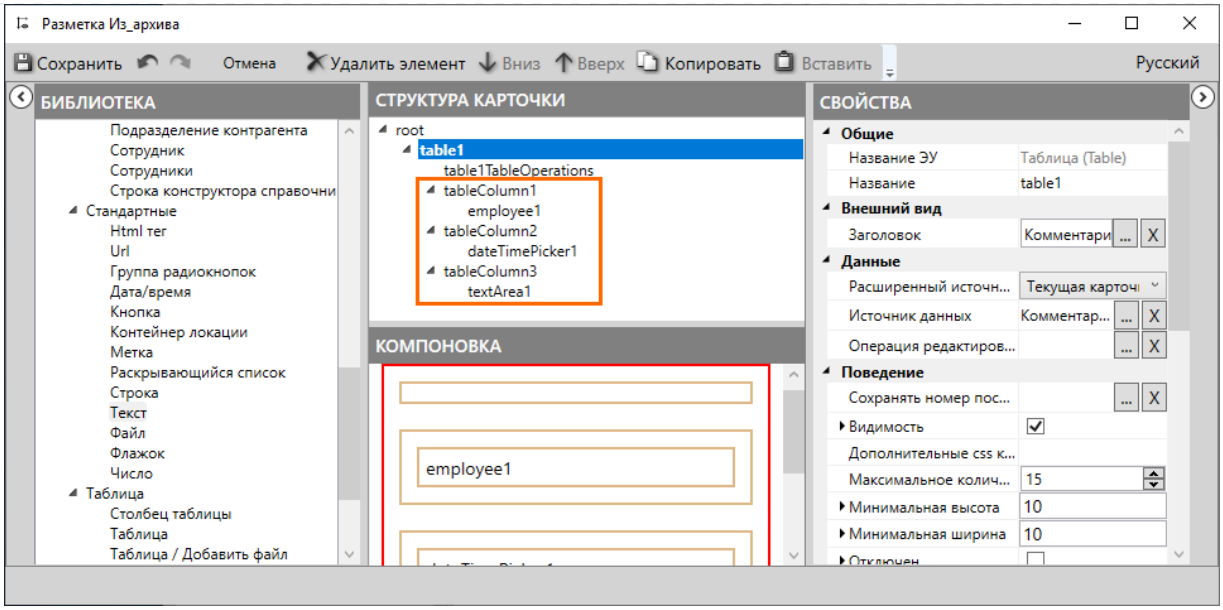


Рисунок 491. Добавление элементов управления в столбцы

Настройте добавленные элементы управления:

1. Для "Сотрудника" выберите Поле данных: *Автор комментария*.
2. Для "Даты комментария" выберите Поле данных: *Дата время комментария*.
3. Для "Содержимого комментария" выберите Поле данных: *Текст комментария*.
4. Сохраните разметку.

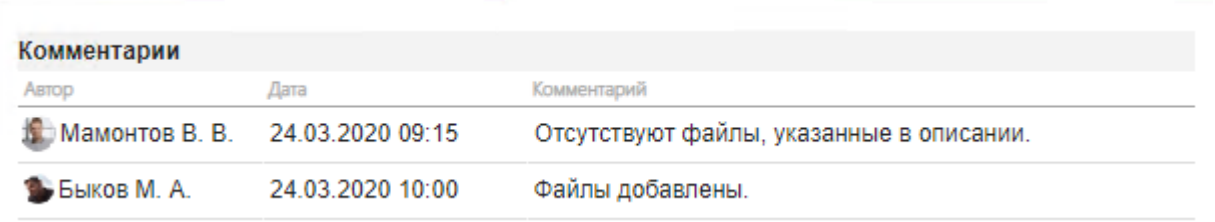


Рисунок 492. Пример отображения настроенной "Таблицы"

Настроить вкладки

Элемент разметки "Вкладки" позволяет формировать логические связи между данными, размещая их на отдельных листах, между которыми можно переключаться.

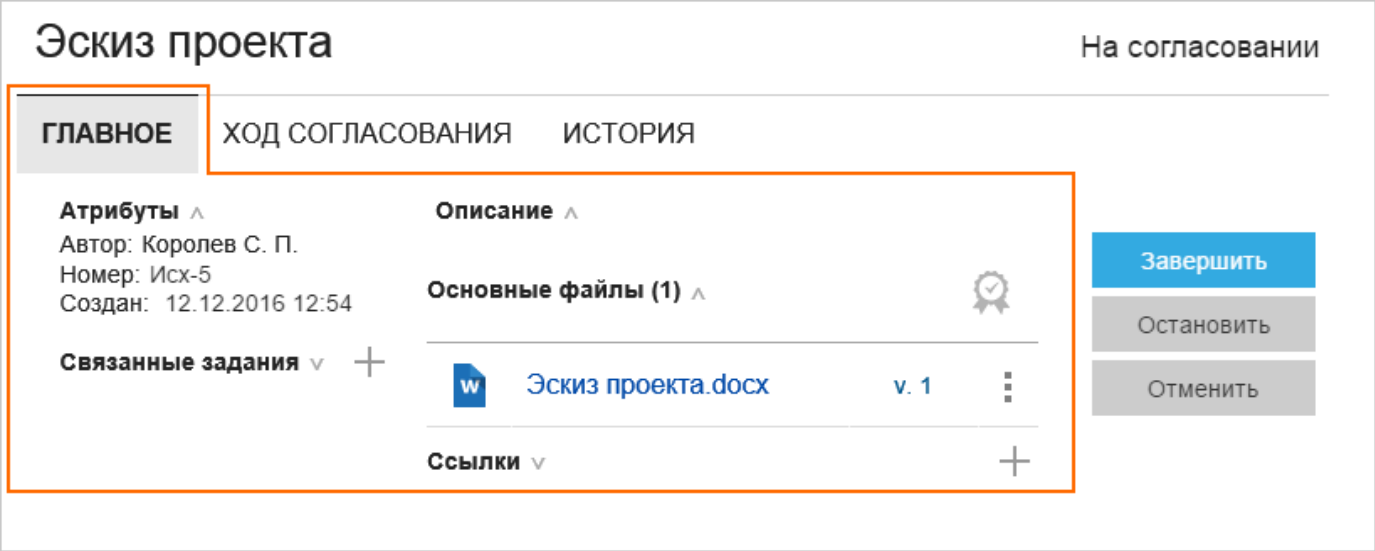


Рисунок 493. Пример листа элемента разметки "Вкладки" с данными карточки

Чтобы настроить разметки с использованием элемента разметки Вкладки*:*

1. В настраиваемую разметку добавьте элемент разметки *Вкладки*.
2. В добавленный элемент вставьте элемент разметки "Страница вкладок".
3. Для каждой страницы укажите Заголовок.
4. На страницах разместите необходимые элементы управления.



Размещайте элементы управления с флагом **Обязательное** только на вкладке открываемой по умолчанию. Элементы управления, размещенные на вкладке, загружаются только после открытия вкладки. Проверить заполненность ЭУ на закрытой вкладке невозможно.

Если требуется обеспечить проверку обязательных элементов управления на вкладке, коротая не открывалась пользователем, разработайте клиентский скрипт со следующим содержимым:

```
function loadTabPages(sender, e)①
{
tabControl.loadTabPage(tabControl.params.tabPages[1]);②
}
```

① Получаем ЭУ Вкладки, на котором размещены вкладка, требующая загрузки.

Вместо ***tabControlName*** нужно указать название своего ЭУ var tabControl

```
= sender.layout.controls["tabControlName"];.
```

② Вызываем метод loadTabPage, который загружает вторую по порядку вкладку (***tabPages[1]***)
В ***tabPages[1]*** вместо 1 укажите порядковый номер вкладки (начинается с 0), которую нужно загрузить. Если требуется загрузить несколько вкладок, скопируйте строку.

Данный скрипт должен быть размещен в каталоге Web-клиента. Метод ***loadTabPages*** нужно указать в качестве обработчика события **После открытия карточки** в корневом блоке разметки. Инструкцию по разработке и публикации клиентских скриптов С чего начать.

Подключение дополнительных источников данных

При настройке разметки по умолчанию предоставляется доступ только к данным текущей карточки. Это значит, что элементы управления, позволяющие отображать и/или изменять данные карточки, могут работать только с секциями и полями данной карточки.

Если требуется, к разметке можно подключить дополнительные источники данных (карточки, связанные с текущей), с которыми смогут работать элементы разметки.

Чтобы подключить дополнительные источники данных:

1. Откройте разметку карточки или справочника.
2. Выберите корневой элемент управления разметки.
3. Нажмите кнопку ... в свойстве **Расширенные источники данных**. Будет открыто окно *Расширенные источники данных*.

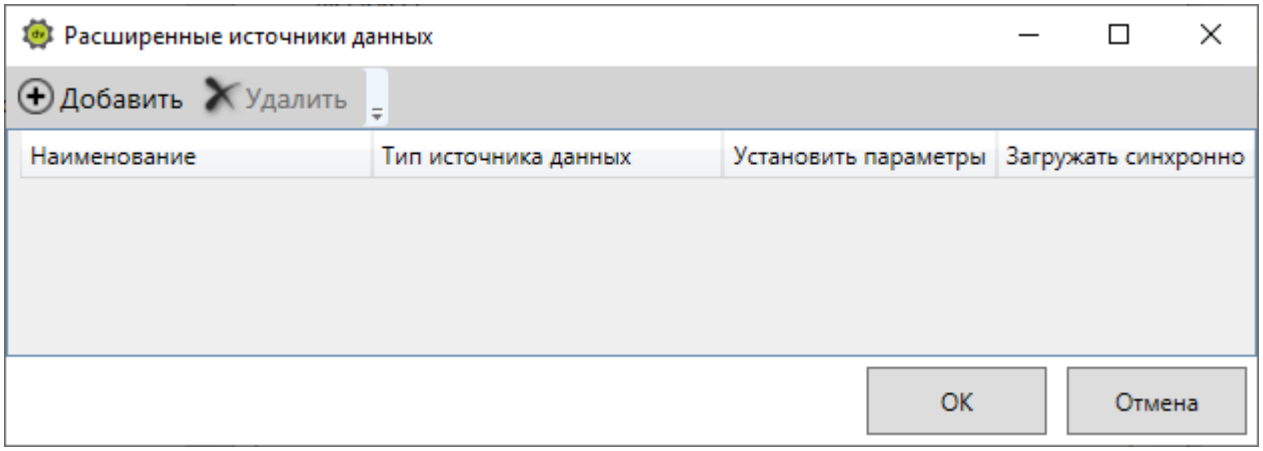


Рисунок 494. Настройка дополнительных источников данных

4. Нажмите кнопку **Добавить**. В список источников будет добавлен новый источник.

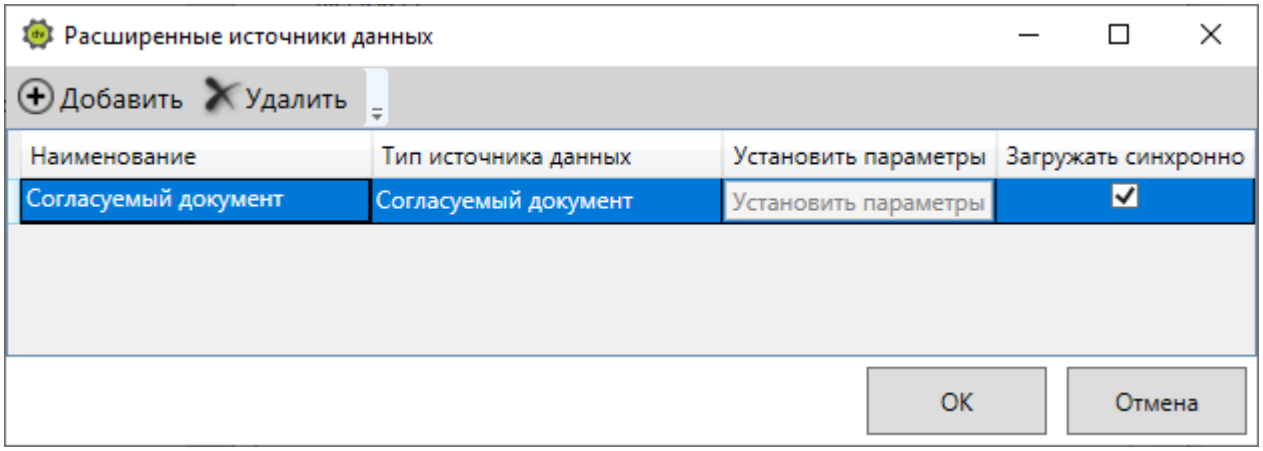


Рисунок 495. Дополнительные источники с добавленным источником

5. В списке **Тип источника данных** выберите тип связанной карточки, которая будет выступать источником дополнительных данных.

Таблица 26. Типы источников данных для карточки:

Для всех локаций:

Ссылка из поля карточки	Источником данных выступает любая карточка, на которую ведёт ссылка из поля текущей карточки. В параметрах получения источника (см. ниже) нужно выбрать поле текущей карточки, содержащее ссылку на карточку, которая будет являться дополнительным источником данных. Данный тип источника может использоваться в любых карточках и справочниках.

Для локаций карточек:

Согласуемый документ	Источником данных выступает Документ, по которому выполняется согласование. Данный тип источника должен использоваться только в разметках карточек Заданий согласования.

Документ для Задания	Источником данных выступает Документ, связанный с заданием (выбирается из списка ссылок). В параметрах получения источника (см. ниже) можно выбрать тип ссылки, указывающей на Документ. Если в списке ссылок несколько карточек Документ с выбранным типом ссылок, дополнительным источником будет выбрана первая найденная карточка. Данный тип источника может использоваться в разметках карточек *Задание* и *Группа Заданий*.
Связанная карточка ГЗ	Источником выступает *Группа заданий* из списка ссылок карточки Задание. В параметрах получения источника (см. ниже) нужно выбрать тип ссылки, указывающей на Группу заданий. Если в списке ссылок несколько карточек Группа заданий с выбранным типом ссылок, дополнительным источником будет являться первая найденная карточка. Данный тип источника предназначен для разметок карточек Задание. Для получения данных из родительской группы заданий нужно выбрать тип ссылки *Связанные карточки*. ====

Для локации Диалог профиля пользователя:

Карточка текущего пользователя	Получает данные из карточки сотрудника.
Строка справочника текущего пользователя	Получает данные из Справочника сотрудников из строки пользователя.

6. Нажмите кнопку **Установить параметры**, чтобы настроить параметры получения дополнительного источника данных. Данная настройка не требуется для типа *Согласуемый документ*.
   - Для типа *Документ для Задания* выберите тип ссылки на Документ, который будет являться источником данных.
   - Для типа *Ссылка из поля карточки* выберите поле текущей карточки, содержащее ссылку на карточку, которая будет являться источником

данных.

7. Нажмите кнопку **ОК**, чтобы сохранить настройки. В колонке *Наименование* будет записано название нового типа источника данных (не редактируемое).
8. Если требуется, включите асинхронную загрузку данных из связанной карточки.

   По умолчанию данные связанной карточки получаются синхронно при открытии текущей карточки. Для включения асинхронного режима снимите флаг `Загружать синхронно`. Элементы управления, получающие данные из асинхронного источника, загружаются асинхронно.

9. Нажмите кнопку **ОК** в окне *Расширенные источники данных* для настроек.

> При настройке элемента управления разметки, поддерживающего получение данных из дополнительных источников (есть свойство `Расширенный источник данных`), добавленный источник выбирается по имени, которое отображается в поле `Наименование` в окне *Расширенные источники данных*.

Настройка элементов управления

Настройка элемента управления может включать следующие операции:

1. Настройка связи элемента управления с данными карточки.
2. Настройка общих свойств элемента управления: текстовой метки, обязательности, видимости и т.д.
3. Настройка обработки событий элемента управления.



> Название обработчика событий должно состоять из латинских букв в любом регистре и цифр, в качестве разделителей допускается использовать пробелы и следующие символы: `.` , `,` , `;` , `|` или `&`. При вводе некоренного названия скрипта появляется предупреждение: `Название события не подходит под заданный формат. Событие будет удалено.`.

Настройка элемента управления выполняется на панели *Свойства* редактора разметок. Настройка становится доступна после выбора элемента управления в области *Структура карточки* или в области *Компоновка* Редактора разметок.

Свойства элементов управления

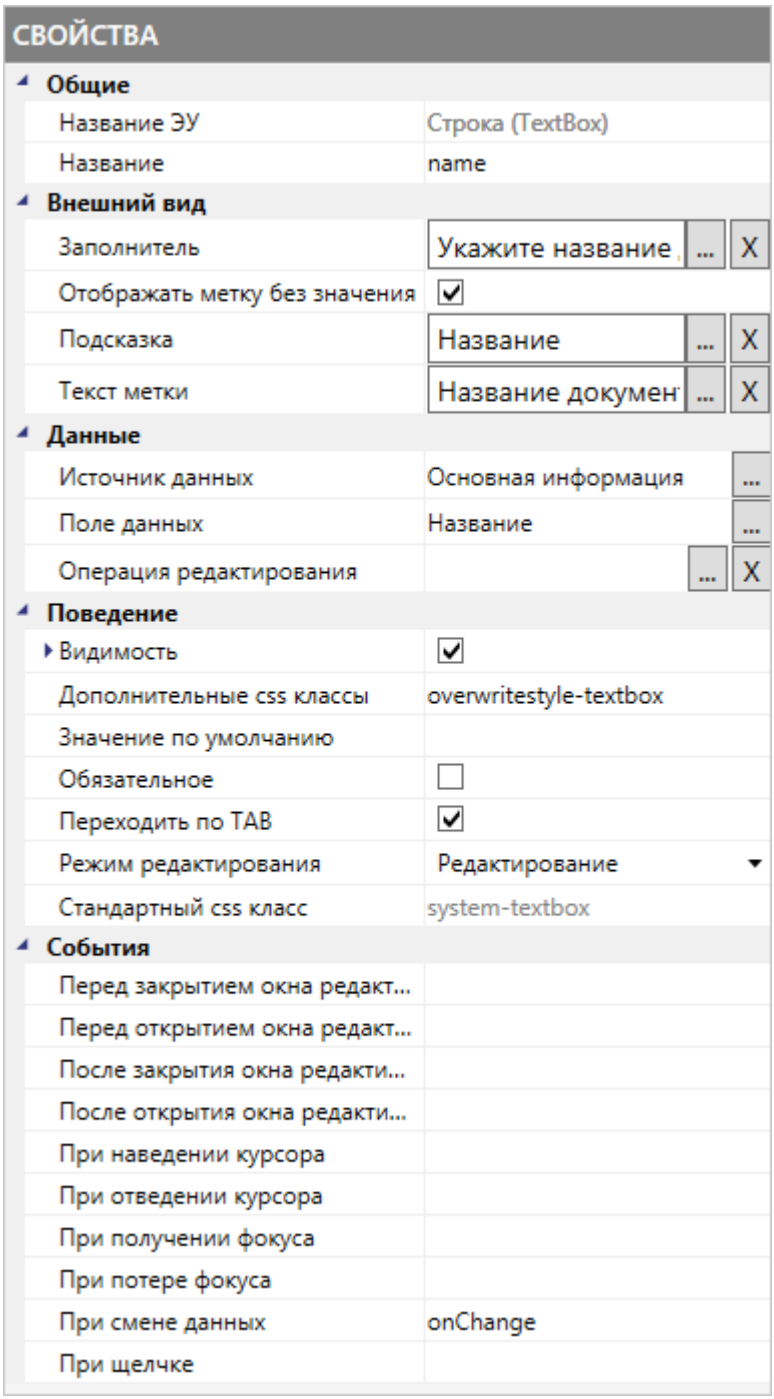


Рисунок 496. Панель свойств элемента управления "Строка"

Таблица 27. Группы свойств элементов управления

Общие	Содержат базовые свойства элементов управления, как правило: `Название ЭУ` (тип элемента управления) и *Название* (название экземпляра элемента управления).

Внешний вид	Содержат настройки, связанные с отображением элемента управления в разметке.
Данные	Содержат настройки связывания элемента управления с данными (например, карточки).
Поведение	Содержат настройки, связанные с работой элемента управления.
События	Содержат перечень событий элемента управления, которые могут быть обработаны в скрипте карточки.



Описание свойств элементов управления приведено в Приложении Стандартные элементы управления.

Некоторые свойства элементов управления могут принимать несколько значений.

Локализуемые свойства

Данные свойства предназначены для локализации разметки. Для каждого языка локализации можно установить собственное значение свойства. В описании элементов управления данные свойства помечены атрибутом `Локализуемый`

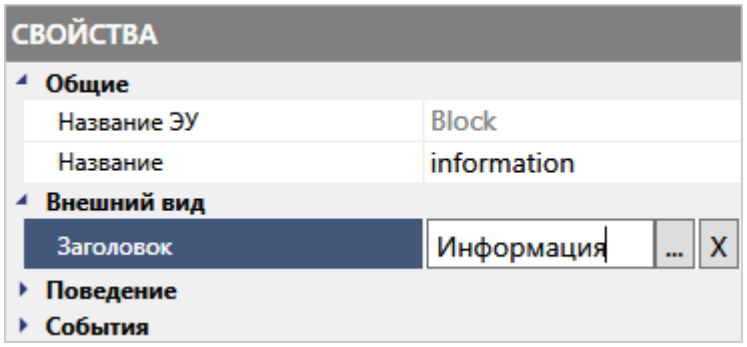


Рисунок 497. Локализуемое свойство "Заголовок"

Адаптивные свойства

Данные свойства предназначены для настройки разметки, зависимой от типа пользовательского устройства. Для каждого типа устройств можно установить собственное значение свойства. В описании элементов управления данные свойства помечены атрибутом `Адаптивный`

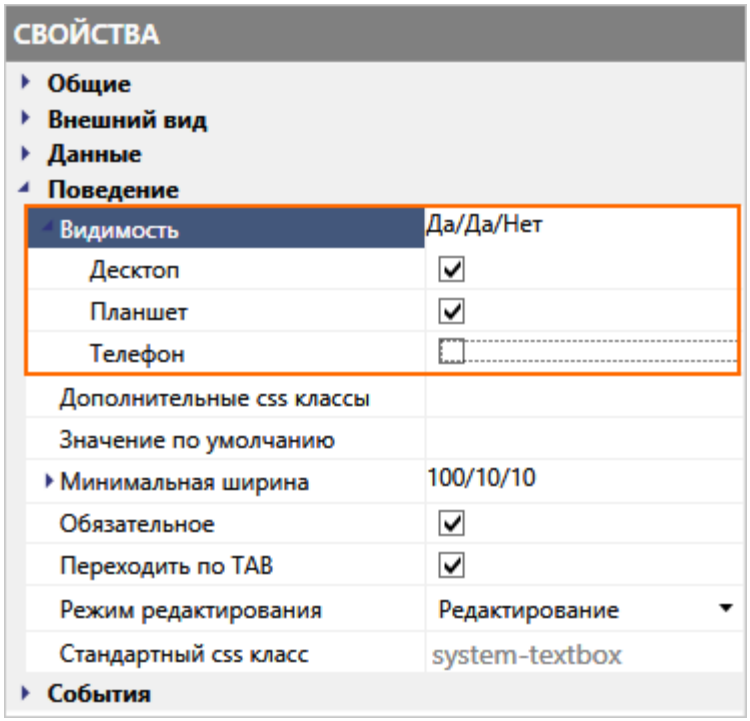


Рисунок 498. Адаптивное свойство "Видимость"

Режим редактирования

Элемент управления, работающий с данными, может иметь данное свойство. Свойство определяет способ изменения значения элемента управления. Подробности в пункте Настройка режима редактирования.

CSS стили элементов управления

У элементов управления есть стиль по умолчанию, который определяет, как они будут выглядеть в браузере пользователя. Стандартное оформление элементов управления можно переопределить с помощью собственного стиля (CSS). Подробности в пункте Изменение стандартного стиля элемента управления.

Как настроить элемент управления

Чтобы настроить ЭУ:

1. Откройте разметку.
2. Выберите элемент разметки в области *Структура карточки* или *Компоновка*. На панели *Свойства* будут отображены настраиваемые свойства элемента управления.
3. В поле Название введите уникальное (для данной разметки) название ЭУ. По данному названию, например, можно получить доступ к элементу управления в скрипте карточки.

4. Если требуется, настройте Текст метки, Подсказку, а также другие свойства ЭУ в разделе *Внешний вид*.
5. Если ЭУ предназначен для работы с данными (например, карточки), выберите источник этих данных. Также можно ограничить возможность изменения значения элемента управления, указав операцию редактирования.
6. Если требуется, настройте свойства в разделе *Поведение*. В частности, здесь можно переопределить стиль элемента управления.
7. Настройте обработку событий элемента управления.



Обязательными для настройки являются свойства: Источник данных и Поле данных. Остальные свойства имеют значения по умолчанию, которые могут быть сохранены.

Режимы редактирования элемента управления

В зависимости от того, в каком режиме открыта карточка — на просмотр или на редактирование, прав пользователя и собственных настроек, элемент управления может работать в одном из трёх режимов редактирования:

- Редактирование — элемент управления позволяет изменять значения.
- Просмотр — элемент управления отображает значение и не позволяет изменять его.
- Редактирование по месту — аналогичен режиму *Просмотр*, но с возможностью изменения значения в отдельном диалоговом окне, открываемом при нажатии на элемент управления.

Режим редактирования выбирается в одноименном свойстве элемента управления.

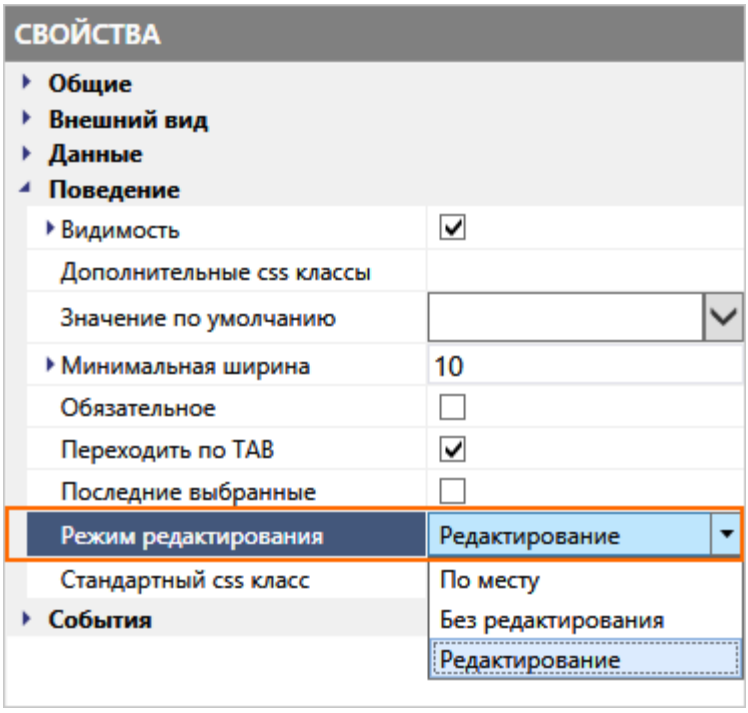


Рисунок 499. Свойства элемента управления "Строка"

Возможность выбора режима редактирования поддерживается не всеми элементами управления.

Выбрать источник данных

Все элементы управления библиотеки стандартных элементов можно разделить на три группы:

1. Элементы управления, которые не связаны с данными карточки.

   Например, "Метка" предназначена для отображения заданного текста, не связанного с данными карточки.

2. Элементы управления, связанные с предусмотренными данными карточки.

   Например, "Вид карточки" связан с полем системной секции карточки, и не требует дополнительной настройки.

3. Элементы управления, которые могут быть связаны с данными карточки.

   Например, "Текст", который может быть связан с различными строковыми данными карточки. Если данные загружаются в ЭУ скриптом и аналогичным образом сохраняются (к примеру, в карточке), настройка связи с данными может не выполняться.

   Если в разметке находятся два ЭУ, имеющих один источник

данных, сохраняться будет всегда состояние того ЭУ, который считан при сохранении последним. Одновременная работа двух ЭУ, привязанных к одному и тому же полю и секции не поддерживается.

Чтобы связать элемент управления с данными карточки:

1. Откройте свойства настраиваемого элемента управления.
2. В разделе *Данные* нажмите кнопку ... в поле Источник данных.

   Будет открыт список всех секций настраиваемого типа карточек.
3. Выберите секцию, содержащую связываемые данные, и нажмите **ОК**.



Если секция является коллекционной, а элемент управления не предназначен для работы с несколькими значениями, то источником данных будет являться первая строка секции.

4. Нажмите кнопку ... в поле Поле данных.

   Будет открыт список метаданных карточки, имеющих тип, с которым может быть связан элемент управления.
5. Выберите связываемое поле и нажмите **ОК**.

Помимо текущей карточки данные могут быть получены из связанной карточки. Подробности в пункте Подключение дополнительных источников данных.

Локализовать свойства элемента управления

Локализация элемента управления подразумевает локализацию его свойств. В свойствах, поддерживающих локализацию, хранятся ссылки на локализованные ресурсы. Локализованные ресурсы — это объекты, содержащие массив значений для различных языков локализации. При отображении элемента управления в веб-браузере, его локализованные свойства принимают значения, определяемые текущим языком пользовательского интерфейса.

К локализуемым свойствам относятся свойства с текстовыми статичными значениями: всплывающая подсказка, метка, заполнитель и др. Локализуемые свойства обозначены тегом Локализуемое в *Описании стандартных элементов управления*.

Чтобы настроить локализуемое свойство элемента управления:

1. Выберите в разметке настраиваемый элемент управления.
2. Выберите локализуемое свойство.
3. Введите значение непосредственно в поле для ввода значения. По мере ввода значения будут предложены существующие локализованные ресурсы. При введение текста в локализуемое поле, текст вставляется в язык по умолчанию. После завершения ввода нажмите клавишу `Enter`.

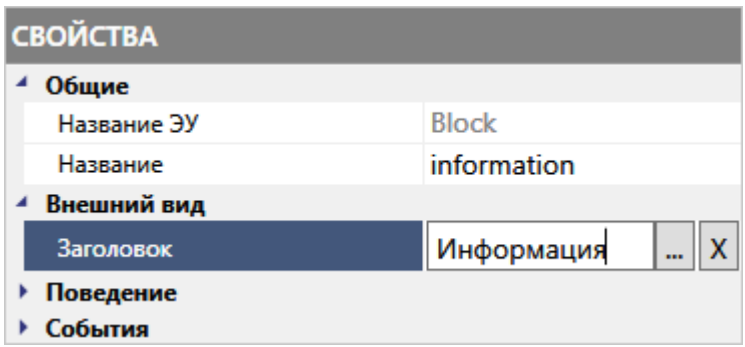


Рисунок 500. Установка значения локализуемого свойства "Заголовок"

Если ресурс с аналогичным названием существует, будет выбран он, иначе создан новый.

При создании нового ресурса подобным образом локализованные значения будут указаны для английского языка и языка локализации по умолчанию (см. Управление локализованными ресурсами). Для других языков будет использовано пустое значение. Необходимо самостоятельно настроить значения для требуемых языков локализации с помощью Панели управления ресурсами локализации.

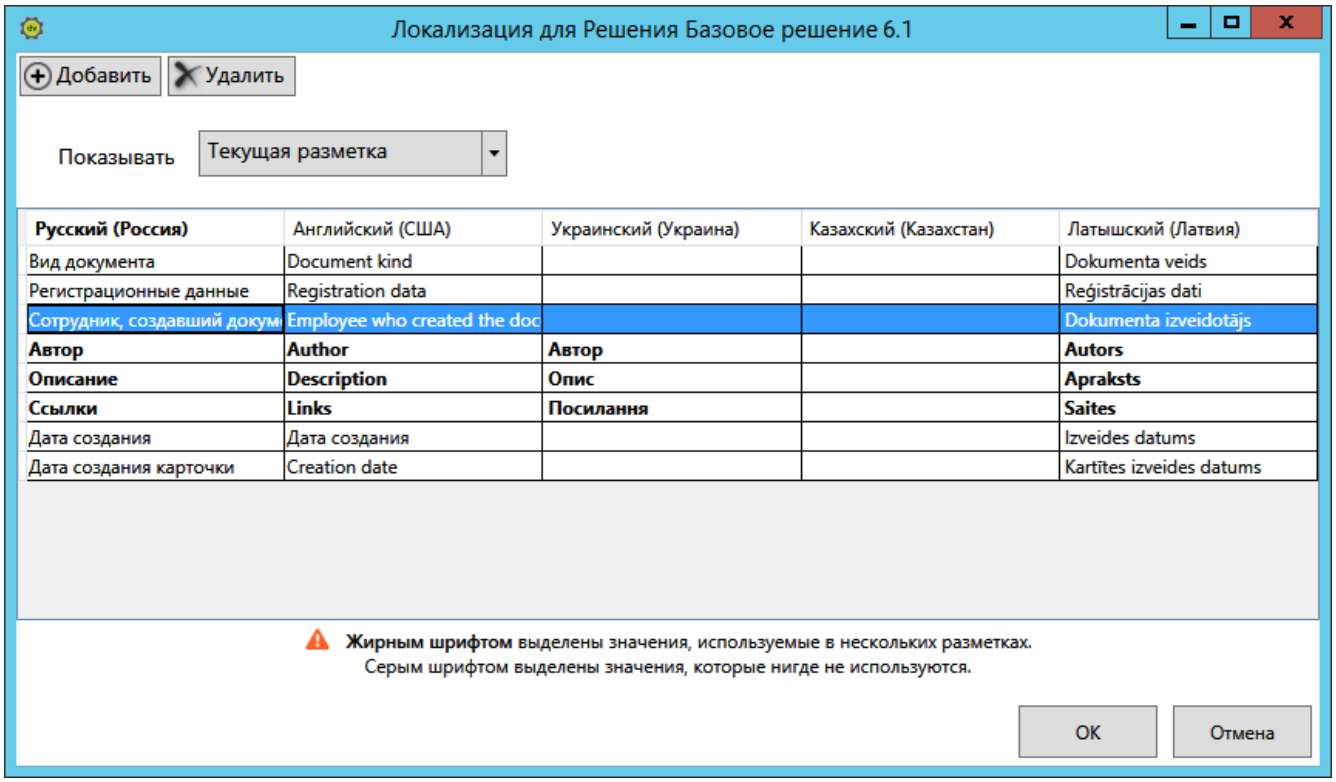


Рисунок 501. Выбор значения локализуемого свойства ЭУ из списка локализованных ресурсов

Подробнее работа с панелью управления ресурсами локализации рассмотрена в пункте Управление локализованными ресурсами.

При изменении локализованного ресурса, изменения затронут также все элементы управления, которые его используют.

Настроить адаптивные свойства элементов управления

Адаптивное свойство — свойство элемента управления, значение которого зависит от типа пользовательского устройства. Использование адаптивных свойств в разметке является альтернативой созданию отдельных разметок для каждого типа клиентского устройства.

Чтобы настроить адаптивное свойство элемента управления:

1. Выберите в разметке настраиваемый элемент управления.
2. Установите значение непосредственно в поле для ввода значения. Данное значение свойства будет использоваться на всех типах устройств.

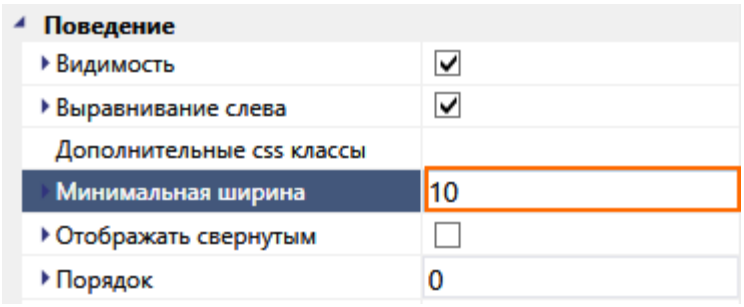


Рисунок 502. Установка значения свойства "Минимальная ширина", общего для всех типов устройств

3. Нажмите кнопку ▸ и введите значения свойства отдельно для каждого типа устройств.

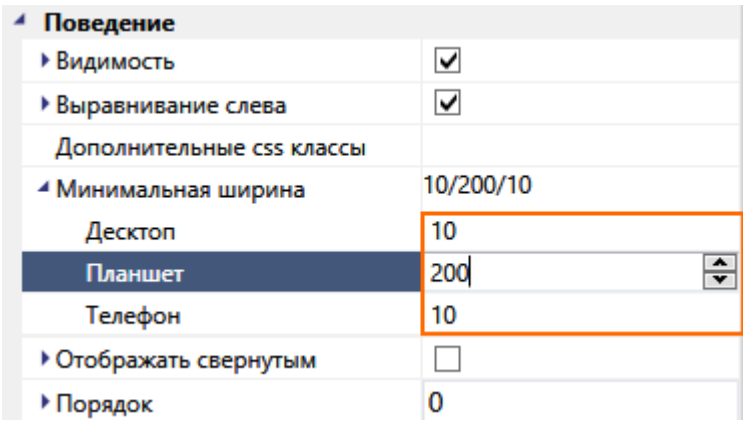


Рисунок 503. Установка значения свойства "Минимальная ширина" для типов устройств

После ввода индивидуальных значений свойства, возможность изменения общего значения блокируется.

Настроить операции редактирования

Если для элемента управления выбрана операция редактирования, при открытии карточки будет выполнена проверка доступности данной операции пользователю по ролевой модели. Если операция разрешена, изменение данных будет разрешено; если операция запрещена, изменение данных будет также запрещено.

Также редактирование может быть запрещено настройкой *Отключен*.

Операция редактирования может наследоваться от родительского Блока, если для элемента управления не указана собственная операция редактирования. В данном случае название настройки *Операция редактирования* меняется на *Операция редактирования (наследовано)*. Наследование может быть прервано

выбором собственной операции редактирования или очисткой значения настройки.

Чтобы выбрать операцию редактирования:

1. Откройте свойства настраиваемого элемента управления.
2. В разделе *Данные*, в поле Операция редактирования нажмите кнопку ....

   Будет открыт список операций, зарегистрированных в *Конструкторе состояний* Docsvision для текущего вида карточки.

3. Выберите операцию редактирования и нажмите **ОК**.

Изменить стандартный стиль элемента управления

Стандартный стиль элемента управления можно переопределить с помощью собственной таблицы стилей. Инструкцию по изменению стандартного стиля см. в пункте Переопределение стиля элемента управления.

Добавить обработчик события элемента управления

Для элемента управления может быть разработана функция-обработчик, которая будет вызываться при определённом событии элемента управления и выполнять требуемую задачу. Инструкцию по разработке обработчиков см. в пункте Разработка расширения с обработчиками событий.



Название обработчика событий должно состоять из латинских букв в любом регистре и цифр, в качестве разделителей допускается использовать пробелы и следующие символы: `.` , `,` , `;` , `|` или `&`. При вводе некоренного названия скрипта появляется предупреждение: `Название события не подходит под заданный формат. Событие будет удалено.`.

Настройка условий выбора разметок

Условия выбора разметки — это правила выбора разметок из общего списка. Согласно настроенным правилам выбирается разметка, которая должна применяться для данной локации в текущих условиях.

Условия выбора настраиваются на вкладке Условия выбора рабочей области.

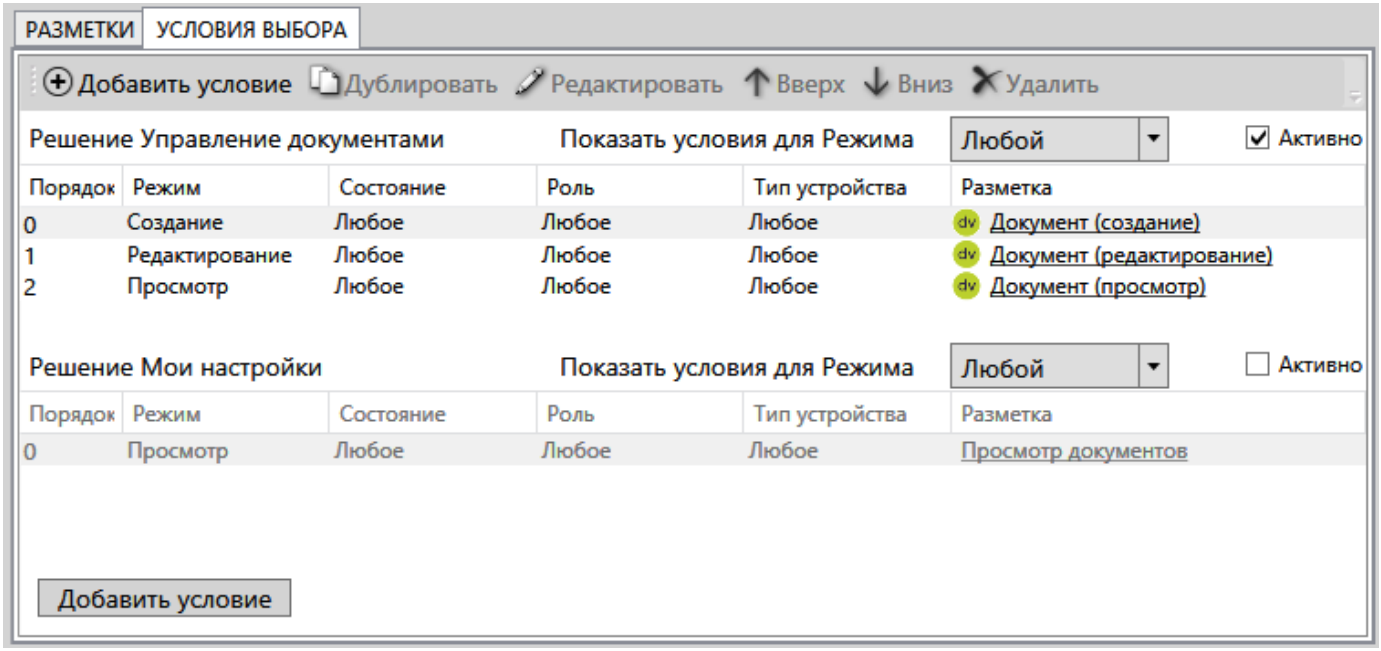


Рисунок 504. Условия выбора разметки

Выбор разметки осуществляется по следующему алгоритму:

1. Определяется список условий, удовлетворяющих заданным критериям.

   Критерии для определения условия зависят от локации, к которой применяется разметка.

   Стандартный набор критериев включает:

   - Тип устройства. Ограничивает использование разметки определённым типом устройства.
   - Режим. Ограничивает использование разметки определённым режимом работы с данными.

Для локаций карточек также могут быть следующие критерии:

- Состояние (карточки).
- Роль (пользователя) и др.

2. Из полученного списка условий выбирается условие с наивысшим приоритетом. Приоритет условия определяется по его порядку в списке: чем меньше порядок, тем выше приоритет.

Приоритет условия выбора разметки

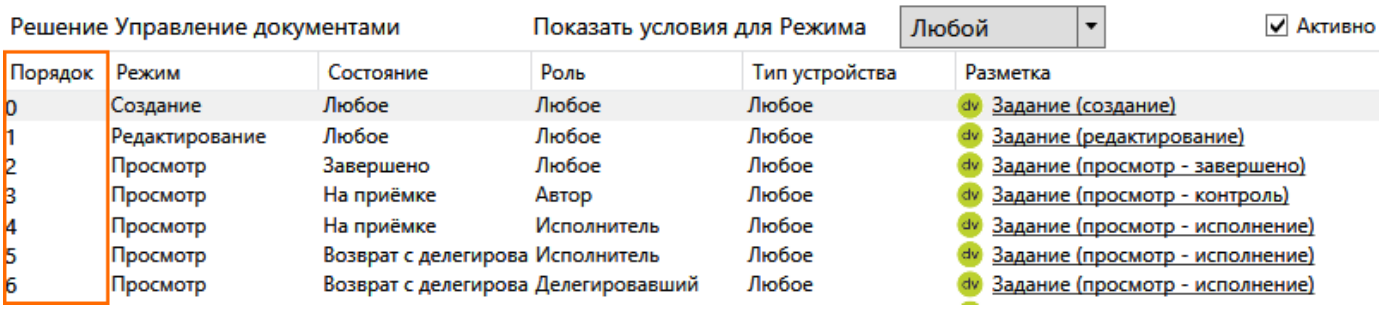
Условия выбора могут быть составлены таким образом, чтобы для каждой ситуации использования разметки было собственное условие. Каждое следующее условие должно содержать критерии, отличные от критериев

предыдущих условий.

Однако, в большинстве случаев такая настройка является излишней. Достаточно указать несколько общих условий для разных режимов использования разметки и также несколько более жестких условий. Например, общие условия не учитывающие тип устройства и/или роль пользователя и более жёсткие условия, учитывающих состояние карточки.

В таком случае разметка с более мягкими условиями будет использоваться, если более жесткие условия не выполнены. При таком подходе важно правильно настроить порядок проверки условий, чтобы более мягкое условие не было проверено до более жесткого условия, иначе оно не будет выполнено ни в каких ситуациях.

Настроить порядок проверки условий можно с помощью кнопок **Вверх** / **Вниз** на странице Условия выбора. Условия, у которых порядок ниже, будут проверяться первыми.


Решение Управление документами Показать условия для Режима: Любой ☑ Активно

Порядок	Режим	Состояние	Роль	Тип устройства	Разметка
0	Создание	Любое	Любое	Любое	Задание (создание)
1	Редактирование	Любое	Любое	Любое	Задание (редактирование)
2	Просмотр	Завершено	Любое	Любое	Задание (просмотр - завершено)
3	Просмотр	На приёмке	Автор	Любое	Задание (просмотр - контроль)
4	Просмотр	На приёмке	Исполнитель	Любое	Задание (просмотр - исполнение)
5	Просмотр	Возврат с делегирова	Исполнитель	Любое	Задание (просмотр - исполнение)
6	Просмотр	Возврат с делегирова	Делегировавший	Любое	Задание (просмотр - исполнение)

Рисунок 505. Порядок проверки условий при выборе разметки

Группировка условий

Все условия на вкладке *Условия выбора* сгруппированы по Решениям, в которые они входят.

Действующим является только одна группа условий, у которой установлен флаг Активно. Условия, включённые в другие Решения, не используются.

Добавить условия выбора разметки

1. Выберите локацию, для которой настраивается условие выбора, и Решение, в которое оно будет включено.

   После создания условия Решение не может быть изменено.

2. В рабочей области откройте вкладку *Условия выбора*.

3. Нажмите кнопку **Добавить условие** на ленте команд.

   Будет открыто диалоговое окно добавления условия выбора.

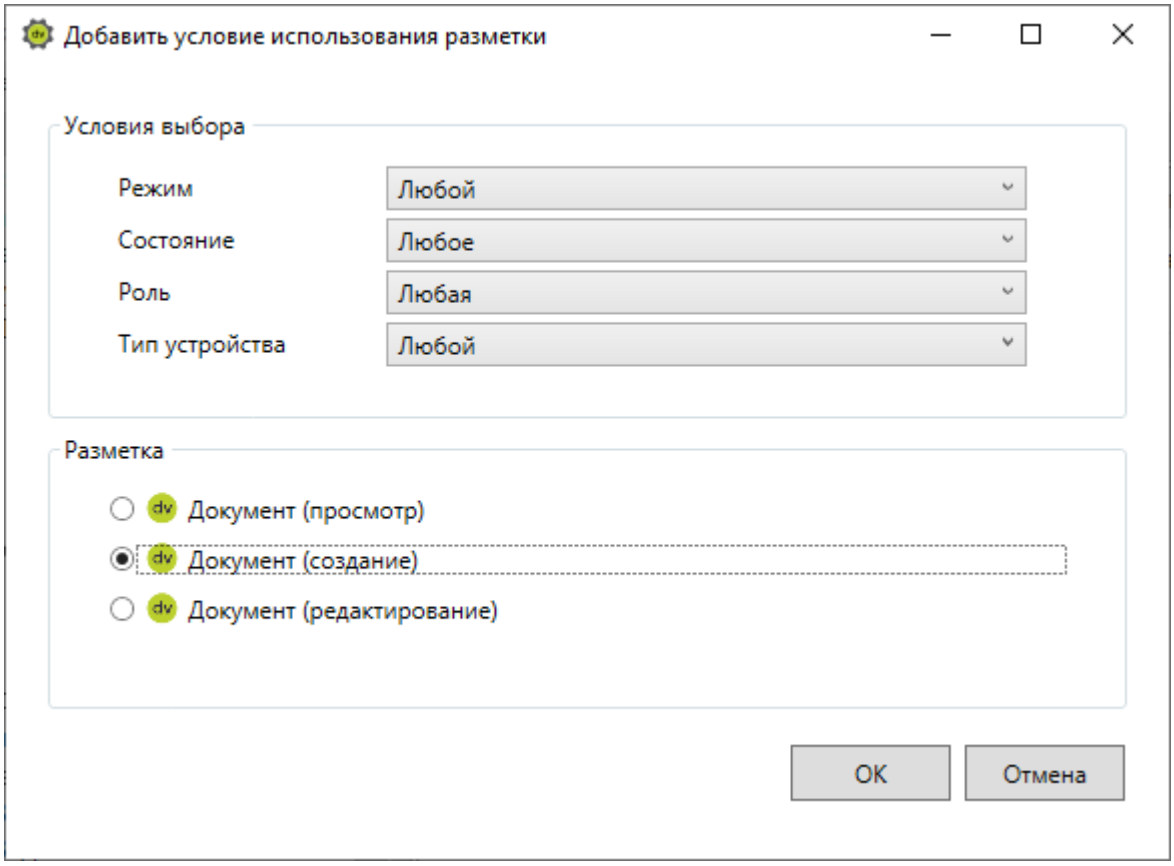


Рисунок 506. Диалоговое окно создания разметки

4. В блоке *Условия выбора* настройте критерии выполнения данного условия.

   Каждая локация обладает собственным списком критериев, которые могут использоваться для настройки условий выбора. К примеру, для группы локаций *BackOffice Cards* предоставляются следующие критерии:

   Режим (открытия карточки)

   Любой

   Разметка может использоваться в любой ситуации: создании, просмотре, редактировании объекта/карточки и т. д.

   Внутри другой разметки

   Разметка может использоваться для отображения данных карточки в другой карточке с использованием Контейнера локации.

   Справка

   Разметка должна использоваться при отображении данных в блоке

справке о ссылке в элементе управления Ссылки.

Описание

Разметка должна использоваться при отображении описания ссылки в таблице ссылок элемента управления Ссылки.

Просмотр

Разметка должна использоваться при отображении объекта/карточки.

Редактирование

Разметка должна использоваться при изменении объекта/карточки.

Создание

Разметка должна использоваться при создании объекта/карточки.

Состояние (карточки)

Роль

Роль пользователя при работе с карточкой.

Поддерживаются следующие роли:

- Администратор,
- Администратор безопасности,
- Оператор архива,
- Продвинутый пользователь,
- Пользователь.

  Если на сервере Docsvision пользователь состоит в соответствующей группе, ему присваивается соответствующая роль.

Тип устройства

Тип пользовательского устройства.

5. В блоке *Разметка* выберите разметку, которая должна использоваться при указанных условиях.
6. Нажмите **ОК** для сохранения. Условие выбора будет добавлено в список условий текущего Решения.
7. Активируйте группу условия, чтобы оно применялось Web-клиентом.

Изменить условия выбора разметки

Чтобы изменить условие выбора разметки:

1. Выберите локацию, содержащую настраиваемое условие.
2. Откройте вкладку "Условия выбора".
3. В списке **Текущее решение** выберите решение, в которое включено настраиваемое условие.
4. Выберите условие и нажмите кнопку **Редактировать**.

> Двойное нажатие по условию выбора также открывает его на редактирование.

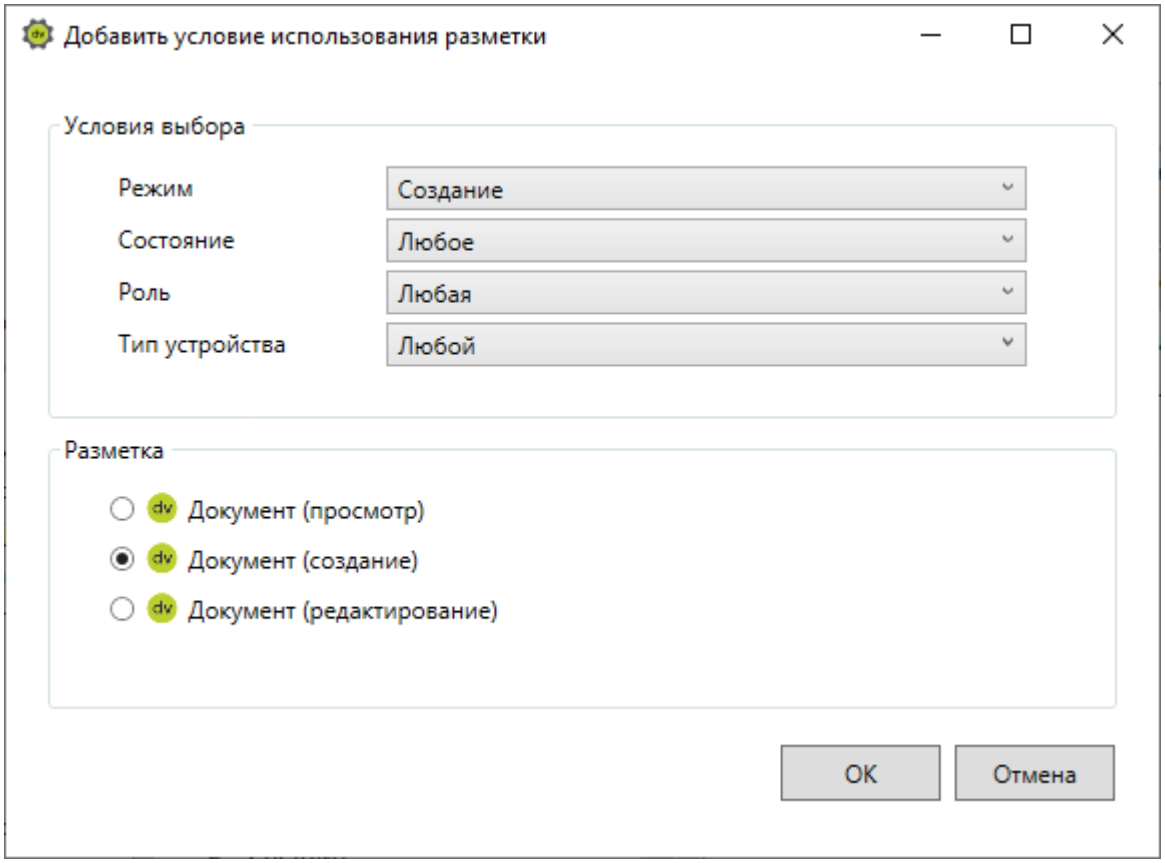


Рисунок 507. Окно редактирования условия выбора разметки

5. Измените настройки условия и нажмите **ОК**.

Удалить условия выбора разметки

Чтобы удалить условие выбора разметки:

1. Выберите локацию, содержащую настраиваемое условие.
2. Откройте вкладку "Условия выбора".

3. В списке *Текущее решение* выберите решение, в которое включено удаляемое условие.
4. Выберите условие и нажмите кнопку **Удалить**.
5. Условие будет удалено без предупреждения.

Копировать условия выбора разметки

Чтобы создать дубликат условия выбора:

1. Выберите локацию, содержащую копируемое условие.
2. Откройте вкладку Условия выбора.
3. В списке **Текущее решение** выберите решение, в которое включено копируемое условие.
4. Выберите условие и нажмите кнопку **Дублировать**.
5. В список условий будет добавлен дубликат условия, выделенный красным цветом.

Изменить порядок условий выбора разметки

Чтобы изменить приоритет условия:

1. Откройте вкладку Условия выбора.

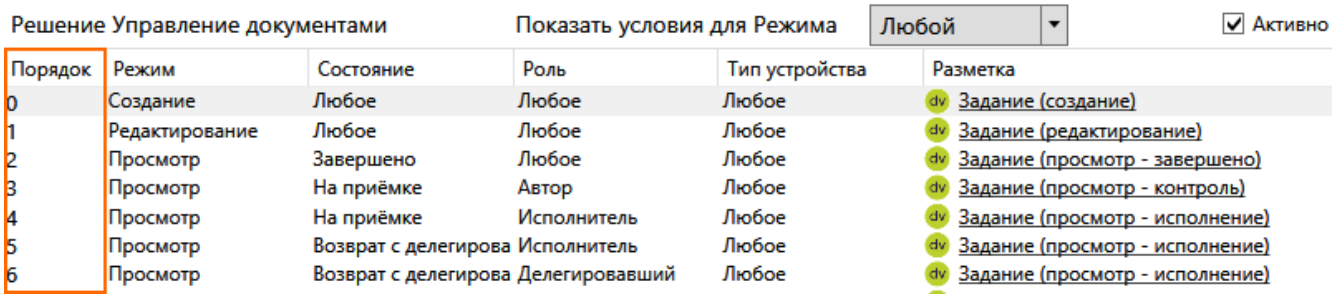


Рисунок 508. Изменение приоритета условия на вкладке "Условия выбора"

2. Выберите условие.
3. На ленте команд нажмите кнопку:
   - **Вверх**, чтобы повысить приоритет условия.
   - **Вниз**, чтобы понизить приоритет условия.



Если более общему условию будет назначен приоритет выше чем более жёсткому условию, более жёсткое условие не будет применяться ни при каких ситуациях. В таком случае условие с более жесткими критериями будет выделено красным цветом.

Назначить группу условий активной

Условия выбора разметок для определённой локации могут предоставляться несколькими Решениями. Условия выбора из разных Решений могут быть взаимоисключающими или излишними. Чтобы исключить возможную коллизию, Web-клиент работает только с условиями активного Решения.

Чтобы выбрать активное Решение:

1. Выберите локацию, для которой настраиваются условия выбора разметок.
2. Откройте вкладку Условия выбора.
3. Установите флаг **Активно** в группе условий, которые должны использоваться.

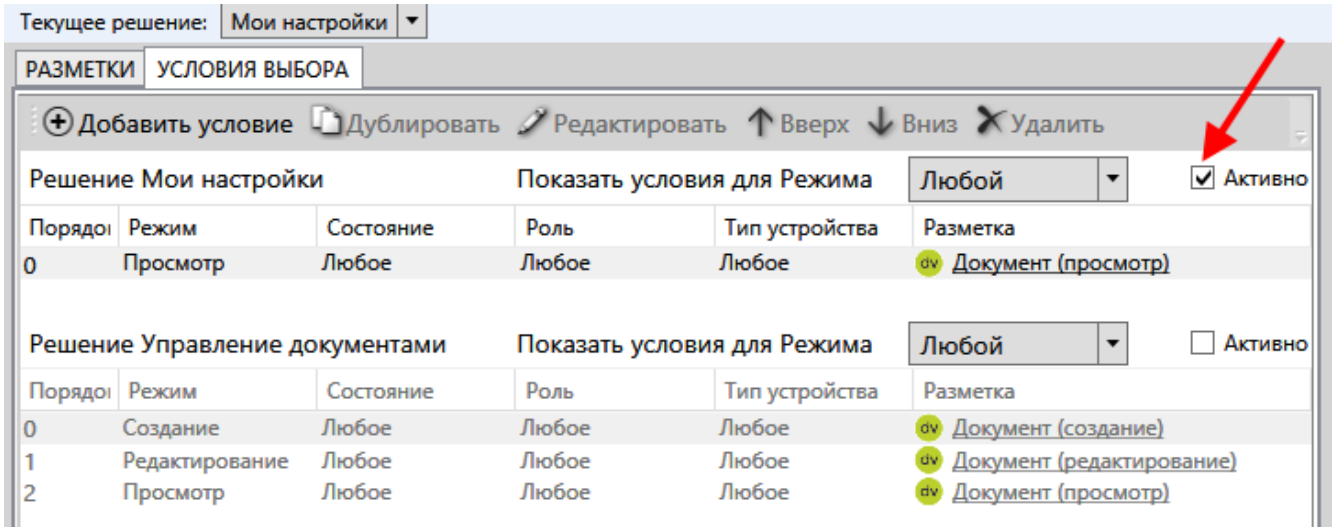


Рисунок 509. Флаг активизации условий Решения "Управления документами"

Управление локализованными ресурсами

Конструктор Web-разметок предоставляет механизм локализации элементов управления. Механизм использует локализованные ресурсы в свойствах элементов управления разметки.



Пример работы с локализованными ресурсами приведён в пункте Локализовать свойства элемента управления.

Управление локализованными ресурсами выполняется с помощью *Панели управления ресурсами локализации*.

Каждый локализованный ресурс закрепляется за определённым *Решением* с набором разметок, использующих данный ресурс. Локализованные ресурсы могут быть использованы только в разметках своего *Решения*.

При введение текста в локализуемое поле, текст вставляется в язык по умолчанию. Язык по умолчанию указывается в параметре `DefaultLanguage` в конфигурационном файле, расположенном в папке по адресу `Каталог-установки-Web-клиента\WebLayoutsDesigner.exe.config`.

Панель управления ресурсами локализации

Панель управления ресурсами локализации — инструмент управления локализованными ресурсами. При помощи панели управления можно создавать и изменять текстовые ресурсы, которые могут быть использованы в разметках для настройки элементов управления: текстовых меток, подсказок и т. п.

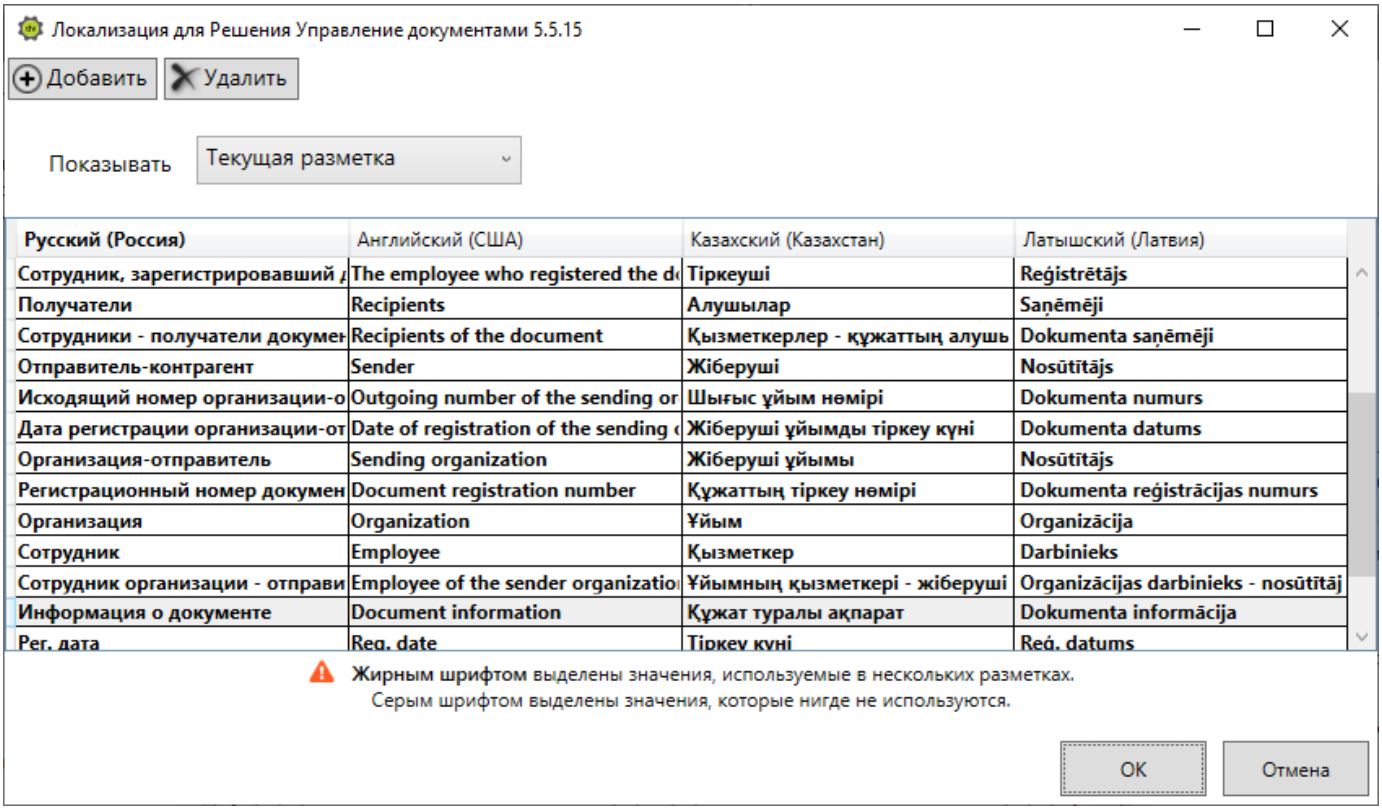


Рисунок 510. Панель управления ресурсами локализации

Панель управления ресурсами локализации может быть открыта двумя способами:

- из редактора разметок — для этого нажмите кнопку ... в настраиваемом свойстве элемента управления с поддержкой локализации.
- вызвана из меню программы Конструктор Web-разметок — команда Файл > Локализация.

  В данном случае в *Таблице локализаций* будут отображаться все ресурсы текущего выбранного Решения.

Если требуется настроить локализацию для языка, отсутствующего в *Панели управления ресурсами локализации*, добавьте его с помощью *Панели управления Web-клиента* и разрешите к использованию. Подробности см. пункт Добавление нового языка.

Компоненты панели управления

Таблица 28. Компоненты панели управления ресурсами локализации:

Добавить, **Удалить**	Добавляют и удаляют ресурсы.
Фильтр ресурсов	Фильтрует ресурсы по принадлежности к текущей разметке. Доступен, если *Панель управления ресурсами локализации* открыта из разметки. Фильтр содержит варианты: Все разметки. Отображаются все ресурсы Решения, в которое входит текущая разметка. Текущая разметка. Отображаются только ресурсы, использованные в текущей разметке.
Таблица локализаций	Таблица, содержащая значения ресурсов для поддерживаемых языков.
ОК, **Отмена**	Кнопки сохранения и отмены сохранения.



Шрифт строк в *Таблице локализаций* указывает на "используемость" локализованного ресурса:

- Стандартный шрифт — локализованный ресурс используется в одной разметке (возможно в нескольких элементах управления).
- Жирный шрифт — локализованный ресурс используется в нескольких разметках.
- Серый цвет шрифта — локализованный ресурс не используется.

Язык локализации по умолчанию

Язык локализации по умолчанию — выбранный в Панели управления Web-клиента язык. Язык локализации по умолчанию выполняет следующие функции:

- Определяет язык ресурсов, по которым выполняется быстрый поиск при присвоении значения свойствам элементов управления.

- Определяет второй обязательный язык (помимо английского), для которого должно быть указано значение при создании нового локализованного ресурса.

Текущий язык по умолчанию указывается в окне редактора разметок.

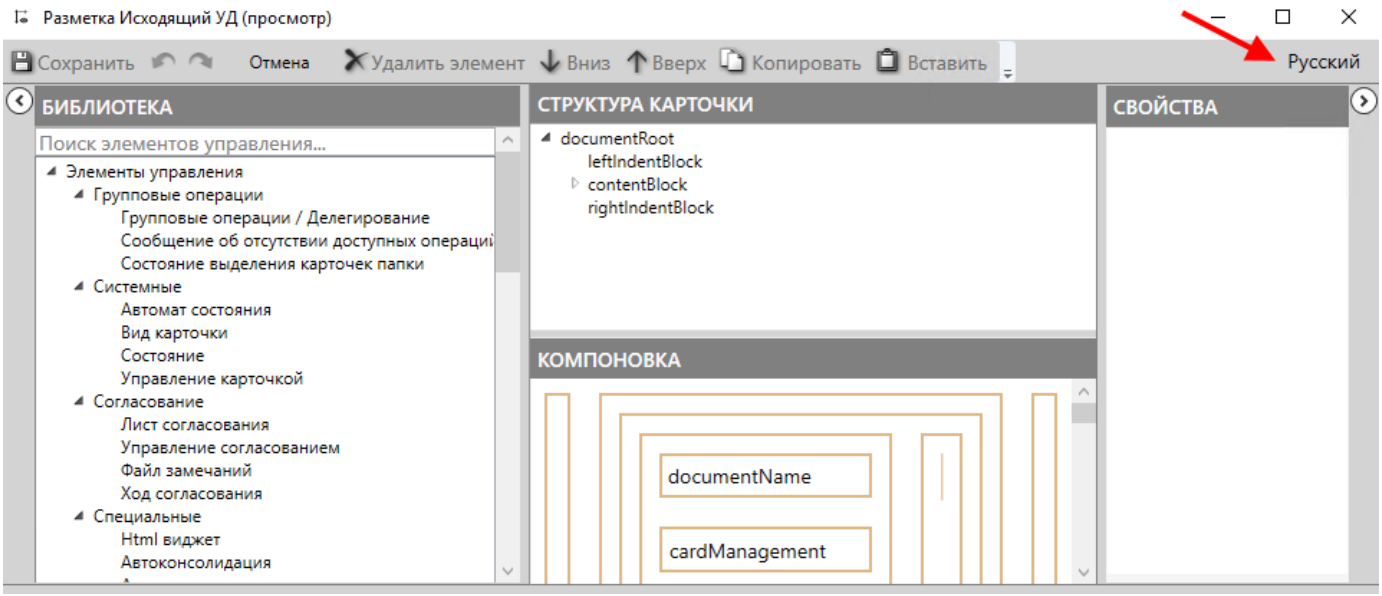


Рисунок 511. Язык локализации по умолчанию в окне редактора разметок

Язык локализации по умолчанию может быть изменён администратором Web-клиента через Панель управления Web-клиентом.

Создать локализованный ресурс

Новый локализованный ресурс создаётся автоматически, как только для настраиваемому свойству элемента управления присваивается новое значение в Редакторе разметок.



В случае возникновения конфликта значений локализованных ресурсов будет использоваться уже существующий ресурс. Новый ресурс для языка локализации по умолчанию не создаётся, если ресурс с аналогичным значением в списке ресурсов локализации текущего *Решения* уже имеется.

Локализованный ресурс может быть добавлен вручную.

Чтобы добавить локализованный ресурс вручную:

1. Откройте Панель управления ресурсами локализации.
2. Нажмите кнопку **Добавить**.

В *Таблицу локализаций* будет добавлена новая строка. Пустые значения для обязательного языка локализации (английского) и для языка локализации по умолчанию будут выделены красным цветом.

3. Перейдите в режим редактирования нажатием по выбранной ячейке в *Таблице локализации*, и введите значение локализованного ресурса для соответствующего языка локализации.



Обязательно должны быть заполнены значения в ячейках, выделенных красным цветом.

4. Нажмите **OK** для сохранения изменений.

Если *Панель управления ресурсами локализации* была открыта при настройке свойства элемента управления в *Редакторе разметки*, перед сохранением выберите локализованный ресурс, значение которого должно быть присвоено настраиваемому свойству.

Добавленный локализованный ресурс будет доступен для выбора при настройке свойств элементов управления Решения, в которое был добавлен ресурс.

Изменить локализованный ресурс

Чтобы изменить значение локализованного ресурса:

1. Откройте *Панель управления ресурсами локализации*.
2. В *Таблице локализаций* выберите изменяемый локализованный ресурс.
3. Активируйте режим редактирования одинарным нажатием по содержимому ячейки, соответствующей языку локализации.
4. Измените значение и нажмите **OK** для сохранения.

После изменения значения локализованного ресурса данное изменение будет отражено во всех использующих его разметках карточек.

Удалить локализованный ресурс

Чтобы удалить локализованный ресурс:

1. Откройте *Панель управления ресурсами локализации*.
2. В *Таблице локализаций* выберите удаляемый локализованный ресурс или

несколько ресурсов, удерживая клавишу `Shift` или `Ctrl`.

3. Нажмите кнопку **Удалить**.

   > Если локализованный ресурс (или ресурсы) используется в разметках, будет выведен дополнительный запрос на удаление.

4. Нажмите **ОК** для сохранения изменений.



Значения соответствующих свойств будут очищены после сохранения изменений, если локализованный ресурс используется в свойствах элементов управления.

Руководство по настройке Решений в программе Конструктор Web-разметок

Настройка Решения в программе Конструктор Web-разметок — процесс, включающий несколько последовательно выполняемых действий:

1. Создайте новое Решение.

   На данном этапе формируется проект Решения, в который далее будут включены разметки, условия выбора, локации и др.

2. Создайте новые локации.

   Локации — страницы Web-клиента или части страницы, к которым могут применяться разметки. Стандартные локации: карточки, дашборд, основное меню и др. При необходимости могут быть добавлены собственные локации, которые могут быть встроены в разметки других локаций или открываться как отдельные страницы по ссылке.

3. Создайте разметки.

   Данный этап может включать создание разметок для карточек, дерева папок, дашборда и других локаций.

   Для каждой локации может быть разработано несколько разметок. Например, можно создать отдельные разметки для каждого типа устройств, роли пользователя, состояния карточки и т.д. Количество разметок, которые потребуется разработать, определяется собственными требованиями. Как минимум может потребоваться разработать разметки для собственных

видов/типов карточек: для создания, изменения и просмотра карточки.



На данном шаге важно, чтобы при добавлении разметки текущим было выбрано решение, созданное в п. 1.

4. Настройте разметки.

   Настройка разметки — фактическая её реализация. Данный этап включает:

   - формирование разметки с помощью предоставляемых и собственных элементов управления,
   - настройка элементов управления разметки (связывание их с данными, настройка общих параметров),
   - локализация и др.

     Этап настройки разметок также может включать разработку клиентских скриптов (для реализации дополнительной бизнес-логики), стилей (для изменения графического интерфейса) и др.

5. Настройте условия выбора разметок.

   Условия выбора разметок — правила, по которым Web-клиент выбирает разметку, необходимую для отображения данных в определённой локации, из всех доступных разметок.

   Условия выбора разметок могут содержать различные критерии. Список критериев зависит от локации, для которой настраивается условие.



На данном шаге важно, чтобы при добавлении условия текущим было выбрано решение, созданное в п. 1.

6. Экспортируйте Решение.

   Настройки Решения (разметки, условия выбора и др.) выгружаются в файл XML, который может быть загружен в другие БД.

При настройке разметок Решений ориентируйтесь на правила, по которым Web-клиент выбирает разметки. Правила приведены в ПРИЛОЖЕНИИ Правила выбора разметки для открытия карточки.

С целью оптимизации Web-клиент кэширует разметки на сервере. Ознакомьтесь с особенностями настройки и использования кэша.

Настройка разметки карточек

1. Выберите локацию карточки, для которой настраиваются разметки.

   Чтобы выбрать локацию:

   a. Выберите группу локаций с требуемой локацией карточки:

   ArchiveManagement Карточки

   Группа локаций библиотеки модуля *Управление архивом*.

   Используйте только совместимые версии модулей Управление архивом и Web-клиент. Версия 6.2 и несовместима с версией 5.5.

   По вопросам совместимости Управление архивом версий 6.1 и 6.2 с Web-клиентом версии 6.1 и выше проконсультируйтесь со специалистом Docsvision.

   BackOffice Карточки

   Группа локаций библиотеки карточек *Базовые объекты*, включает локации карточек приложения *Управление документами* и др. приложений, основанных на карточках данной библиотеки карточек.

   TakeOffice Карточки

   Группа локаций библиотеки карточек *Делопроизводство 4.5*.

   В Web-клиенте отсутствуют элементы управления для работы с карточками Делопроизводство 4.5. Такие ЭУ могут быть реализованы самостоятельно.

   ApprovalDesigner Карточки

   Группа локаций библиотеки карточек *Конструктор согласований*.

   Workflow Карточки

   Группа локаций библиотеки карточек *Управление процессами*.

   b. Выберите локацию, разметки которой настраиваются.

Название локации соответствует названию типа/вида карточек. Например: *Документ / ДокументУД / Исходящий*.

2. Выберите Решение, в которое включаются разметки и условия выбора, в списке **Текущее решение**.

   Новые разметки автоматически включаются в Решение, выбранное в списке **Текущее решение**. Если распространять разметки в виде готового Решения не предполагается, можно выбрать Решение *Мои настройки*.

3. Создайте разметки карточек.

   В Web-клиенте есть два типа разметок: для просмотра и для редактирования карточек. Разметка редактирования также используется для создания карточек.

   В зависимости от требований, могут быть созданы разметки для просмотра, создания и редактирования карточек, или, если возможность создания и изменения карточки не требуется, только для просмотра карточек.

   Для каждой роли пользователя и/или состояния карточки могут быть разработаны собственные разметки. Также собственные разметки могут быть разработаны для каждого типа клиентского устройства: персональный компьютер, планшет, смартфон.

4. Настройте разметки.

   На данном этапе в новые разметки добавляются элементы управления для ввода/просмотра данных карточки. Список элементов управления определяется индивидуально и зависит от типа/вида карточки и бизнес-требований. Параллельно могут быть разработаны скрипты карточек, реализующие/переопределяющие бизнес-логику работы с ней.

   Некоторые ЭУ жестко связаны с данными карточки, другие должны содержать настройки *Источник данных*, *Поле данных*.

   В разметку редактирования необходимо добавить ЭУ Кнопки сохранения, иначе возможность сохранения изменений будет недоступна.

5. Добавьте условия выбора разметок.

   Условия выбора определяют, какая разметка должна быть использована для

открытия/создания карточки.

На данном этапе нужно добавить условия, которые будут выбирать созданные на шаге 3 разметки, в зависимости от режима работы с карточкой, роли пользователя, типа устройства и других доступных критериев.

Для полноценной работы с карточками (создание/просмотр/редактирование) необходимо добавить условия с режимами *Просмотр*, *Редактирование*, *Создание*, выбирающие соответствующие разметки.

Новые условия автоматически включаются в Решение, выбранное текущим.

6. Назначьте действующий набор условий.

   Решение, в которое были добавлены условия, должно быть назначено действующим, иначе соответствующие разметки выбираться не будут.

Настройка разметки справочников

Конструктор Web-разметок предоставляет возможность настраивать разметки для справочников и строк справочников.



Для доступа к строкам справочников Web-клиент предоставляет специальные методы, которые перечислены в пункте Список специальных адресов Web-клиента. Используя данные методы можно работать со строками справочников, когда это недоступно стандартными средствами.

Чтобы создать разметку:

1. Выберите настраиваемое Решение, в которое будет включена разметка.
2. Выберите локацию узла справочника, для строк которого настраиваются разметки.

   Чтобы выбрать локацию:

   a. Выберите группу локаций с требуемой локацией справочников:

   ArchiveManagement Справочники

   Содержит локации справочников модуля *Управление архивом*.



Используйте только совместимые версии модулей Управление архивом и Web-клиент. Версия 6.2 и несовместима с версией 5.5.

По вопросам совместимости Управление архивом версий 6.1 и 6.2 с Web-клиентом версии 6.1 и выше проконсультируйтесь со специалистом Docsvision.

BackOffice Справочники

Содержит локации справочников модуля *Базовые объекты*.

TakeOffice Справочники

Содержит локации справочников модуля *Делопроизводство 4.5*.



В Web-клиенте отсутствуют элементы управления для работы с карточками Делопроизводство 4.5. Такие ЭУ могут быть реализованы самостоятельно.

Platform Справочники

Содержит локации справочников модуля *Платформа*.

WebClient Справочники

Содержит локации справочников модуля *WebClient*.

Workflow Справочники

Содержит локации справочников модуля *Управление процессами*.

b. Выберите локацию, разметки которой вы хотите настроить.

Название локации соответствует названию справочника\узла\строки. Например: **Справочник контрагентов › Подразделения**.

3. Создайте разметки.

Для справочника достаточно создать разметку для просмотра.

Для строк справочников можно создать разметки для создания, просмотра и редактирования.

4. Настройте разметки.

Для разметок строк добавьте элементы, позволяющие отображать\изменять значения полей данной строки, а также кнопки сохранения.

Для разметки самого справочника можно добавить элементы для работы с деревом или списком значений: отображение, открытие.

5. Добавьте условия выбора разметок.

   В зависимости от типа справочника/строки будут доступны различные критерии для создания условий выбора:

   - для всех элементов доступна возможность создания условий на основе режима открытия элемента, роли пользователя и типа пользовательского устройства;
   - для Справочника сотрудников и Справочника контрагентов доступен дополнительный критерий: *Тип строки* — позволяет выбирать разметку в зависимости от типа редактируемой строки;
   - для строк Подразделения и Сотрудники Справочника сотрудников и Справочника контрагентов, а также строк Узлы справочника и Строка справочника Конструктора справочников доступен дополнительный критерий: *Вид карточки*. Критерий: *Вид карточки* позволяет выбирать разметку в зависимости от *Вида карточки*, который выбран для узла данной строки в справочнике в системе Docsvision.

Настройка разметки основного меню

Основное меню — сворачиваемая панель с деревом папок, размещенная в левой части окна Web-клиента.

Настройка разметки основного меню в программе Конструктор Web-разметок включает несколько последовательно выполняемых действий:

1. Выберите локацию основного меню: **WebFrame › Основное меню**.
2. Выберите Решение, в которое включаются разметки и условия выбора, в списке Текущее решение.
3. Создайте разметки основного меню.

   Для основного меню требуется разметка с типом *Просмотр*, режим *Редактирование* не используется. Может быть разработана одна разметка для всех типов пользовательских устройств или отдельные разметки для каждого устройства.

4. Настройка разметок.

   Типовая разметка основного меню может включать следующие элементы:

- Контейнер / Главное меню.
  - Контейнер / Панель элемента меню — содержит ЭУ, необходимые для настройки и обновления дерева папок пользователем.
    - Кнопка / Настроить.
    - Кнопка / Обновить.
    - Кнопка / Закрепить меню.
  - Ссылка / Страница Web-клиента — настроен на открытие страницы `#/Dashboard`.
  - Контейнер / Настраиваемая область — содержит папки, которые могут быть скрыты пользователем.
    - Контейнер / Группа элементов.
      - Папка / Известная папка.
      - Другие папки и группы папок:
    - Папки / Пользовательские папки.
- Скрипт / Настройка пользовательских папок — является контейнером для подключаемых пользователем папок.

  > При настройке разметки следует обратить внимание, что некоторые ЭУ предназначены для работы в связке с другими ЭУ, и в их настройках предусмотрено указывать названия связанных ЭУ. Например, при настройке ЭУ `Скрипт / Настройка пользовательских папок` должны быть указаны названия элементов управления "Контейнер / Настраиваемая область" и "Папки / Пользовательские папки".

5. Добавьте условия выбора разметок.

   Должно быть добавлено условие, использующее разметку, созданную на шаге 3, для открытия основного меню.

6. Назначьте действующий набор условий.

   Решение, в которое были добавлены условия, должно быть назначено действующим, иначе соответствующие разметки выбираться не будут.

Настройка разметки главной страницы

Главная страница — страница, которая открывается при запуске Web-клиента и при переходе по ссылке `#/Dashboard`.

Настройка разметки главной страницы в программе Конструктор Web-разметок включает несколько последовательно выполняемых действий:

1. Выберите локацию главной страницы: **WebFrame › Главная страница**.
2. Выберите Решение, в которое включаются разметки и условия выбора, в списке `Текущее решение`.
3. Создайте разметку главной страницы.

   Для главной страницы требуется разметка с типом *Просмотр*, режим *Редактирование* не используется. Может быть разработана одна разметка для всех типов пользовательских устройств или отдельные разметки для каждого устройства.

   Рекомендуемый вариант: разработать разметку для устройства с маленьким экраном и разметку для других устройств.

4. Настройте разметки.

   Типовая разметка основного меню может включать следующие элементы:

   - Виджет / Группа папок.
     - Виджет / Известная папка.
     - Другие папки и группы папок:
   - Виджет / Последние карточки — содержит список последних карточек.

5. Добавьте условия выбора разметок.

   Должно быть добавлено условие, использующее разметку, созданную на шаге 3, для открытия главной страницы.

6. Назначьте действующий набор условий.

   Решение, в которое были добавлены условия, должно быть назначено действующим, иначе соответствующие разметки выбираться не будут.

Настройка разметки основного окна Web-клиента

Пользователь программы Конструктор Web-разметок может настроить разметку основного окна Web-клиента, определяющую расположение основного меню, панели меню, области представления и пр.



Количество и порядок элементов в разметке основного окна не ограничены. В зависимости от размещения элементов на разметке, вариантов взаимодействия элементов и других факторов возможно возникновение программных ошибок или ошибок, связанных с отображением элементов управления.

Проверьте работу разметок в тестовом окружении перед их применением в рабочем окружении.

Чтобы настроить разметки основного окна Web-клиента:

1. В программе Конструктор Web-разметок перейдите в локацию **WebFrame** › **Корень WebFrame**.
2. Создайте копию стандартных разметок в **собственном решении**:
   - *WebFrame* — разметка для полной версии Web-клиента.
   - *WebFrame_Smartphone* — разметка для мобильной версии Web-клиента.
3. Измените разметки созданной копии по необходимости.

   Размещение элементов, присутствующих по умолчанию в стандартной разметке может быть изменено. Также в разметку могут быть добавлены любые из доступных элементов Библиотеки элементов управления с небольшими ограничениями.

   Ограничения на изменение разметки WebFrame:

   Следующие элементы могут быть добавлены только в Панель навигации:

   - Панель навигации — кнопка "Назад".
   - Панель навигации — кнопка "Создать".
   - Поиск по штрих-коду.

   Следующие элементы могут быть добавлены только в Контейнер заголовка:

   - Кнопка главного меню.
   - Логотип компании.

- Панель поиска.
- Панель поиска по справочнику.
- Панель пользователя.
- Кнопка "Помощь".

Следующие элементы могут быть добавлены в Контейнер заголовка и Панель навигации:

- Кнопка.
- Строка.
- Число.
- Url.
- Раскрывающийся список.
- Метка.
- Группа радиокнопок.

В разметке WebFrame с элементами управления доступны следующие действия:

- Для ЭУ Контейнер заголовка и Панель навигации доступны только вставка и копирование.
- Для содержимого ЭУ Контейнер заголовка и Панель навигации доступны все действия: можно менять порядок, удалять элементы.
- Для всех остальных ЭУ доступно только копирование.

4. Добавьте условие выбора, по которому будут выбираться созданные разметки.

Настройка разметки диалога профиля пользователя

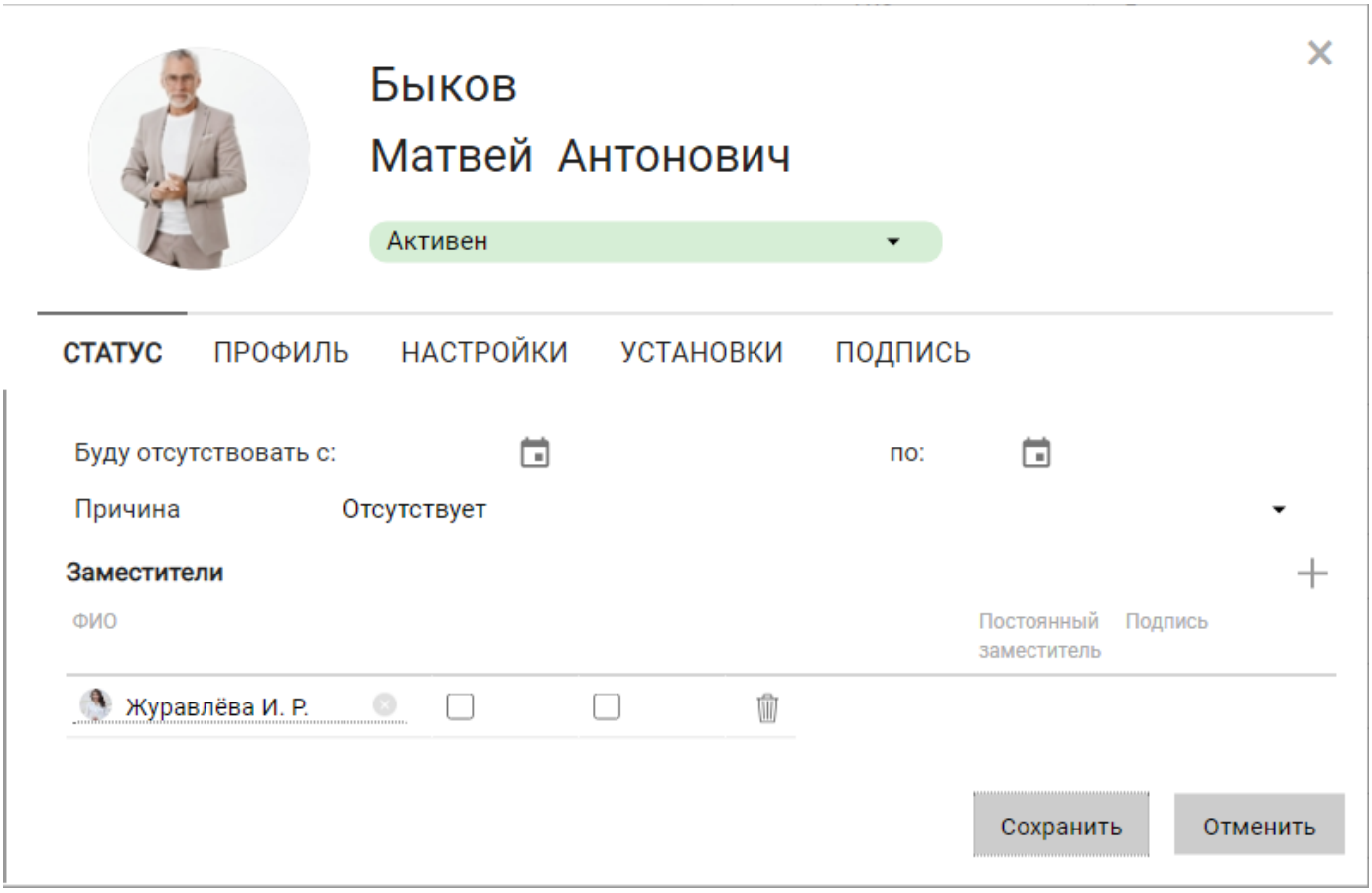


Рисунок 512. Профиль пользователя в полной версии Web-клиента

1. Откройте в программе Конструктор Web-разметок разметку:
   - **WebFrame › Диалог профиля пользователя › DV.UserProfile** — для полной версии Web-клиента;
   - **WebFrame › Диалог профиля пользователя › DV.UserProfile_mobile** — для мобильной версии Web-клиента.
2. Настройте разметку требуемым образом и сохраните изменения.

Изменение изображения по умолчанию для фотографии сотрудника

Фотография сотрудника отображается в элементе Изображение. Если у сотрудника отсутствует фотография в Справочнике сотрудников, будет подставлено изображение по умолчанию.

Функция получения изображения по умолчанию указывается в свойствах элемента Изображение в событии *Получение изображения по умолчанию*. Для Диалога профиля пользователя указана функция ***getDefaultImage***, которая вставляет изображение `/lib/docsvision/webclient/Content/App/Assets/Images/default-user-pic-male.svg` или `default-user-pic-female.svg` в зависимости от пола сотрудника.

Если требуется, изображение по умолчанию можно заменить:

1. Добавьте в папку своего решения в Web-клиенте изображения по умолчанию в формате PNG, SVG или JPEG.
2. Разработайте скрипт с функцией, которая будет устанавливать изображение по умолчанию в элемент управления `Изображение`. Пример такой функции смотрите в стандартной реализации: файл `/lib/docsvision/webclient/Content/App/BackOffice/Solution/UserProfile.ts`, функция ***getDefaultImage***.
3. Разработанный скрипт разместите в каталоге скриптов решения.
4. Создайте новую разметку для диалога профиля пользователя в собственном решении.

> Если изменить стандартную разметку или стандартные изображения, изменения будут перезаписаны при следующем обновлении Web-клиента.

5. В обработчике события *Получение изображения по умолчанию* укажите название разработанной функции установки изображения по умолчанию.
6. Сохраните изменения. Перезапустите сервер Web-клиента.

Настройка панели пользователя

По умолчанию на панели меню отображаются ФИО сотрудника, его фото и состояние учётной записи (и псевдоним текущей БД если пользователь является администратором).



Рисунок 513. Информация о пользователе на панели меню

Чтобы изменить настройки панели пользователя:

1. В программе Конструктор Web-разметок скопируйте в собственное Решение разметку **WebFrame › Корень WebFrame › WebFrame**.
   - **WebFrame › Корень WebFrame › WebFrame** — для полной версии Web-клиента;
   - **WebFrame › Корень WebFrame › WebFrame_Smartphone** — для мобильной версии Web-клиента.

2. Откройте созданную копию разметки и перейдите к настройкам элемента управления `webFrameUserPanel`.

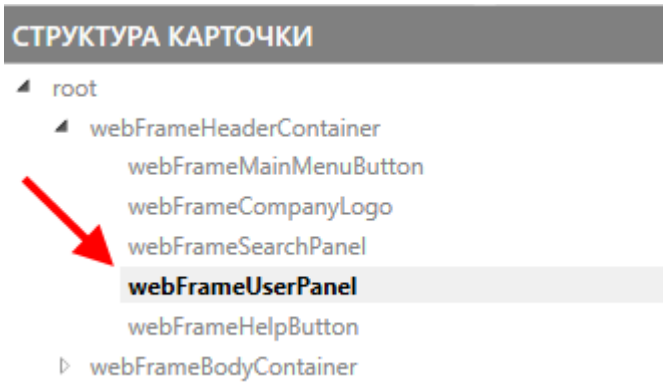


Рисунок 514. Элемент управления "webFrameUserPanel" в схеме разметки



Количество и порядок элементов в разметке основного окна не ограничены. В зависимости от размещения элементов на разметке, вариантов взаимодействия элементов и других факторов возможно возникновение программных ошибок или ошибок, связанных с отображением элементов управления.

Проверьте работу разметок в тестовом окружении перед их применением в рабочем окружении.

3. Измените настройки элемента управления. Описание параметров элемента управления приведено в пункте Панель пользователя.
4. Сохраните разметку.
5. Добавьте условие выбора, по которому будет выбираться добавленная разметка.

Настройка режима открытия карточки при поиске по штрих-коду

По умолчанию найденная по штрих-коду карточка будет открыта в новой вкладке веб-браузера.

Чтобы изменить поведение по умолчанию:

1. В программе Конструктор Web-разметок скопируйте в собственное Решение разметку **WebFrame › Корень WebFrame › WebFrame**:
   - **WebFrame › Корень WebFrame › WebFrame** — для полной версии Web-клиента;
   - **WebFrame › Корень WebFrame › WebFrame_Smartphone** — для

мобильной версии Web-клиента.

2. Откройте созданную копию разметки и перейдите к настройкам элемента управления `barcodeSearchButton`.

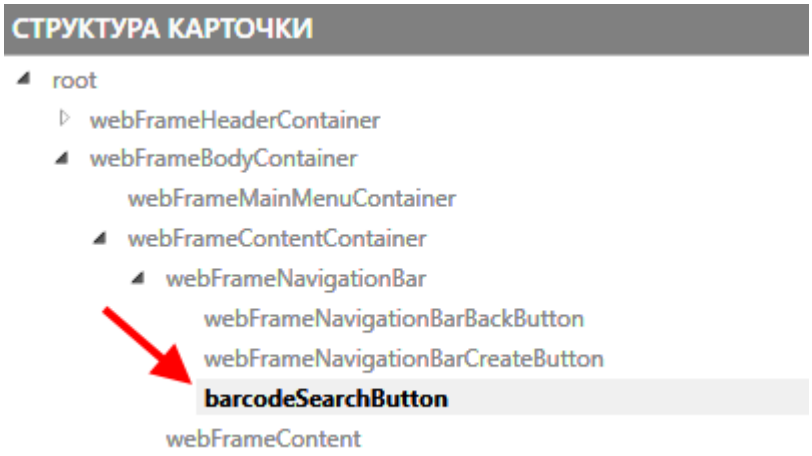


Рисунок 515. Элемент управления "barcodeSearchButton" в схеме разметки

3. Измените значение параметра ***Режим отображения***. Описание параметров элемента управления приведено в пункте Поиск по штрих-коду.
4. Сохраните разметку.
5. Добавьте условие выбора, по которому будет выбираться добавленная разметка.

Встраивание одной карточки в другую

Web-клиент предоставляет возможность отображать в карточке данные связанной карточки или карточки, получаемой с помощью расширенного источника данных. При этом может использоваться стандартная разметка отображаемой карточки или специальная разметка для отображения одной карточки в другой.

Чтобы настроить разметку в разметке:

1. Если требуется, создайте и настройте для вида встраиваемой карточки отдельную разметку, которая будет использоваться при её отображении в другой карточке. Для данной разметки добавьте условие выбора *Внутри другой разметки*.



При необходимости для встраивания можно использовать существующую разметку с любым условием выбора.

2. Настройте разметку карточки, в которую встраивается другая карточки.

   Встраиваемая карточка может являться связанной карточкой или быть карточкой, доступной через расширенные источники данных.

3. Создайте разметку, в которой будет отображать другая карточка.
4. Если требуется, подключите расширенные источники данных.
5. Добавьте в разметку элемент управления Контейнер локации — используется для отображения другой карточки.
6. Настройте элемент Контейнер локаций:
   - *Источник локации* — выберите значение ***Связанная карточка/строка***.
   - Выберите источник данных в параметрах: **Расширенный источник данных**, **Источник данных** и **Поле данных**.
   - *Локация* — выберите локацию с разметкой встраиваемой карточки (добавлена в п. 1).
   - *Режим* — выберите режим разметки, которая будет выбираться при открытии встраиваемой карточки.

     Если для встраиваемой карточки была разработана разметка с условием выбора *Внутри другой разметки*, в настройке *Режим* выберите ***Внутри другой разметки***.

   - *Максимальная высота* — укажите максимальный размер в пикселях для блока, в котором будет отображаться встраиваемая карточка.
   - Если требуется, настройте другие параметры.
7. Добавьте другие элементы в разметку.
8. Сохраните разметку.

Использование стилей Web-клиент в собственных разметках

В Web-клиент для разметок карточек модуля *Управление документами* были разработаны новые стили, которые по умолчанию используются только в стандартных карточках разметок. Для использования новых стилей в собственных разметках необходимо назначить специальные обработчики и стили на элементы разметок.

1. Назначьте обработчик на событие **Перед открытием карточки** корневого элемента

разметки documentRoot:

addNewStyleForControls

добавляет для разметки и её модальных окон класс ***new-style-layout***. Новые стили могут быть применены только при наличии у разметки данного класса.

2. Назначьте обработчики на событие **После открытия карточки** корневого элемента разметки *documentRoot*:

addStyleToCard

Включает в себя три обработчика:

getControlsForExpanding

Осуществляет поиск Блоков с классом ***block-with-border***, для которых включает поддержку возможности сворачивания/разворачивания по щелчку на общей кнопке сворачивания/разворачивания всех блоков.

addGrayBackgroundToCard

Добавляет в карточку фон серого цвета.

addShadowBorder

Осуществляет поиск раскрытых Блоков с классом ***block-with-border***, для которых добавляет серую рамку.

Вместо одного обработчика ***addStyleToCard*** можно указать обработчики ***getControlsForExpanding***, ***addGrayBackgroundToCard*** и ***addShadowBorder***. Если отдельные функции не требуются, обработчики можно не указывать.

outgoingDocument_CardManagementIsVisible

Загружает почту и телефон контрагентов, указанных в списке Адресаты, при загрузке карточки, а также скрывает пустой блок с примечанием.

Используется только в разметке *Исходящего* документа.

3. Назначьте обработчик на события **При сворачивании** и **При разворачивании** элементов Блок, Список файлов, *Ссылки* и др. сворачиваемых элементов:
   - ***switchBorderWithShadow*** — в развернутом состоянии добавляет к элементу рамку с тенью, и скрывает её при сворачивании элемента.

Обработчик ***switchBorderWithShadow*** следует добавлять только к элементам, для которых нужна тень. Например, это блок с информацией о карточке, список файлов или блок, в котором отображается ход исполнения.

Постоянную рамку для несворачиваемых блоков можно добавить с помощью стиля **block-with-shadow**: указать в параметре **Дополнительные CSS классы** требуемого элемента.



Для устаревших элементов управления `TaskCardFilePanel`, `TaskCardReportPanel` и т. п., у которых отсутствуют события **При сворачивании** / **При разворачивании**, рамка с тенью может быть добавлена только с помощью стиля **block-with-shadow**.

4. Укажите стили в настройке *Дополнительные CSS классы*:
   - Для корневого элемента `documentRoot`:

     system-layout-independent-scroll

     Включает отдельную прокрутку для элемента `Предпросмотр файла` и области с данными карточки.

     expansion-panel-header

     Включает новый дизайн для заголовков Блоков.

     center-screen

     Включает динамическое изменение боковых отступов от края страницы до элементов с данными карточки: чем уже окно браузера, тем меньше отступы.

     control-label-opacity-half

     Включает использование серого цвета для меток элементов управления.

   - Для Блоков:

     block-with-toggle-button

     Добавляет в Блок кнопку сворачивания/разворачивания всех Блоков со стилем **block-with-border**. Для избежания ошибок: стиль следует использовать только в одном Блоке разметки; у данного блока не должно быть обработчиков события *При щелчке*.

top-saving-buttons-block

Включает выравнивание содержимого по высоте, убирает лишнее пространство у кнопок.



Следует устанавливать только в Блоке, в котором размещается название карточки и кнопки сохранения, если дизайн пользовательской разметки повторяет дизайн стандартной разметки документов.

block-with-shadow

Добавляет тень и рамку к Блоку.



Если для блока включён обработчик ***switchBorderWithShadow***, добавлять стиль **block-with-shadow** не нужно.

Для Блоков, Списка файлов, Ссылок и др. сворачиваемых элементов, для которых назначен обработчик ***switchBorderWithShadow*** **block-with-border** добавляет функцию сворачивания/разворачивания элемента, вызываемую кнопкой сворачивания/разворачивания всех Блоков.

- Прочие доступные стандартные стили:

document-name-small

Устанавливает размер шрифта 1.4rem. В разметках Web-клиент используется в элементе с названием карточки;

document-name

Устанавливает размер шрифта 1.6875rem. Стандартные размер для элемента с названием карточки в Web-клиент 14.

card-kind-large

При ширине экрана более 420px устанавливает размер шрифта 1.125rem. В разметках Web-клиент используется в элементе Вид карточки.

document-view-state-wide

Переводит строчные буквы в прописные и добавляет межбуквенный

интервал. В разметках Web-клиент в элементе Состояние.

taskstable-with-hidden-filters

Скрывает фильтры в элементе Таблица исполнения.

colored-button

Для элемента Кнопка включает использование стандартной цветовой гаммы открытой карточки.

block-border-top, block-border-right, block-border-bottom, block-border-left

Добавляет серую линию в 1px по одной из границ элемента соответственно: верхней, правой, нижней, левой.

block-with-disclosure-body-center

Для Блоков включает выравнивание содержимого по центру.

block-with-disclosure-resize

Для Блока включает особый режим выравнивания дочерних блоков, если у Блока два дочерних блока шириной 50% от родительского и установленной минимальной шириной. Если ширины родительского блока достаточно для отображения обоих блоков в строку — они будут отображаться в строку, иначе — в столбец.

block-with-one-height

Для Блока включает особый режим выравнивания дочерних Блоков по высоте. Всем дочерним Блокам будет установлена высота самого высокого Блока.

block-with-state-buttons

Для Блока устанавливает особый режим выстраивания элементов Кнопка. При ширине окна меньше 421px кнопки будут размещаться вертикально с выравниванием по ширине родительского Блока.

Добавление ссылки на справочник в Основное меню

Для удобства работы со Справочниками Web-клиента рекомендуется добавить ссылки на справочники в Основное меню Web-клиента.



Если ссылки уже добавлены в основное меню, достаточно настроить условия выбора разметки.



Если нужно добавить ссылку на справочник в другую разметку, для разметки необходим режим управления решением *Поставщик*.

Чтобы добавить ссылку на справочник в Основное меню:

1. В окне программы Конструктор Web-разметок выберите разметку *WebFrame*, затем выберите *Основное меню*.
2. В текущую активную разметку добавьте новый элемент управления `Link/Web-client page`:

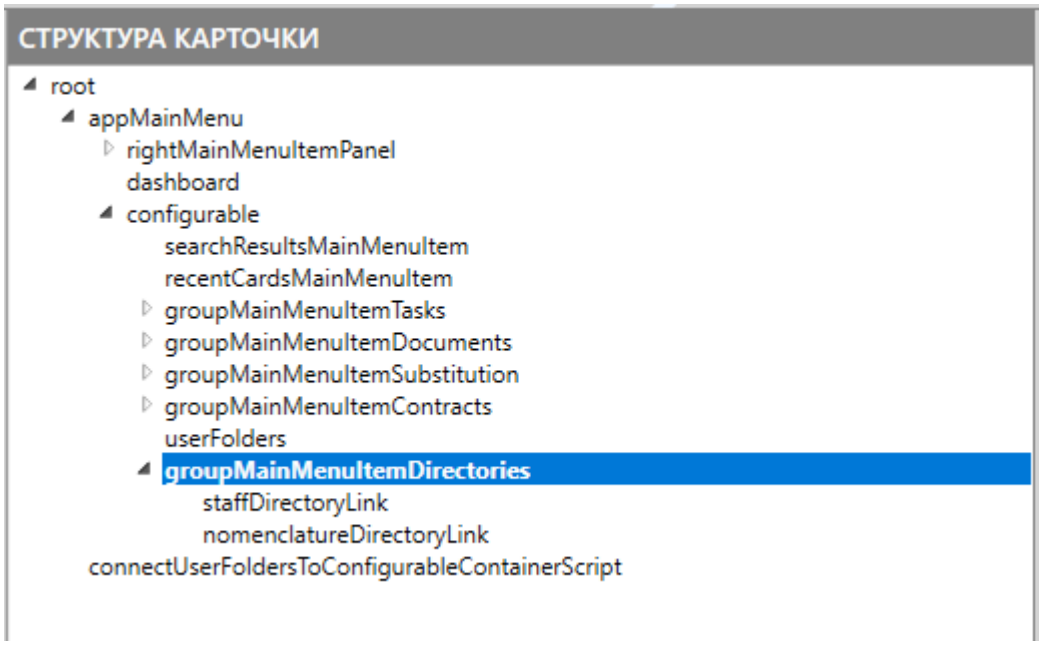


Рисунок 516. Добавление элемента управления "Ссылка"

3. Выберите добавленный элемент управления, заполните свойства, указав отображаемый текст ссылки в поле `Текст`:

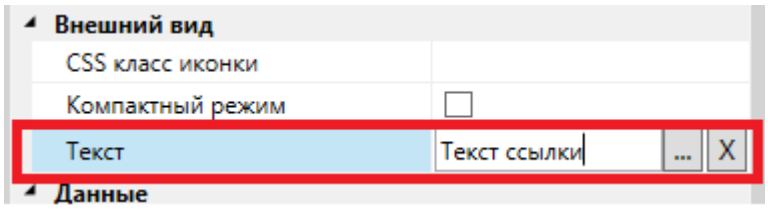


Рисунок 517. Заполненное поле "Текст"

4. В блоке свойств *Данные* укажите Адрес справочника:
   - Для справочника сотрудников: `/DirectoryView/6710b92a-e148-4363-8a6f-1aa0eb18936c`.

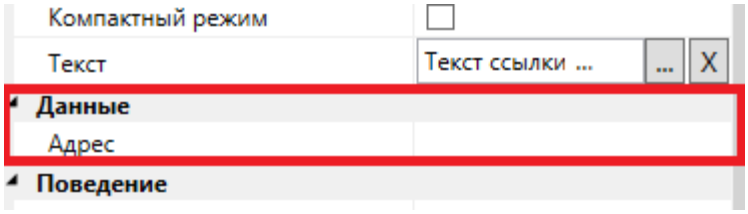


Рисунок 518. Свойство "Адрес"

5. Закройте Конструктор Web-разметок с сохранением изменений.

   В основном меню в разделе *Справочники* отобразится ссылка для перехода в справочник.

Добавление элемента управления "Дело" в карточку

Установка компонентов модуля *Управление архивом* открывает возможность работы со справочником номенклатуры дел. Перед работой со справочником необходимо выполнить настройку программы Конструктор Web-разметок и карточек вида *Документ*.

Для модуля Управление архивом 6.2 и выше используется ЭУ Контрол табличных данных, см. документацию модуля Управление архивом 6.2: "Контрол табличных данных".

Добавление полей в форму строки справочника

По умолчанию в форму добавления строки справочника выводятся только поля по умолчанию. При необходимости в форму можно выводить дополнительные поля.

Такое поведение поддерживается в Windows-клиенте. Если строка создаётся в конструкторе справочников в нужном справочнике, поля отображаются. В Web-клиенте необходимо выполнить дополнительные настройки, чтобы поля также выводились для заполнения.

1. В *Справочнике видов карточек* для типа карточки *Карточка строки справочника* добавьте новый вид, например, *Контрагенты*.

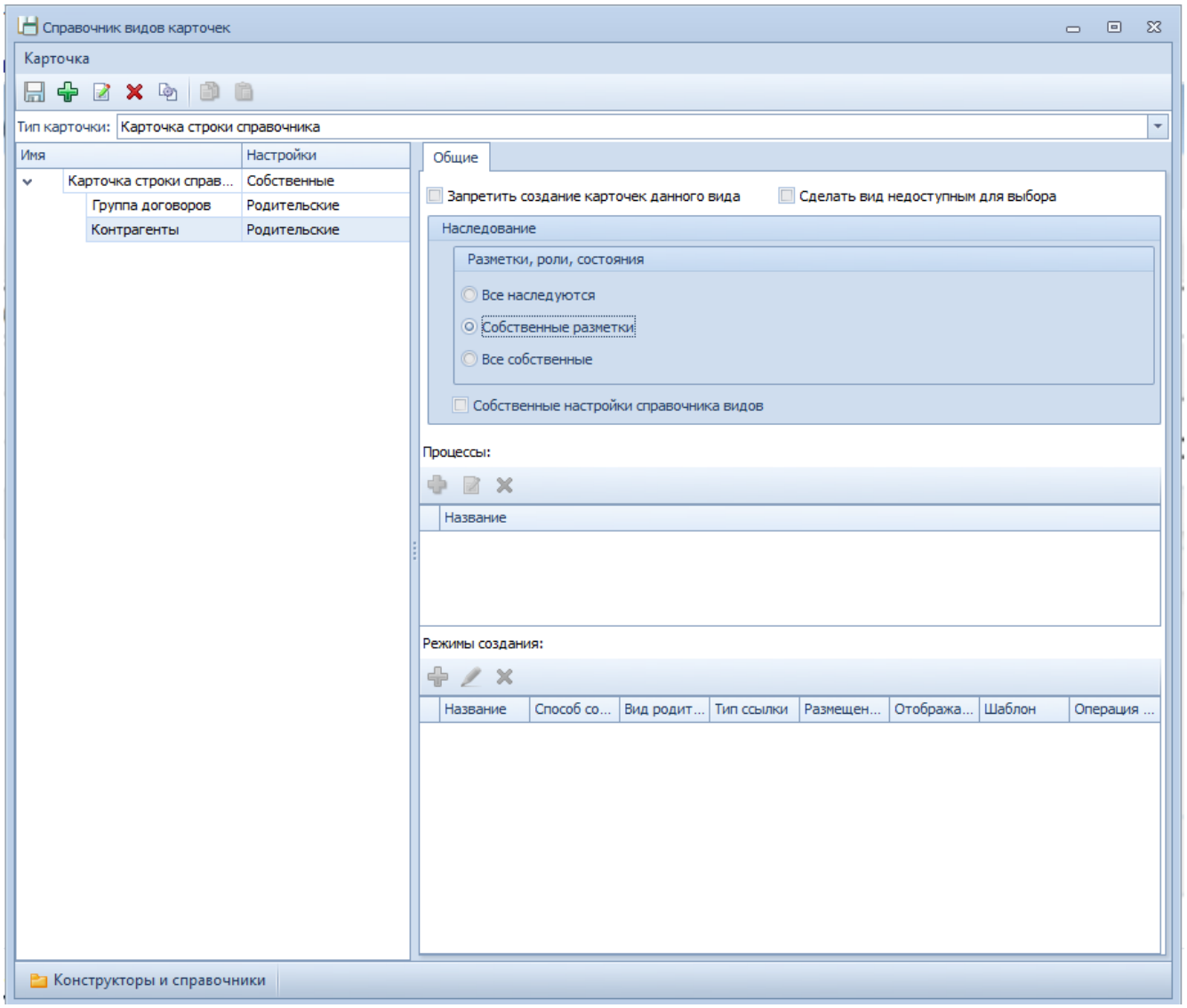


Рисунок 519. Добавление нового вида карточек

2. В настройках *Конструктора справочников* укажите созданный вид.

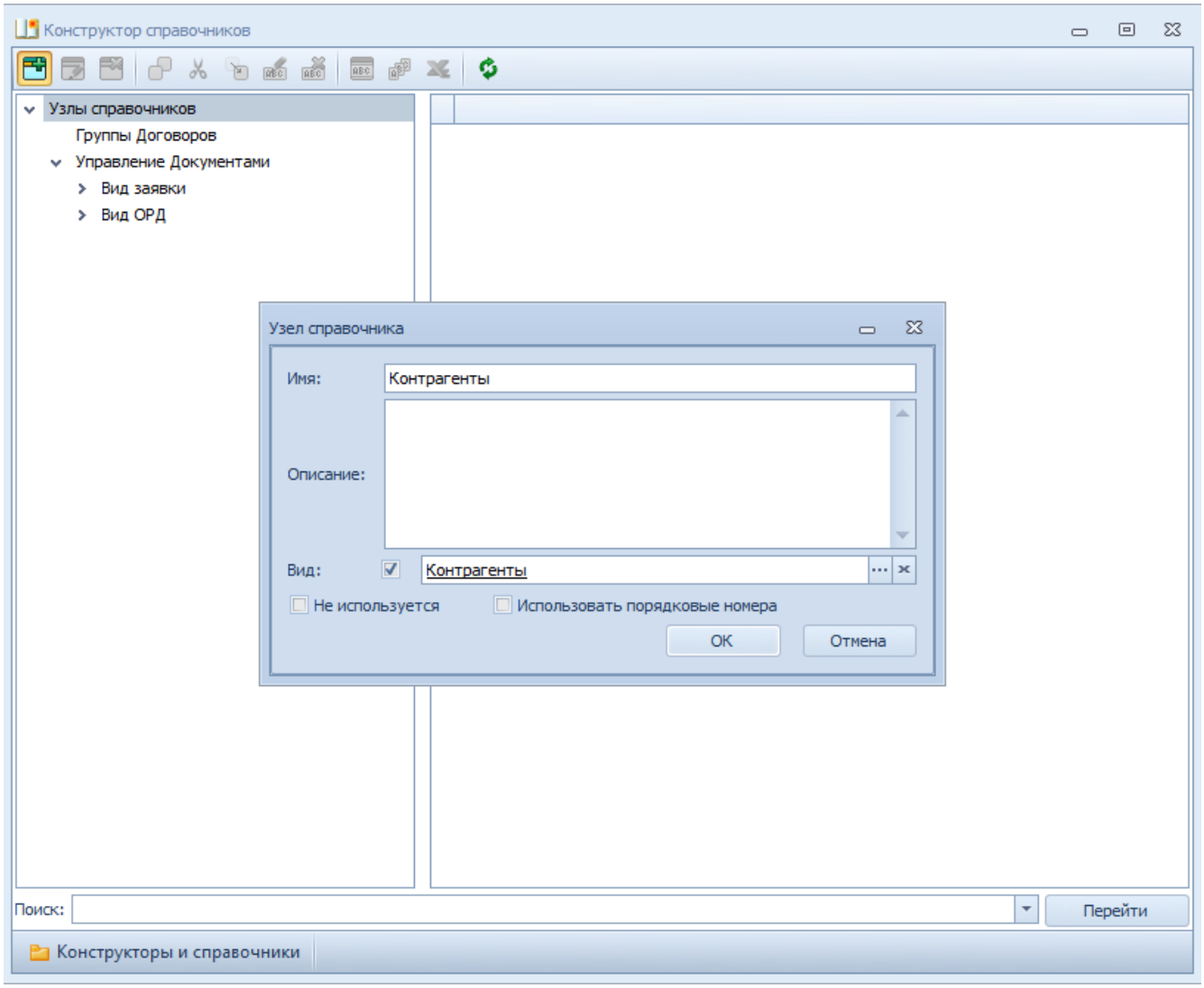


Рисунок 520. Настройки в конструкторе справочников

3. В программе Конструктор Web-разметок добавьте и настройте разметки создания, просмотра и редактирования для строки с контрагентами. Для упрощения задачи можно скопировать базовые разметки и внести изменения в их копии.

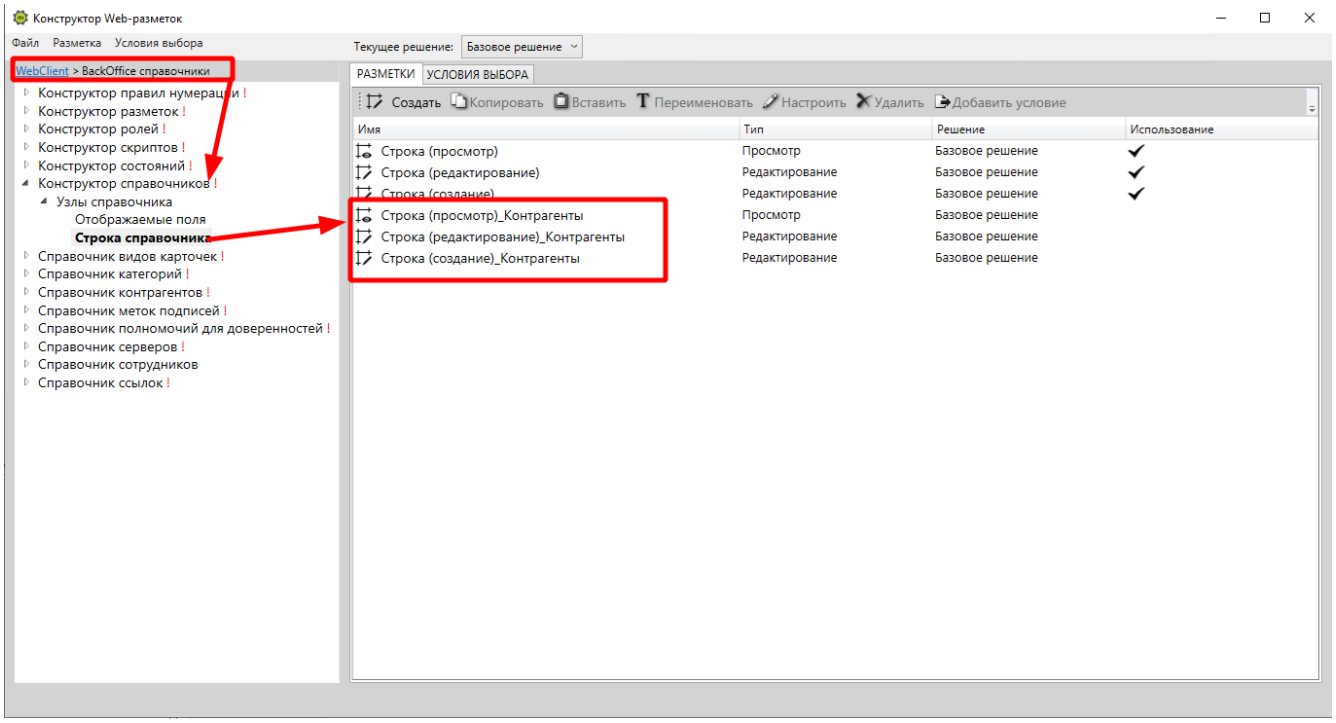


Рисунок 521. Настройка разметок для строки с контрагентами

4. На вкладке *Условия выбора* добавьте условия для созданных разметок. При этом укажите вид карточки, добавленный на первом шаге.

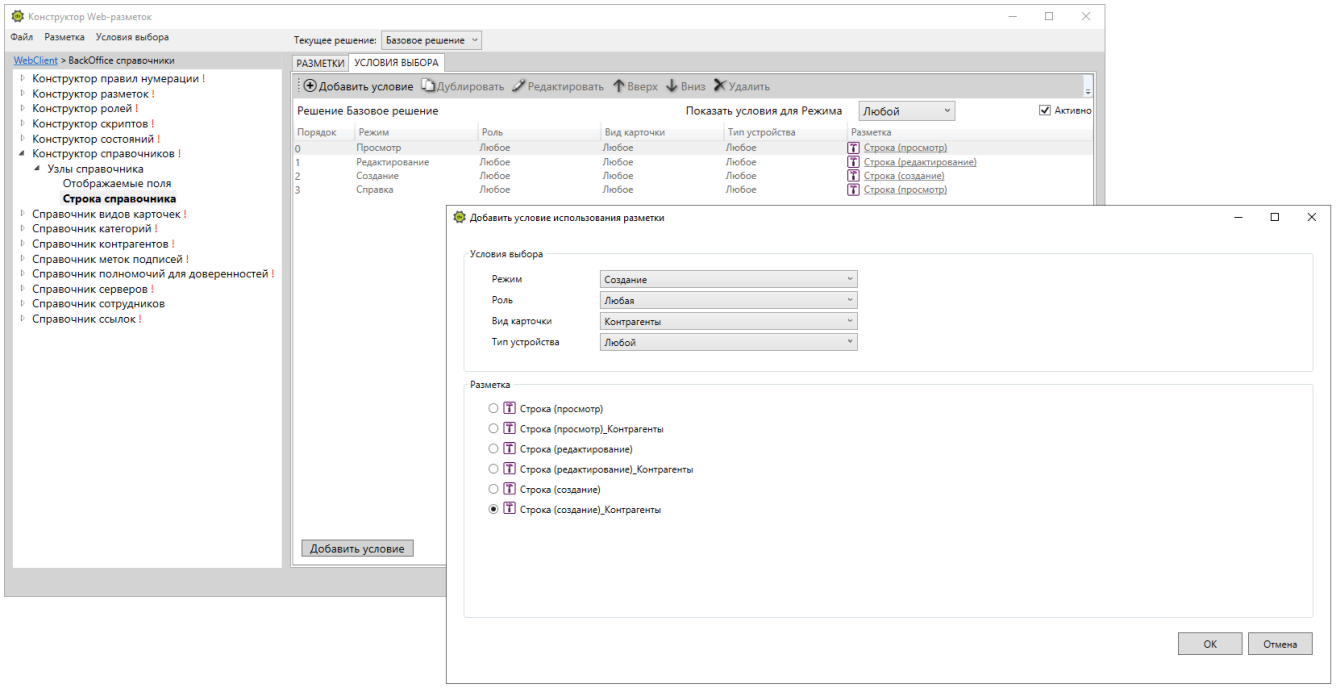


Рисунок 522. Настройка условий выбора

5. Настройте разметку в соответствии с предпочтениями.

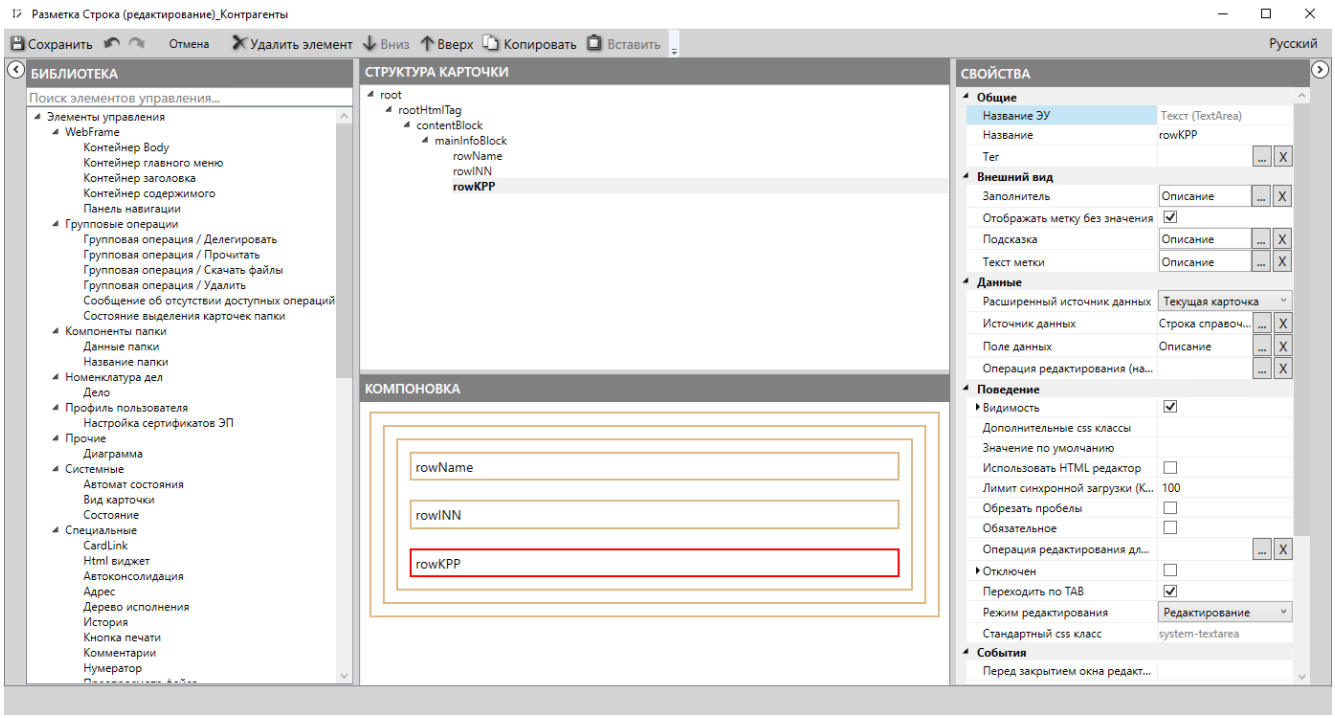


Рисунок 523. Настройка разметки карточки

6. В Конструкторе разметок Windows-клиента добавьте секцию метаданных. Для редактирования потребуется заблокировать разметку и метаданные карточки.

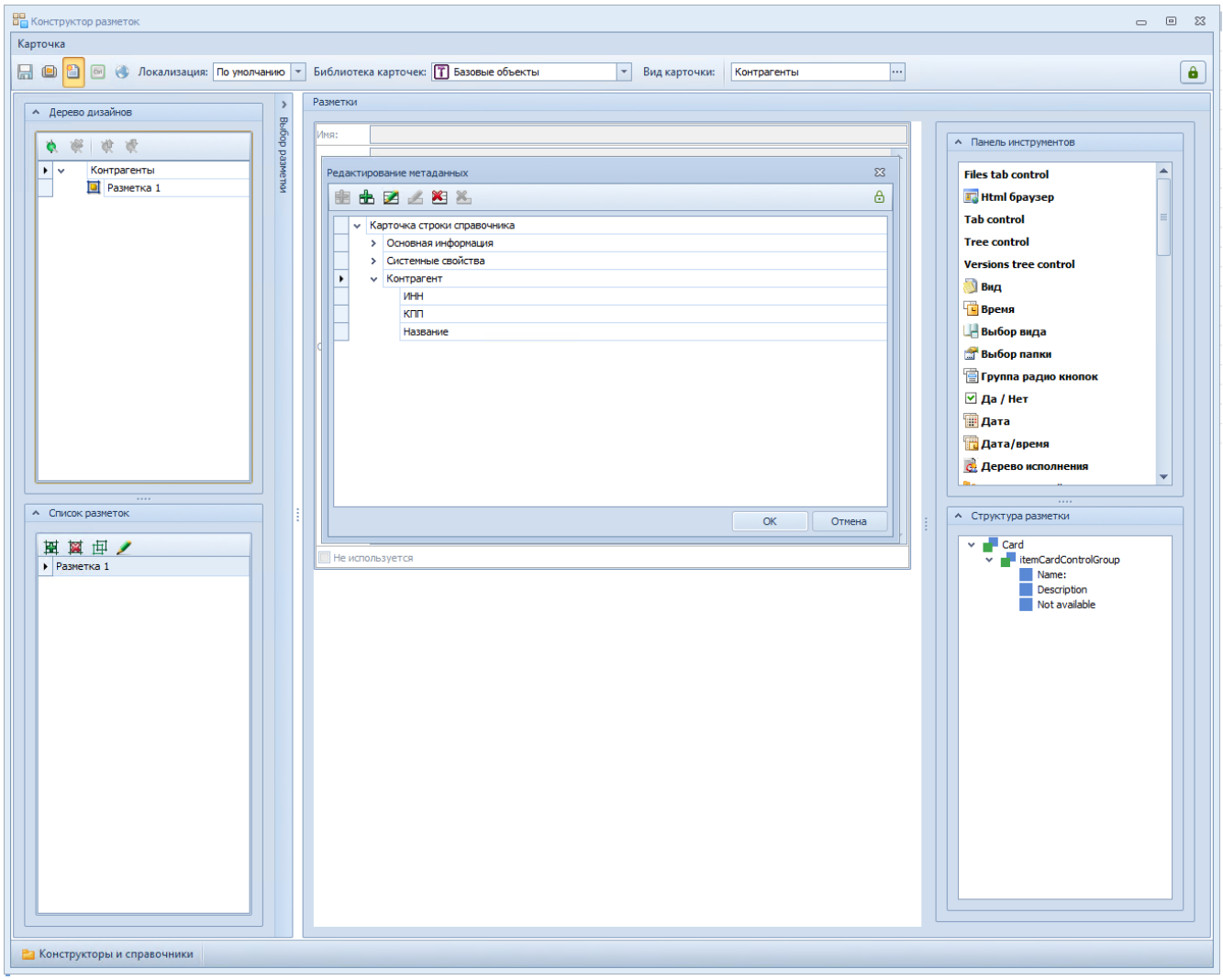


Рисунок 524. Редактирование метаданных карточки

7. Откройте Конструктор Web-разметок и добавьте расширенный источник данных для элемента `root` разметки:

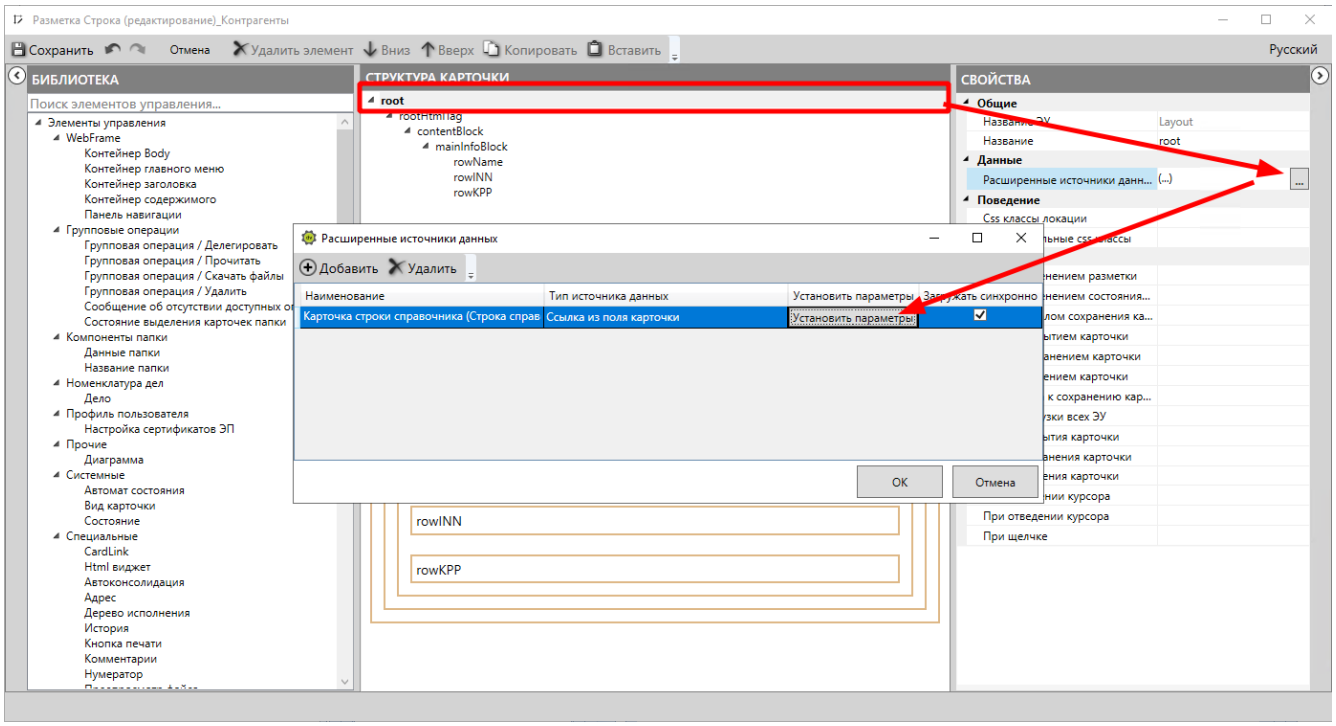


Рисунок 525. Редактирование расширенного источника данных

Выполните действие для разметок *Просмотра*, *Редактирования* и *Создания* карточки.

8. Укажите поле данных из расширенного источника для полей карточки:

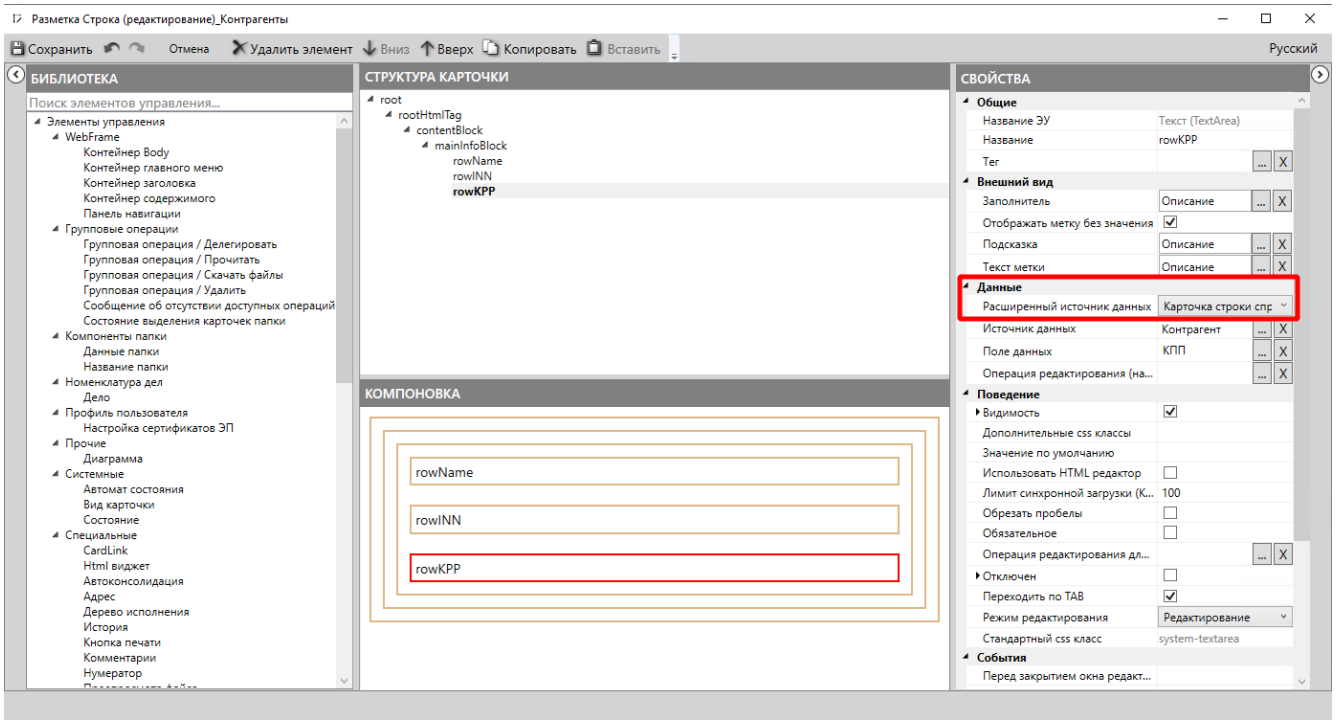


Рисунок 526. Выбор расширенного источника данных для поля

Выполните действие для каждого поля в разметках *Просмотра*,

Редактирования и *Создания* карточки.

9. Поднимите порядок добавленных условий в списке, чтобы они имели приоритет над стандартными.

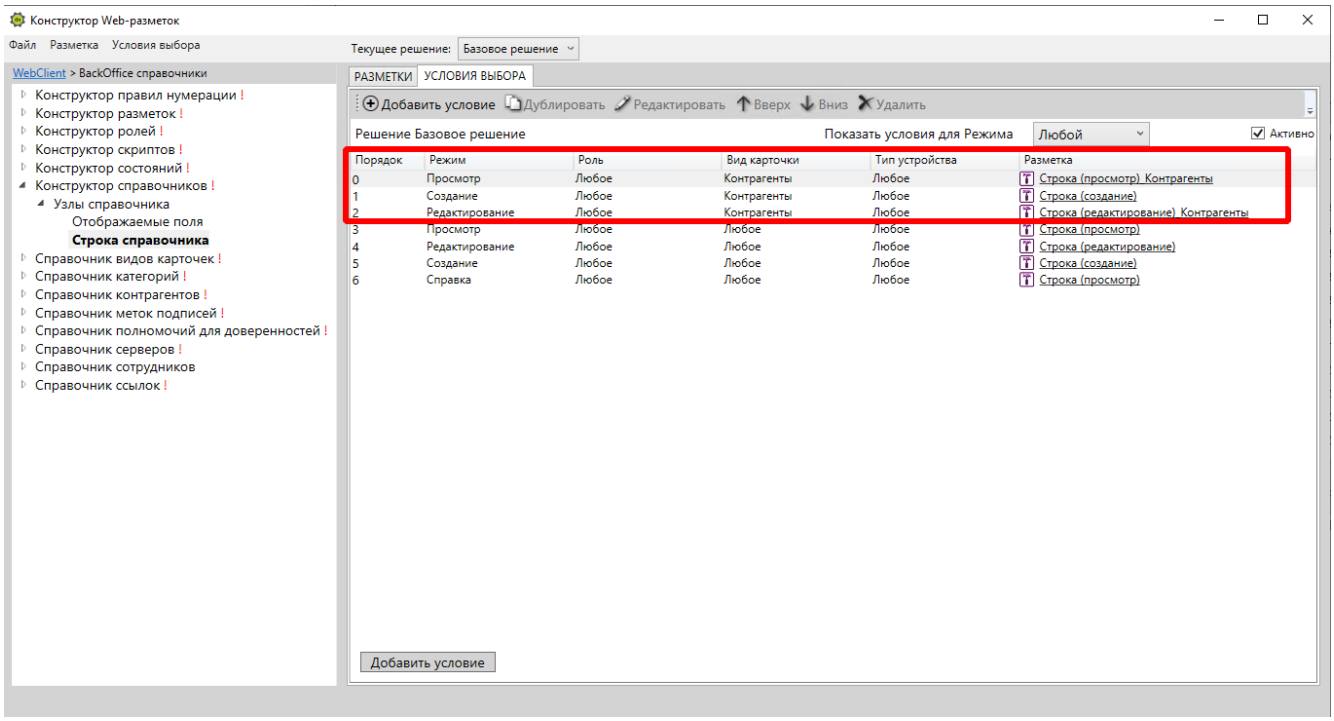


Рисунок 527. Повышение приоритета условий выбора

10. Убедитесь, что в Web-клиенте разметка отображается корректно.

    Для быстрого отображения изменений на период настройки можно изменить значение параметра `SharedDictionaryRefreshMode` в конфигурационном файле модуля, см. подробнее в разделе "Функциональность "Общие справочники"".

Описание стандартных элементов управления

В данном разделе приведено описание стандартных ЭУ, предоставляемых с программой Конструктор Web-разметок. Описание включает перечень настраиваемых свойств и событий элементов управления.

В зависимости от назначения разметки и типа карточек, для добавления и настройки могут быть недоступны некоторые элементы управления или свойства элементов управления. В описании будут отмечены такие ограничения.

Описание некоторых свойств отмечено тегом:

- Тег `Локализуемый` указан, когда для свойства могут быть установлены несколько значений, зависящих от **языка локализации**.

- Тег `Адаптивный` указан, когда для свойства могут быть установлены несколько значений, зависящих от **типа клиентского устройства**.

Разметка

Элемент `Разметка` является корневым элементом разметок (карточек, основного меню и прочих). Данный элемент автоматически добавляется в корень структуры разметки и не может быть удалён, самостоятельное его добавление недоступно.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Данные

Расширенные источники данных

Определяет дополнительные источники данных, с которыми могут работать элементы управления разметки.

По умолчанию элементы управления могут быть связаны только с данными текущей карточки. С помощью настройки *Расширенные источники данных* можно добавить карточки, связанные с текущей карточкой, которые также могут выступать источниками данных для элементов управления разметки.

Настройка позволяет определить несколько дополнительных источников данных и указать для каждого источника режим загрузки данных:

- Синхронно при открытии карточки (значение по умолчанию).
- Асинхронно — при снятом флаге `Загружать синхронно`.

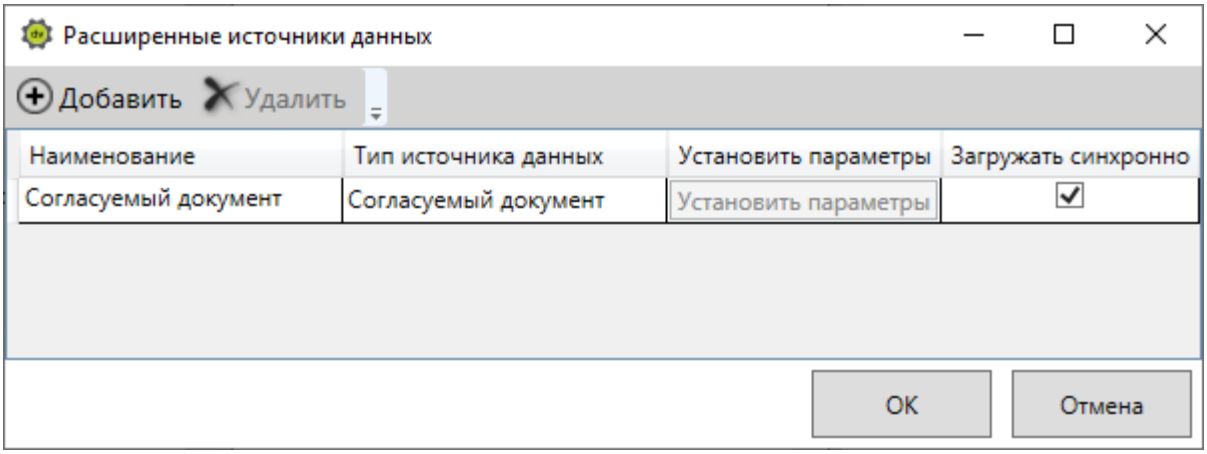


Рисунок 528. Настройка расширенных источников данных

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

События

Перед изменением разметки

Вызывается перед изменением разметки.

Перед изменением состояния карточки

Вызывается перед изменением состояния карточки. *Для использования в разметке карточек.*

Перед открытием карточки

Вызывается перед открытием карточки. *Для использования в разметке карточек.*

Перед сохранением карточки

Вызывается перед сохранением карточки. *Для использования в разметке карточек.*

Перед удалением карточки

Вызывается перед удалением карточки. *Для использования в разметке карточек.*

Подготовка к сохранению карточки

Вызывается при сохранении карточки до события `Перед сохранением карточки`. *Для использования в разметке карточек.*

После загрузки всех ЭУ

Вызывается после загрузки всех элементов разметки. В зависимости от режима загрузки данных, на момент вызова события:

- Данные элементов управления будут загружены. Применимо для ЭУ с синхронной загрузкой данных и ЭУ с загрузкой данных из внешнего источника, включая источники с асинхронным режимом загрузки.
- Данные могут быть загружены не полностью. Применимо для ЭУ с асинхронным режимом загрузки данных, например, Задания, Ссылки и т.п.



Если в разметке размещен элемент управления Вкладки, то событие вызывается только после загрузки элементов первой активной вкладки.

После открытия карточки

Вызывается после открытия карточки. *Для использования в разметке карточек.*

После сохранения карточки

Вызывается после сохранения карточки. *Для использования в разметке карточек.*

После удаления карточки

Вызывается после удаления карточки. *Для использования в разметке карточек.*

При наведении курсора

Вызывается при входе курсора мыши в область разметки.

При отведении курсора

Вызывается, когда курсор мыши покидает область разметки.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.



Если отсутствуют ограничения, содержимое разметки будет распределяться на всю ширину окна браузера, что может оказаться неудобным для пользователя. Ограничить область отображения разметки можно с помощью стандартного стиля **center-screen**, который нужно указать в свойстве Дополнительные css классы. При этом разметка будет отображаться по центру страницы с

добавлением боковых отступов, ширина которых зависит от ширины окна браузера.

WebFrame



После изменения разметок данной локации потребуется перезапуск службы Сервера Web-клиента, чтобы отобразить изменения.

Кнопка "Помощь"

Элемент управления Кнопка "Помощь" предназначен для открытия страницы с пользовательской документацией модуля Web-клиент.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка главного меню

Элемент управления Кнопка главного меню предназначен для открытия основного меню Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Показать справа

Включает режим отображения главного меню справа. По умолчанию меню отображается с левой стороны окна.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер Body

Элемент управления Контейнер Body является контейнером для элементов управления, которые должны отображаться в основной области окна Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер главного меню

Элемент управления Контейнер главного меню является контейнером, в котором отображается содержимое основного меню Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер заголовка

Элемент управления Контейнер заголовка является контейнером для элементов управления, которые должны отображаться в верхней части окна Web-клиента.

В контейнер могут быть добавлены только следующие элементы управления:

- Кнопка главного меню
- Логотип компании
- Панель поиска
- Панель поиска по справочнику
- Панель пользователя
- Кнопка "Помощь"
- Название страницы

- Кнопка
- Строка
- Число
- Url
- Раскрывающийся список
- Метка
- Группа радиокнопок

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Высота в пикселях

Свойство определяет высоту контейнера. По умолчанию указано текущее значение высоты.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента

управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер содержимого

Элемент управления Контейнер содержимого является контейнером для размещения элементов, отображаемых в области представления Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Логотип компании

Элемент управления Логотип компании предназначен для отображения логотипа организации в окне Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный То же относится к логотипу. Если флаг снят, логотип и название папки не отображается нигде, кроме справочников и карточек. В карточках всегда остается заголовок ("Документ", "Задание", "Справочник сотрудников" и т.д.).

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Отображаться всегда

По умолчанию флаг установлен. Если его снять, то при переходе в папку логотип не будет заменён на название папки. Установить логотип можно, указав путь к нему в соответствующем поле.

Путь к файлу логотипа

В свойстве нужно указать путь к файлу логотипа в файловой системе.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Название страницы

Элемент управления Название страницы предназначен для отображения названия текущей страницы в верхней части окна Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заголовок

Текст, отображаемый в элементе управления. Если текст не указан, в элементе управления отображается название текущей веб-страницы.

Панель навигации

Элемент управления Панель навигации предназначен для размещения элементов, отображаемых в области навигации панели меню.

В контейнер могут быть добавлены только следующие элементы управления:

- Панель навигации – кнопка "Назад"
- Панель навигации – кнопка "Создать"
- Поиск по штрих-коду
- Кнопка
- Строка
- Число
- Url
- Раскрывающийся список
- Метка
- Группа радиокнопок

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Панель навигации — кнопка "Назад"

Элемент управления Панель навигации — кнопка "Назад" предназначен для добавления на Панель навигации кнопки возврата к предыдущей странице.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Панель навигации — кнопка "Создать"

Элемент управления Панель навигации — кнопка "Создать" предназначен для добавления на Панель навигации кнопки создания новой карточки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Панель поиска

Элемент управления Панель поиска предназначен для отображения строки поиска в окне Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Панель поиска по справочнику

Элемент управления Панель поиска по справочнику предназначен для отображения специальной строки поиска в справочниках Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Панель пользователя

Элемент управления Панель пользователя предоставляет панель настройки учётной записи пользователя, также содержащую ссылки для установки дополнительных компонентов Web-клиента. Элемент управления представляет собой кнопку, которая открывает меню с панелью пользователя.

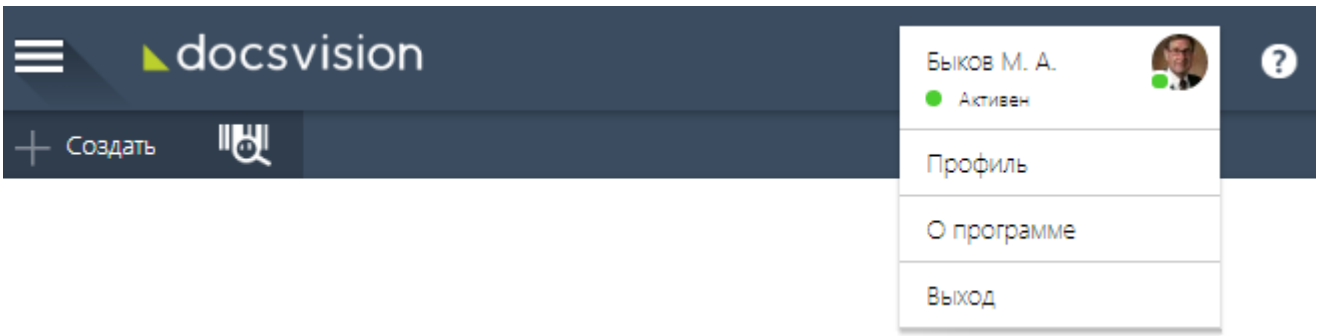


Рисунок 529. Меню элемента "Панель пользователя"

Данный элемент управления предназначен для использования только в разметке "корень WebFrame". ЭУ не может быть удалён/добавлен.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Отображать пользовательские данные

Включает показ фамилии и имени пользователя рядом с кнопкой открытия "Панели пользователя".

Отображать текущий статус

Включает показ статуса (значка и текстовой метки) учётной записи пользователя в элементе управления.

Отображать фото

Включает показ фото из Справочника сотрудников. В ЭУ отображается первое фото пользователя из Справочника сотрудников.

Формат отображения

Устанавливает формат данных о пользователе, отображаемых рядом с кнопкой открытия "Панели пользователя":

- *Строка отображения*. Будет использоваться формат отображения, установленный в Справочнике сотрудников.
- *Фамилия и инициалы*. Будут отображаться фамилия и инициалы пользователя.

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Поиск по штрих-коду

Элемент управления Поиск по штрих-коду предназначен для поиска карточек по штрих-коду. Элемент управления представляет собой кнопку, которая открывает поле для ввода штрих-кода.



Рисунок 530. Кнопка "Поиск по штрих-коду" с открытой строкой поиска

Данный элемент управления предназначен для использования только в разметке "корень WebFrame". ЭУ не может быть удалён/добавлен.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на кнопке открытия строки поиска. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства `Видимость`:

- Когда флаг `Видимость` установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг `Видимость` установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг `Видимость` НЕ установлен, ЭУ всегда скрыт.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим отображения

Свойство определяет способ открытия найденной карточки:

- ***Текущая вкладка*** — найденная карточка будет открыта в текущей вкладке.
- ***Новая вкладка*** — найденная карточка будет открыта в новой вкладке с переходом на данную вкладку.
- ***Новое окно браузера*** — найденная карточка будет открыта в новом окне веб-браузера.
- ***Потоковое считывание*** — найденная карточка не открывается, а добавляется в очередь для открытия. Количество карточек в очереди отображается на кнопке открытия карточек в поле ввода штрих-кода.



Рисунок 531. "Поиск по штрих-коду" в режиме потокового ввода

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Содержимое

Элемент управления Содержимое является контейнером, в котором отображается содержимое области представления Web-клиента: карточка, представление папки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Фон

Свойство позволяет задать фон для разметки.

Доступен выбор из двух вариантов:

- ***Без фона***.
- ***Специальный фон***.

При выборе ***Специальный фон*** становятся доступными следующие два свойства.

Фоновое изображение

В свойстве указывается путь к локальному файлу. Если по указанному пути есть картинка, она будет использоваться как пользовательский фон для главной страницы.

Картинка будет также использована как фон для папок, если задано свойство Стандартный фон для элемента **root** в локациях *Папка*, *Последние карточки*, *Результаты поиска*, *Главная страница*. Если свойство не задано, для главной страницы и папок будет использоваться фон по умолчанию.

Отображение фона

Свойство Отображение фона — выпадающий список с вариантами ***Растянуть*** или ***Замостить***.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Главная страница



После изменения разметок данной группы, для отображений изменений потребуется перезапуск службы сервера Web-клиента.

Виджет / Группа папок

Элемент управления `Виджет / Группа папок` предназначен для отображения группы папок на *Дашборде*.

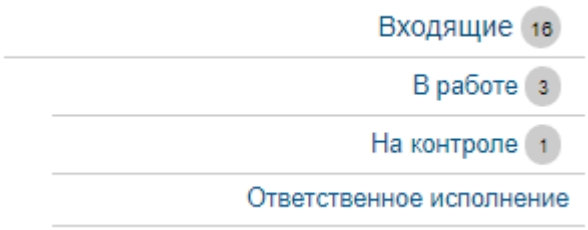


Рисунок 532. Пример виджета "Группа папок"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Непрозрачность

Свойство регулирует прозрачность грида. Прозрачность может быть настроена в диапазоне от `0` до `100` (в процентах), где `0` означает максимальную прозрачность, а `100` — максимальную непрозрачность. Значение по умолчанию — `100`.

Отступы

Устанавливает дополнительные внешние отступы в *Блоке*:

- ***true*** — по границе *Блока* будут добавлены дополнительные отступы.
- ***false*** — *Блок* в карточке отображается со стандартными отступами от других ЭУ.

Цвет страницы

Определяет цвет оформления страницы, которая открывается при переходе по ссылке ЭУ. Варианты: ***Папка*** — темно-синий цвет, ***Задача*** — зеленый цвет, ***Документ*** — синий цвет.

Данные

Каталог

Идентификатор папки, которая должна открываться при щечке по заголовку виджета.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Принудительный поиск в виртуальной папке

Если флаг установлен, для виртуальной папки, указанной в Каталог, будет подсчитываться число непрочитанных карточек. Такое поведение повышает нагрузку на систему, чтобы не подсчитывать число непрочитанных карточек виртуальной папки, снимите флаг.

Настройка не распространяется на обычную папку. Также свойство не влияет на подсвечивание непрочитанных карточек, за это отвечает настройка в Windows-клиенте, см. раздел "Отображение количества непрочитанных карточек в папке" в документации модуля Windows-клиент.

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Виджет / Известная папка

Элемент управления Виджет / Известная папка предназначен для отображения ссылки на папку на *Дашборде*.



Рисунок 533. Пример виджета "Известная папка"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Непрозрачность

Свойство регулирует прозрачность грида. Прозрачность может быть настроена в диапазоне от `0` до `100` (в процентах), где `0` означает максимальную прозрачность, а `100` — максимальную непрозрачность. Значение по умолчанию — `100`.

Цвет страницы

Определяет цвет оформления страницы, которая открывается при переходе по ссылке ЭУ. Варианты: ***Папка*** — темно-синий цвет, ***Задача*** — зеленый цвет, ***Документ*** — синий цвет.

Данные

Каталог

Идентификатор папки, которая должна открываться при щечке по заголовку виджета.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Принудительный поиск в виртуальной папке

Если флаг установлен, для виртуальной папки, указанной в `Каталог`, будет подсчитываться число непрочитанных карточек. Такое поведение повышает

нагрузку на систему, чтобы не подсчитывать число непрочитанных карточек виртуальной папки, снимите флаг.

Настройка не распространяется на обычную папку. Также свойство не влияет на подсвечивание непрочитанных карточек, за это отвечает настройка в Windows-клиенте, см. раздел "Отображение количества непрочитанных карточек в папке" в документации модуля Windows-клиент.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Виджет / Карточки папки

Элемент управления Виджет / Карточки папки предназначен для отображения содержимого папки на *Дашборде*.

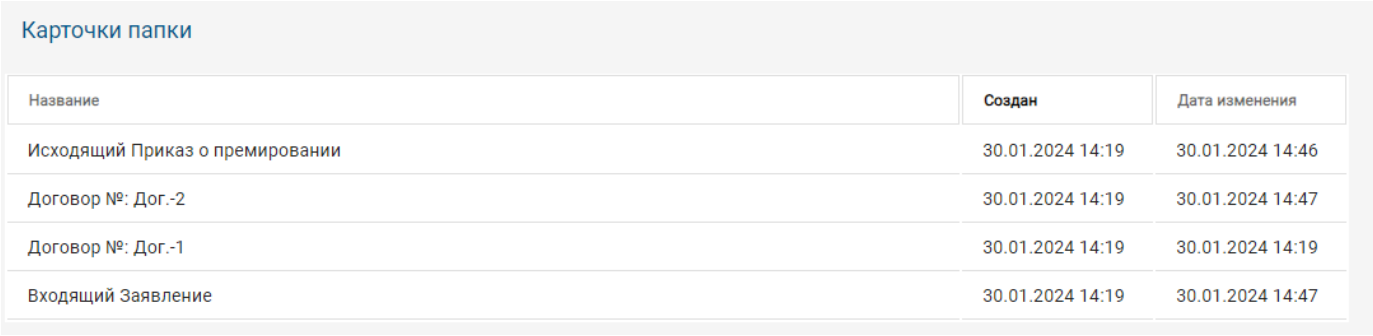


Рисунок 534. Пример виджета "Карточки папки"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заголовок

Текст, отображаемый над списком карточек.

Локализуемый

Количество карточек

Количество отображаемых карточек. По умолчанию: 5.

Использовать ленту

Регулирует использование режима "Лента" или "Грид" в виджете. Если флаг активен, карточки отображаются в режиме ленты, если флаг снят — в режиме грида. По умолчанию флаг активен.

Адаптивный

Колонка сортировки

Столбец, по которому выполняется сортировка. По умолчанию используется сортировка папки. Сортировка по убыванию работает, только если в поле *Колонка сортировки* указано название колонки.

Сортировка по убыванию

Определяет направление сортировки. Флаг установлен — по убыванию, иначе — по возрастанию.

Данные

Папка

Идентификатор папки, карточки которой отображаются.

Представление

Идентификатор представления, используемого для отображения карточек. По умолчанию используется представление папки.

Поведение

Непрозрачность

Свойство регулирует прозрачность грида. Прозрачность может быть настроена в диапазоне от `0` до `100` (в процентах), где `0` означает максимальную прозрачность, а `100` — максимальную непрозрачность. Значение по умолчанию — `100`.

Для заголовка грида прозрачность не применяется, карточки из списка при прокрутке будут скрыты под заголовком. Поэтому при добавлении прозрачности возникнет наложение элементов грида.

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента

управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Виджет / Последние карточки

Элемент управления Последние карточки предназначен для отображения списка последних карточек на *Дашборде*.

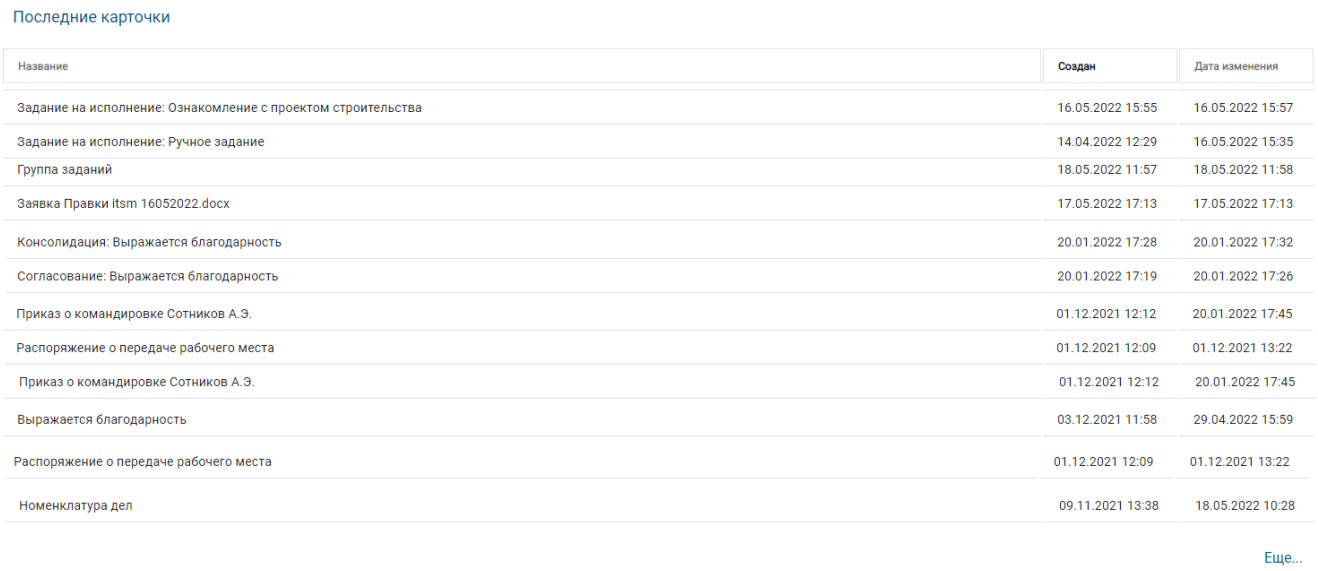


Рисунок 535. Пример виджета "Последние карточки"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заголовок

Текст, отображаемый над списком последних карточек.

Локализуемый

Использовать ленту

Регулирует использование режима "Лента" или "Грид" в виджете. Если флаг активен, карточки отображаются в режиме ленты, если флаг снят — в режиме грида. По умолчанию флаг активен.

Адаптивный

Поведение

Непрозрачность

Свойство регулирует прозрачность грида. Прозрачность может быть настроена в диапазоне от `0` до `100` (в процентах), где `0` означает максимальную прозрачность, а `100` — максимальную непрозрачность. Значение по умолчанию — `100`.

Для заголовка грида прозрачность не применяется, карточки из списка при прокрутке будут скрыты под заголовком. Поэтому при добавлении прозрачности возникнет наложение элементов грида.

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Главное меню



После изменения разметок данной локации потребуется перезапуск службы Сервера Web-клиента, чтобы отобразить изменения.

Кнопка / Закрепить меню

Элемент управления Кнопка / Закрепить меню — кнопка закрепления/открепления панели, содержащей дерево папок.



Рисунок 536. Кнопка "Закрепить меню" на панели дерева папок

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Иконка закрепленного меню

Название CSS класса с иконкой закрепленного меню.

Иконка открепленного меню

Название CSS класса с иконкой открепленного меню.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Данные

Имя ЭУ "Главное меню"

Название элемента управления типа Контейнер / Главное меню, в котором размещена данная кнопка.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка / Настроить

Элемент управления `Кнопка / Настроить меню` предназначен для отображения меню настройки панели *Основное меню*.



Рисунок 537. Кнопка "Настроить" в основном меню

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на кнопке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Данные

Имя ЭУ "Настраиваемый контейнер"

Название элемента управления типа "Контейнер / Настраиваемая область" с частью дерева папок, которую может изменять пользователь.

Имя ЭУ "Пользовательские папки"

Название элемента управления типа "Папки / Пользовательские папки", в котором размещены добавляемые пользователем папки.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Отображать пункт меню "Вернуть настройки по умолчанию"

Включает отображение пункта "Вернуть настройки по умолчанию" в меню кнопки.

Отображать пункт меню "Настроить панель"

Включает отображение пункта "Настроить панель" в меню кнопки.

Отображать пункт меню "Присоединить папку"

Включает отображение пункта "Добавить папку" в меню кнопки.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

После клика по кнопке "Обновить дерево папок"

Вызывается при щелчке мыши по пункту "Обновить" в меню кнопки.

После клика по кнопке "Настроить панель"

Вызывается при щелчке мыши по пункту "Настроить панель" в меню кнопки.

После клика по кнопке "Прикрепить папку"

Вызывается при щелчке мыши по пункту "Добавить папку" в меню кнопки.

После клика по кнопке "Восстановить настройки умолчанию"

Вызывается при щелчке мыши по пункту "Вернуть настройки по умолчанию" в меню кнопки.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка / Обновить

Элемент управления Кнопка / Обновить предназначен для обновления содержимого дерева папок.



Рисунок 538. Кнопка "Обновить" на панели дерева папок

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на кнопке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер / Главное меню

Элемент управления Контейнер / Главное меню является контейнером, в котором должны размещаться элементы панели *Дерево папок*.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При изменении состояния закрепления

Вызывается при закреплении/откреплении панели с помощью элемента управления Кнопка / Закрепить меню.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер / Группа элементов

Элемент управления Контейнер / Группа элементов является сворачиваемым контейнером для создания группы папок.

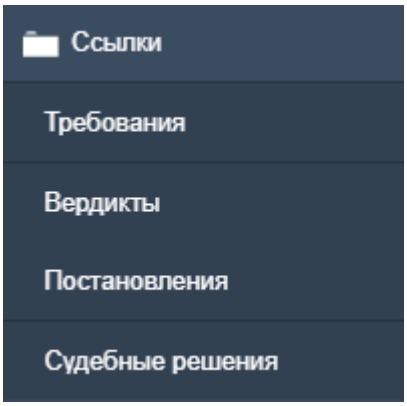


Рисунок 539. Элемент управления "Группа элементов" с вложенными элементами

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на кнопке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Анимация

Включает или выключает использование анимации при раскрытии группы.

Компактный режим

Включает или выключает минимизацию пространства, необходимого для отображения ЭУ.

Показывать корневой элемент

Включает или выключает отображения папок первого уровня, вложенных в Группу элементов.

Текст

Текст, отображаемый в заголовке элемента управления.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Развернут

Флаг установлен — содержимое ЭУ по умолчанию отображается, флаг снят — содержимое ЭУ по умолчанию скрыто.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Уровень автоматического раскрытия дочерних элементов

Указывает уровень вложенности папок, до которого вложенные в ЭУ папки будут раскрыты по умолчанию. При значении `0` автоматическое раскрытие папок не выполняется. Если установлено любое значние, кроме `0`, автоматически активируется свойство Развернут.

События

Перед выбором

Вызывается перед изменением свойства `isSelected`.

Перед сворачиванием/разворачиванием элемента

Вызывается перед сворачиванием/разворачиванием элемента управления.

После выбора

Вызывается после изменения свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

После сворачивания/разворачивания элемента

Вызывается после сворачивания/разворачивания элемента управления.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер / Компактная группа

Элемент управления Контейнер / Компактная группа является контейнером для создания группы папок.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на кнопке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Компактный режим

Включает или выключает минимизацию пространства, необходимого для отображения ЭУ.

Показывать корневой элемент

Включает или выключает отображения папок первого уровня, вложенных в Группу элементов.

Текст

Текст, отображаемый в заголовке элемента управления.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента

управления. Перечисляются через пробел.

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Уровень автоматического раскрытия дочерних элементов

Указывает уровень вложенности папок, до которого вложенные в ЭУ папки будут раскрыты по умолчанию. При значении `0` автоматическое раскрытие папок не выполняется. Если установлено любое значние, кроме `0`, автоматически активируется свойство `Развернут`.

События

Перед выбором

Вызывается перед изменением свойства `isSelected`.

Перед сворачиванием/разворачиванием элемента

Вызывается перед сворачиванием/разворачиванием элемента управления.

После выбора

Вызывается после изменения свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

После сворачивания/разворачивания элемента

Вызывается после сворачивания/разворачивания элемента управления.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер / Настраиваемая область

Элемент управления Контейнер / Настраиваемая область является контейнером, содержимое которого может быть настроено пользователем.

Режим настройки включается кнопкой Настроить.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

После завершения конфигурации

Вызывается после выхода из режима настройки.

После начала конфигурации

Вызывается при входе в режим настройки.

После сохранения конфигурации

Вызывается при сохранении изменений.

Перед завершением конфигурации

Вызывается перед выходом из режима настройки.

Перед началом конфигурации

Вызывается перед входом в режим настройки.

Перед сохранением конфигурации

Вызывается перед сохранением изменений.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер / Панель элемента меню

Элемент управления Контейнер / Панель элемента меню является контейнером для размещения элементов Кнопка / Закрепить меню, Кнопка / Настроить, Кнопка / Обновить.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Локация

Данный ЭУ не создаёт собственной строки в дереве папок, а встраивается в существующую строку (располагается в правой части строки). Настройка Локация определяет, в какую строку должен быть встроен данный ЭУ:

- ***Ниже***. Панель элемента меню будет встроена в элемент, расположенный ниже него в структуре разметки.
- ***Выше***. Панель элемента меню будет встроена в элемент, расположенный выше него в структуре разметки.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер / Простой контейнер

Элемент управления Контейнер / Простой контейнер является контейнером для размещения других элементов управления.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Уровень автоматического раскрытия дочерних элементов

Указывает уровень вложенности папок, до которого вложенные в ЭУ папки будут раскрыты по умолчанию. При значении `0` автоматическое раскрытие папок не выполняется. Если установлено любое значние, кроме `0`, автоматически активируется свойство `Развернут`.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Папка / Известная папка

Элемент управления `Папка / Известная папка` предназначен для добавления папки с известным идентификатором в дерево папок.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на папке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Анимация

Включает или выключает использование анимации при раскрытии папки.

Всегда добавлять отступ для кнопки разворачивания

Сдвигает название папки вправо на размер кнопки , когда кнопка не отображается (нет вложенных папок).

Дочерние папки в компактном режиме

Включает или выключает минимизацию пространства, необходимого для отображения подчиненных папок.

Компактный режим

Включает или выключает минимизацию пространства, необходимого для отображения данной папки.

Показывать иконки у дочерних папок

Включает или выключает отображение иконок вложенных папок.

Показывать иконку

Включает или выключает отображение иконок у данной папки.

Показывать корневой элемент

Включает или выключает отображения данной папки. Если флаг снят, будут отображаться только вложенные папки.

Показывать подпапки

Включает или выключает отображения вложенных папок.

Текст

Текст, отображаемый в строке папки. Если значение — пустая строка, в строке ЭУ отображается название папки.

Уровень предзагрузки дочерних папок

Определяет уровень вложенности папок данного каталога, до которого подпапки будут загружены без дополнительных запросов. Данные подпапок с более высоким уровнем вложенности загружаются через отдельные клиент-серверные запросы. Должно быть указано целое число.

Цвет страницы

Определяет цвет оформления страницы для отображения содержимого папки. Варианты: ***Папка*** — темно-синий цвет, ***Задача*** — зеленый цвет, ***Документ*** — синий цвет.

Управление папками

Включает/отключает меню папки, позволяющее создавать подпапки, настраивать и удалять данную папку, создавать подпапки. По умолчанию свойство отключено.

Данные

Каталог

Идентификатор папки, на которую ссылается ЭУ.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Принудительный поиск в виртуальной папке

Если флаг установлен, для виртуальной папки, указанной в `Каталог`, будет подсчитываться число непрочитанных карточек. Такое поведение повышает нагрузку на систему, чтобы не подсчитывать число непрочитанных карточек виртуальной папки, снимите флаг.

Настройка не распространяется на обычную папку. Также свойство не влияет на подсвечивание непрочитанных карточек, за это отвечает настройка в Windows-клиенте, см. раздел "Отображение количества непрочитанных карточек в папке" в документации модуля Windows-клиент.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Уровень автоматического раскрытия дочерних элементов

Указывает уровень вложенности папок, до которого вложенные в ЭУ папки будут раскрыты по умолчанию. При значении `0` автоматическое раскрытие папок не выполняется. Если установлено любое значние, кроме `0`, автоматически активируется свойство `Развернут`.

События

Перед выбором

Вызывается перед изменением свойства `isSelected`.

После выбора

Вызывается после изменения свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Папка / Результаты поиска

Элемент управления `Папка / Результаты поиска` предназначен для добавления папки с результатами последнего поискового запроса в дерево папок.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на папке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Компактный режим

Включает или выключает минимизацию пространства, необходимого для

отображения данной папки.

Текст

Текст, отображаемый в строке папки. Если значение — пустая строка, в строке ЭУ отображается название "Поиск".

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

После выбора

Вызывается после изменения свойства `isSelected`.

Перед выбором

Вызывается перед изменением свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Папка / Шаблонная папка

Элемент управления `Папка / Шаблонная папка` предназначен для добавления шаблонной папки пользователя в дерево папок.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на папке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Всегда добавлять отступ для кнопки разворачивания

Сдвигает название папки вправо на размер кнопки , когда кнопка не отображается (нет вложенных папок).

Дочерние папки в компактном режиме

Включает или выключает минимизацию пространства, необходимого для отображения подчиненных папок.

Компактный режим

Включает или выключает минимизацию пространства, необходимого для отображения данной папки.

Показывать иконки у дочерних папок

Включает или выключает отображение иконок вложенных папок.

Показывать иконку

Включает или выключает отображение иконок у данной папки.

Показывать корневой элемент

Включает или выключает отображения данной папки. Если флаг снят, будут отображаться только вложенные папки.

Показывать подпапки

Включает или выключает отображения вложенных папок.

Текст

Текст, отображаемый в строке папки. Если значение — пустая строка, в строке ЭУ отображается название папки.

Уровень предзагрузки дочерних папок

Определяет уровень вложенности папок данного каталога, до которого подпапки будут загружены без дополнительных запросов. Данные подпапок с более высоким уровнем вложенности загружаются через отдельные клиент-серверные запросы. Должно быть указано целое число.

Цвет страницы

Определяет цвет оформления страницы для отображения содержимого папки. Варианты: ***Папка*** — темно-синий цвет, ***Задача*** — зеленый цвет, ***Документ*** —

синий цвет.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Принудительный поиск в виртуальной папке

Если флаг установлен, для виртуальной папки, указанной в Каталог, будет подсчитываться число непрочитанных карточек. Такое поведение повышает нагрузку на систему, чтобы не подсчитывать число непрочитанных карточек виртуальной папки, снимите флаг.

Настройка не распространяется на обычную папку. Также свойство не влияет на подсвечивание непрочитанных карточек, за это отвечает настройка в Windows-клиенте, см. раздел "Отображение количества непрочитанных карточек в папке" в документации модуля Windows-клиент.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Уровень автоматического раскрытия дочерних элементов

Указывает уровень вложенности папок, до которого вложенные в ЭУ папки будут раскрыты по умолчанию. При значении `0` автоматическое раскрытие папок не выполняется. Если установлено любое значние, кроме `0`, автоматически активируется свойство `Развернут`.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

Перед выбором

Вызывается перед изменением свойства `isSelected`.

Перед сворачиванием/разворачиванием элемента

Вызывается перед сворачиванием/разворачиванием элемента управления.

После выбора

Вызывается после изменения свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

После сворачивания/разворачивания элемента

Вызывается после сворачивания/разворачивания элемента управления.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Папки / Групповые папки

Элемент управления Папка / Групповые папки предназначен для добавления папок рабочей группы пользователя в дерево папок.

ЭУ отображается в дереве папок, если папки рабочей группы настроены для текущего пользователя.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на папке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Компактный режим

Включает или выключает минимизацию пространства, необходимого для отображения данной папки.

Папки в компактном режиме

Включает или выключает минимизацию пространства, необходимого для

отображения папок группы.

Показывать иконки

Включает или выключает отображение иконок у папок группы.

Показывать иконки у дочерних папок

Включает или выключает отображение иконок у вложенных папок группы.

Показывать корневой элемент

Флаг установлен — отображаются корневые папки группы и их дочерние папки, флаг снят — отображаются только дочерние папки.

Текст

Текст, отображаемый в строке папки.

Управление папками

Включает/отключает меню папки, позволяющее создавать подпапки, настраивать и удалять данную папку, создавать подпапки. По умолчанию свойство отключено.

Уровень предзагрузки дочерних папок

Определяет уровень вложенности папок данного каталога, до которого подпапки будут загружены без дополнительных запросов. Данные подпапок с более высоким уровнем вложенности загружаются через отдельные клиент-серверные запросы. Должно быть указано целое число.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Уровень автоматического раскрытия дочерних элементов

Указывает уровень вложенности папок, до которого вложенные в ЭУ папки будут раскрыты по умолчанию. При значении `0` автоматическое раскрытие папок не выполняется. Если установлено любое значние, кроме `0`, автоматически активируется свойство `Развернут`.

События

Перед выбором

Вызывается перед изменением свойства `isSelected`.

Перед сворачиванием/разворачиванием элемента

Вызывается перед сворачиванием/разворачиванием элемента управления.

После сворачивания/разворачивания элемента

Вызывается после сворачивания/разворачивания элемента управления.

После выбора

Вызывается после изменения свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Папки / Пользовательские папки

Элемент управления Папка / Пользовательские папки предназначен для добавления группы настраиваемых папок в дерево папок. В данной группе отображаются папки, подключенные пользователем.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на папке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Анимация

Включает или выключает использование анимации при раскрытии папки.

Компактный режим

Включает или выключает минимизацию пространства, необходимого для отображения данной папки.

Папки в компактном режиме

Включает или выключает минимизацию пространства, необходимого для отображения папок группы.

Показывать иконки

Включает или выключает отображение иконок у пользовательских папок.

Показывать корневой элемент

Включает или выключает отображения данной папки. Если флаг снят, будут отображаться только вложенные папки.

Текст

Текст, отображаемый в строке папки. Если значение — пустая строка, в строке ЭУ отображается название папки.

Управление папками

Включает/отключает меню папки, позволяющее создавать подпапки, настраивать и удалять данную папку, создавать подпапки. По умолчанию свойство отключено.

Уровень предзагрузки дочерних папок

Определяет уровень вложенности папок данного каталога, до которого подпапки будут загружены без дополнительных запросов. Данные подпапок с более высоким уровнем вложенности загружаются через отдельные клиент-серверные запросы. Должно быть указано целое число.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Уровень автоматического раскрытия дочерних элементов

Указывает уровень вложенности папок, до которого вложенные в ЭУ папки будут раскрыты по умолчанию. При значении `0` автоматическое раскрытие папок не выполняется. Если установлено любое значние, кроме `0`, автоматически активируется свойство Развернут.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

Перед выбором

Вызывается перед изменением свойства `isSelected`.

Перед сворачиванием/разворачиванием элемента

Вызывается перед сворачиванием/разворачиванием элемента управления.

После выбора

Вызывается после изменения свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

После сворачивания/разворачивания элемента

Вызывается после сворачивания/разворачивания элемента управления.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Скрипт / Настройка пользовательских папок

Скрипт `Настройка пользовательских папок` добавляет в разметку дерева папок клиентский скрипт, обеспечивающий возможность настройки дерева папок пользователем.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например,

JSON).

Данные

Имя ЭУ "Настраиваемый контейнер"

Название экземпляра ЭУ "Контейнер / Настраиваемая область" в разметке. Для данного ЭУ будет обеспечена возможность настройки папок пользователем.

Имя ЭУ "Пользовательские папки"

Название экземпляра ЭУ "Папки / Пользовательские папки" в разметке. Для данного ЭУ будет обеспечена возможность подключения/отключения папок пользователем.

Ссылка / HTML страница

Элемент управления `Ссылка / HTML страница` добавляет в дерево папок ссылку для открытия HTML страницы с известным адресом. Страница ссылки открывается в области просмотра карточки/папки Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на ссылке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
```

```
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Заголовок страницы

Текст, отображаемый справа от открытия/скрытия дерева папок, при открытии ссылки.

Локализуемый

Компактный режим

Включает или выключает минимизацию пространства, необходимого для отображения данной ссылки.

Текст

Текст, отображаемый в строке ссылки в дереве папок.

Локализуемый

Цвет страницы

Определяет цвет оформления страницы для отображения содержимого папки. Варианты: ***Папка*** — темно-синий цвет, ***Задача*** — зеленый цвет, ***Документ*** — синий цвет.

Данные

Адрес

Адрес HTML-страницы в формате `http://www.company.org`.



Целевая страница должна разрешать встраивание в другие страницы/фреймы.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед выбором

Вызывается перед изменением свойства `isSelected`.

После выбора

Вызывается после изменения свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Ссылка / Локация

Элемент управления Ссылка / Локация добавляет в дерево папок ссылку для открытия локации.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на ссылке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Заголовок страницы

Текст, отображаемый справа от открытия/скрытия дерева папок, при открытии ссылки.

Локализуемый

Компактный режим

Включает или выключает минимизацию пространства, необходимого для отображения данной ссылки.

Максимальная высота

Устанавливает ограничение по высоте для элемента управления. Значение в пикселях.

Текст

Текст, отображаемый в строке ссылки в дереве папок.

Локализуемый

Цвет страницы

Определяет цвет оформления страницы для отображения содержимого папки. Варианты: ***Папка*** — темно-синий цвет, ***Задача*** — зеленый цвет, ***Документ*** — синий цвет.

Данные

Локация

Локация, которая должна открываться при переходе по ссылке.



Поддерживается локации только из группы локаций "WebFrame".

Источник локации

Выбор источника локации. Для выбора доступны варианты ***Локация*** и ***Связанная карточка/строка***.

Параметры загрузки разметки

Не используется. **Оставить пустым.**

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Режим

Режим отображения элемента управления. Для выбора доступны варианты: ***любой***, ***просмотр***, ***редактирование***, ***создание***.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед выбором

Вызывается перед изменением свойства `isSelected`.

После выбора

Вызывается после изменения свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Ссылка / Страница Web-клиента

Элемент управления `Ссылка / Страница Web-клиента` добавляет в дерево папок ссылку для открытия страницы Web-клиента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на ссылке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Компактный режим

Включает или выключает минимизацию пространства, необходимого для отображения данной ссылки.

Текст

Текст, отображаемый в строке ссылки в дереве папок.

Локализуемый

Данные

Адрес

Адрес страницы Web-клиента в формате `#/Folder/RecentCards`.

Поведение

HTML Accesskey

См. https://www.w3schools.com/TAGS/att_accesskey.asp.

HTML Download

См. https://www.w3schools.com/TAGS/att_a_download.asp.

HTML Hreaflang

См. https://www.w3schools.com/TAGS/att_a_hreflang.asp.

HTML Target

См. https://www.w3schools.com/TAGS/att_a_target.asp.

HTML Type

См. https://www.w3schools.com/TAGS/att_a_type.asp.

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Настраиваемый

Включает или выключает режим, при котором пользователь может скрыть ЭУ. ЭУ должен находиться внутри контейнера Настраиваемая область.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Уровень автоматического раскрытия дочерних элементов

Указывает уровень вложенности папок, до которого вложенные в ЭУ папки будут раскрыты по умолчанию. При значении `0` автоматическое раскрытие папок не выполняется. Если установлено любое значние, кроме `0`, автоматически активируется свойство `Развернут`.

События

Перед выбором

Вызывается перед изменением свойства `isSelected`.

После выбора

Вызывается после изменения свойства `isSelected`.

После изменения настройки видимости пункта

Вызывается после изменения пользователем признака видимости элемента в дереве папок.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Групповые операции



После изменения разметок данной локации потребуется перезапуск службы Сервера Web-клиента, чтобы отобразить изменения.

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame** › **Компоненты папки** › **Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".
- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame** › **Компоненты папки** › **Групповые операции с карточками**.

Групповая операция / Делегирование

Элемент управления `Групповая операция / Делегирование` вызывает операцию делегирования для группы выбранных карточек Задание.

ЭУ может быть добавлен в разметку локации *Групповые операции с карточками* и в корневой элемент для ЭУ групповых операций (`BatchOperationsNode`).

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame › Компоненты папки › Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".
- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame › Компоненты папки › Групповые операции с карточками**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый на кнопке делегирования.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Групповая операция / Копирование ярлыков

Элемент управления Групповая операция / Копирование ярлыков вызывает операцию копирования ярлыков группы выбранных карточек из текущей папки в выбранную. Перед выполнением операции будет представлена возможность выбора папки, в которую копируются ярлыки.

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame › Компоненты папки › Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".
- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame › Компоненты папки › Групповые операции с карточками**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый на кнопке перемещения.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Групповая операция / Перемещение ярлыков

Элемент управления Групповая операция / Переместить ярлык вызывает операцию перемещения ярлыков группы выбранных карточек из текущей папки в выбранную. Перед выполнением операции будет представлена возможность выбора папки, в которую перемещаются ярлыки.

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame › Компоненты папки › Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".
- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame › Компоненты папки › Групповые операции с карточками**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый на кнопке перемещения.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Групповая операция / Прочитать

Элемент управления Групповая операция / Прочитать позволяет отметить выбранные карточки прочитанными.

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame › Компоненты папки › Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".
- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame › Компоненты папки › Групповые операции с карточками**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый на кнопке прочтения.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Групповая операция / Скачать файлы

Элемент управления Групповая операция / Скачать файлы вызывает операцию скачивания файлов из карточки.

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame › Компоненты папки › Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".
- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame › Компоненты папки › Групповые операции с карточками**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый на кнопке операции.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Скачать дополнительные файлы

Когда свойство активно, скачивание дополнительных файлов разрешено.

Скачать основные файлы

Когда свойство активно, скачивание основных файлов разрешено.

Скачать только файлы с подписью

Когда свойство активно, скачиваются только файлы с подписью. В зависимости от выполненных для ЭУ настроек свойств будут скачаны основные и/или дополнительные.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Формат подписи

Когда свойство `Скачать только файлы с подписью` активно, подпись будет скачиваться в выбранном из списка формате: ***.p7s***, ***.sgn***, ***.sig***, ***.sign***.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Групповая операция / Удалить карточку

Элемент управления Групповая операция / Удалить карточку вызывает операцию удаления карточки.

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame › Компоненты папки › Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".
- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame › Компоненты папки › Групповые операции с карточками**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый на кнопке операции.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Групповая операция / Удаление ярлыков

Элемент управления Групповая операция / Удаление ярлыков вызывает операцию удаления ярлыков группы выбранных карточек из текущей папки.

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame › Компоненты папки › Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".
- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame › Компоненты папки › Групповые операции с карточками**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый на кнопке перемещения.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Сообщение об отсутствии доступных операций

Элемент управления Сообщение об отсутствии доступных операций предназначен для отображения сообщения об отсутствии общей групповой операции для карточек, выбранных в режиме групповых операций.

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame › Компоненты папки › Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".
- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame › Компоненты папки › Групповые операции с карточками**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст метки

Текст, отображаемый в сообщении об отсутствии доступных операций.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства `Видимость`:

- Когда флаг `Видимость` установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг `Видимость` установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг `Видимость` НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Состояние выделения карточек папки

Элемент управления `Состояние выделения карточек папки` предназначен для отображения количества карточек, выбранных в режиме групповых операций.

ЭУ может быть добавлен в разметку локации *Групповые операции с карточками* и в корневой элемент для ЭУ групповых операций (`BatchOperationsNode`).

Групповые операции могут быть добавлены:

- **Для нового грида:** в разметку *DV.Folder* в локации **WebFrame** ›

Компоненты папки › **Папка**. Новые операции добавляются внутрь ЭУ "Узел для групповых операций".

- **Для старого грида:** в разметку *DV.GridBatchOperations* в локации **WebFrame** › **Компоненты папки** › **Групповые операции с карточками**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости*

НЕ выбрана, ЭУ отображается всегда.

- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Компоненты папки

Web-клиент Конструктор Web-разметок

Компоненты папки

Описание элементов управления группы "Компоненты папки".

- Данные папки
- Карточки папки
- Тулбар папки
- Другие элементы управления группы в навигационном меню слева…

Данные папки



Данный элемент управления находится в разработке и может быть изменён в будущем.

Элемент управления может использоваться для отображения папки, выбранной в Главном Меню, либо папки, заданной в свойствах элемента управления. Также

может являться контейнером для стандартных и специальных элементов управления для работы с данными папки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Данные

Режим

Определяет режим отображения данных папки:

1. ***Источник данных***. При выборе данного режима элемент управления будет получать данные из указанного источника. Режим ***Источник данных*** позволяет выбрать свойства Источник данных, Расширенный источник данных и Поле данных.
2. ***Из адреса страницы*** (значение по умолчанию). При выборе данного режима элемент управления будет отображать папку, указанную в адресе страницы.
3. ***Известная папка***. При выборе данного режима элемент управления будет отображать папку с идентификатором указанным в поле Идентификатор папки.
4. ***Системная папка***. При выборе данного режима элемент управления будет отображать выбранную из списка системную папку.

Источник данных

Секция карточки, содержащая данные элемента управления. Свойство доступно, только если в свойстве Режим выбран вариант ***Источник данных***.

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки. Свойство доступно,

только если в свойстве Режим выбран вариант ***Источник данных***.

Поле данных

Поле карточки, содержащее ссылку на организацию *Справочника сотрудников*. Свойство доступно, только если в свойстве Режим выбран вариант ***Источник данных***.

Идентификатор папки

Позволяет указать идентификатор папки. Свойство доступно, только если в свойстве Режим выбран вариант ***Известная папка***.

Системная папка

Позволяет выбрать отображаемую в элементе управления системную папку. Свойство доступно, только если в свойстве Режим выбран вариант ***Системная папка***.

Доступные системные папки:

1. Последние карточки.
2. Результаты поиска.
3. Избранное.

Выбор представления

В данном свойстве отображается список всех представлений, доступных для выбора. Свойство позволяет указать разрешённые для выбора представления. Чтобы выбрать представление, установите флаг напротив его названия.

В окне настроек грида будет отображаться список выбранных в данном свойстве представлений. Если в свойстве не указано ни одно представление, будут отображаться все доступные пользователю представления.



Свойство доступно, только если в свойстве Режим выбран вариант ***Системная папка***.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Карточки папки



Данный элемент управления находится в разработке и может быть изменён в будущем.

Элемент управления Карточки папки предназначен для отображения содержимого папки с применением выбранного представления.

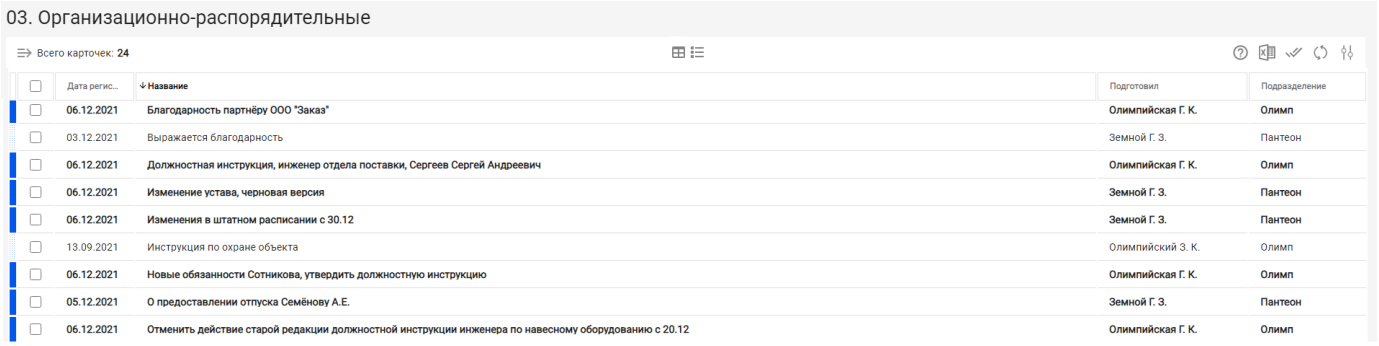


Рисунок 540. Пример элемента управления "Карточки папки"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Использовать ленту

Регулирует использование режима "Лента" или "Грид" в виджете. Если флаг активен, карточки отображаются в режиме ленты, если флаг снят — в режиме грида. По умолчанию флаг активен.

Адаптивный

Свойство может быть активировано отдельно или отключено для каждого типа устройств: Десктоп, Планшет, Телефон. Например, чтобы отключить свойство, для телефона, нужно снять флаг в строке "Телефон".

Непрозрачность

Свойство регулирует прозрачность грида. Прозрачность может быть настроена в диапазоне от `0` до `100` (в процентах), где `0` означает максимальную прозрачность, а `100` — максимальную непрозрачность. Значение по умолчанию — `100`.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Редактирование группировки

Свойство отвечает за доступность группировки значений в гриде. Когда флаг

напротив свойства установлен, группировка разрешена.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Контекстное меню включено

Свойство регулирует доступность контекстного меню грида. Когда флаг напротив свойства установлен, контекстное меню доступно. Свойство может быть активировано отдельно или отключено для каждого типа устройств: Десктоп, Планшет, Телефон. Например, чтобы отключить свойство, для телефона, нужно снять флаг в строке "Телефон".

Сортировка

Свойство отвечает за сортировку значений в колонках грида. Когда флаг напротив свойства установлен, сортировка разрешена.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Фильтрация

Свойство отвечает за доступность фильтрации значений в гриде. Когда флаг напротив свойства установлен, фильтрация разрешена.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Тулбар папки

Панель управления нового грида. Свойства панели управления регулируют операции, доступные для представления папки.

Элемент управления является контейнером, который может быть расширен добавлением новых команд. Панель управления можно расширять при помощи добавления ЭУ "Кнопка-иконка". Каждая новая кнопка соответствует команде.

Чтобы добавленная кнопка работала как команда, необходимо настроить её свойства, включая CSS класс, обработчик события, отображаемое название функции (название ЭУ). Если для кнопки не задан CSS класс, по умолчанию кнопка будет иметь следующий вид: . Если не задано название кнопки, будет отображаться название ЭУ по умолчанию.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Свойства категории "Внешний вид" могут быть активированы или отключены отдельно для каждого типа устройств: Десктоп, Планшет, Телефон. Например, чтобы отключить свойство, для телефона, нужно снять флаг в строке "Телефон".

Изменить отображение

Позволяет или запрещает пользователям переключаться между режимами: Грид или Лента.

Настройки

Регулирует доступность окна пользовательских настроек с панели инструментов.

Обновить

Позволяет отключить кнопку обновления данных в представлении.

Прочитать все

Позволяет отключить кнопку обновления данных в представлении.

Размер папки

Включает или отключает отображение количества карточек в папке.

Расположение дополнительных элементов

Определяет, где по отношению к стандартным кнопкам панели усправления будут располагаться дополнительные элементы управления (добавленные кнопки): ***Слева*** или ***Справа***.

Сбросить настройки

Когда свойство активно, пользователю будет доступна кнопка "Сбросить пользовательские настройки". Когда свойство неактивно, сбросить настройки на стандартные нельзя.

Экспорт

Разрешает или запрещает возможность экспортировать представление в Excel.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Узел для групповых операций

Встроенный элемент управления нового грида, содержит групповые операции нового грида.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Текст, если нет групповых операций

Сообщение, отображаемое для пользователей, если групповые операции недоступны.

Название папки



Данный элемент управления находится в разработке и может быть изменён в будущем.

Элемент управления отображает название папки, в контейнере которой он расположен. Внешний вид может определяться свойством Дополнительными CSS классами.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Параметры поискового запроса



После изменения разметок данной локации потребуется перезапуск службы Сервера Web-клиента, чтобы отобразить изменения.

Таблица параметров запроса

Элемент управления Таблица параметров запроса предназначен для настройки разметки диалога параметрического поискового запроса в локации **WebFrame › Диалог поисковых параметров**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Профиль пользователя



После изменения разметок данной локации потребуется перезапуск службы Сервера Web-клиента, чтобы отобразить изменения.

Информация о плагине КриптоПро

Элемент управления Информация о плагине КриптоПро предназначен для отображения статуса установки компонентов КриптоПро на компьютере пользователя, и ссылок для загрузки компонентов КриптоПро в разметке локации **WebFrame › Диалог профиля пользователя**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Информация о пользователе

Элемент управления Информация о пользователе предназначен для отображения информации о пользователе в разметке локации **WebFrame › Диалог профиля пользователя**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Информация об утилите DVWebTool



Программа *DVWebTool* устанавливается на клиентскую машину только в ОС Microsoft Windows. Для ОС Linux используется *DVSupService*, см. подробнее Установка и запуск службы DVSupService.

Элемент управления Информация об утилите DVWebTool предназначен для отображения ссылки на установщик программы *DVWebTool* в разметке локации **WebFrame › Диалог профиля пользователя**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка сброса настроек

Элемент управления Кнопка сброса настроек добавляет кнопку сброса пользовательских настроек в разметку локации **WebFrame › Диалог профиля пользователя**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента

управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Настройка сертификатов ЭП

Элемент управления отвечает за настройку сертификатов подписи в профиле пользователя.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Разблокировать карточки

Элемент управления Разблокировать карточки предназначен для отображения команды снятия блокировок со всех карточек, установленных данным пользователем, в разметке локации **WebFrame › Диалог профиля пользователя**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Расположение кнопок

Элемент управления Расположение кнопок предназначен для переключения расположения кнопок **ОК** / **Сохранить** и **Отмена** в диалоговых окнах и разметках. Элемент можно добавить в разметку локации **WebFrame** › **Диалог профиля пользователя**.

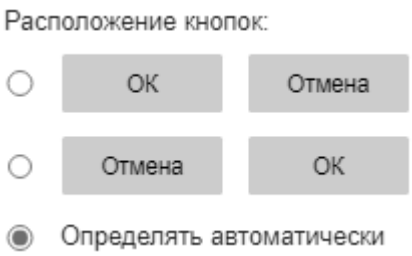


Рисунок 541. Пример элемента управления "Расположение кнопок"

Выбранное пользователем значение сохраняется в локальное хранилище. На разных устройствах может быть выбрано собственное значение настройки.

Пользователь может выбрать определённое значение **ОК** / **Отмена** или **Отмена** / **ОК**, или автоматический режим:

- в ОС MacOS — используется вариант **Отмена** / **ОК**;
- в ОС Windows,Linux, Android и пр. — используется вариант **ОК** / **Отмена**.

По умолчанию используется вариант расположения, указанный в настройке `DefaultButtonPosition` в конфигурационном файле Web-клиента. Предустановленный вариант — автоматический.

Выбранное пользователем значение сбрасывается при нажатии кнопки **Вернуть значение по умолчанию**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например,

JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Смена языка

Элемент управления `Смена языка` добавляет в раскрывающийся список для смены языка пользовательского интерфейса Web-клиента. Предназначен для добавления в разметку локации **WebFrame › Диалог профиля пользователя**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Номенклатура дел

Номенклатура дел 6.2

Управление архивом 6.2

См. описание элементов управления группы "Номенклатура дел" в документации модуля "Управление архивом": "Номенклатура дел".

- Список видов документов
- Справочник номенклатуры дел
- Дело
- Отображение документов

Справочник номенклатуры дел

ЭУ Справочник номенклатуры дел предназначен для работы администраторов. Справочник открывает возможности создания иерархической структуры разделов для создания дел и списания документов в дело при помощи ЭУ Дело, просмотра документов, списанных в дело, см. подробнее в документации модуля Управление архивом: "Справочник номенклатуры дел".

Дело



Используйте только совместимые версии модулей Управление архивом и Web-клиент. Версия 6.2 и несовместима с версией 5.5.

По вопросам совместимости Управление архивом версий 6.1 и 6.2 с Web-клиентом версии 6.1 и выше проконсультируйтесь со

специалистом Docsvision.

Элемент управления Дело предназначен для списания текущего документа в выбранное дело *Справочника номенклатуры дел*, см. подробнее в документации модуля Управление архивом: "Дело".

Отображение документов



Используйте только совместимые версии модулей Управление архивом и Web-клиент. Версия 6.2 и несовместима с версией 5.5.

По вопросам совместимости Управление архивом версий 6.1 и 6.2 с Web-клиентом версии 6.1 и выше проконсультируйтесь со специалистом Docsvision.

ЭУ выводит все документы, добавленные в дело из первого найденного тома данного тела. Том создаётся автоматически при добавлении в дело первого документа, см. подробнее в документации модуля Управление архивом: "Отображение документов".

Системные

Web-клиент Конструктор Web-разметок

Системные

Описание элементов управления группы "Системные".

- Автомат состояния
- Вид карточки
- Состояние
- Управление карточкой

Автомат состояния

Элемент управления Автомат состояния предназначен для смены состояния карточки. Возможность смены состояния определяется автоматом состояний, настроенным в *Конструкторе состояний* Docsvision, и ролевой моделью.

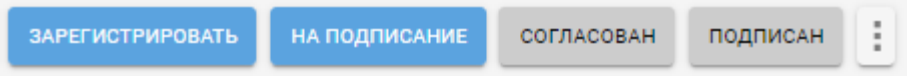


Рисунок 542. Пример элемента управления "Автомат состояний

Количество отображаемых кнопок может ограничено настройкой **Ограничение количества кнопок**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Вертикальная ориентация

Определяет формат отображения кнопок элемента управления.

- Когда флаг установлен, кнопки выстраиваются вертикально.
- Когда флаг не установлен, кнопки выстраиваются горизонтально.

Адаптивный

Данные

Операции

Выбор операций, из которых будет сформирован список отображаемых в элементе управления операций перехода.

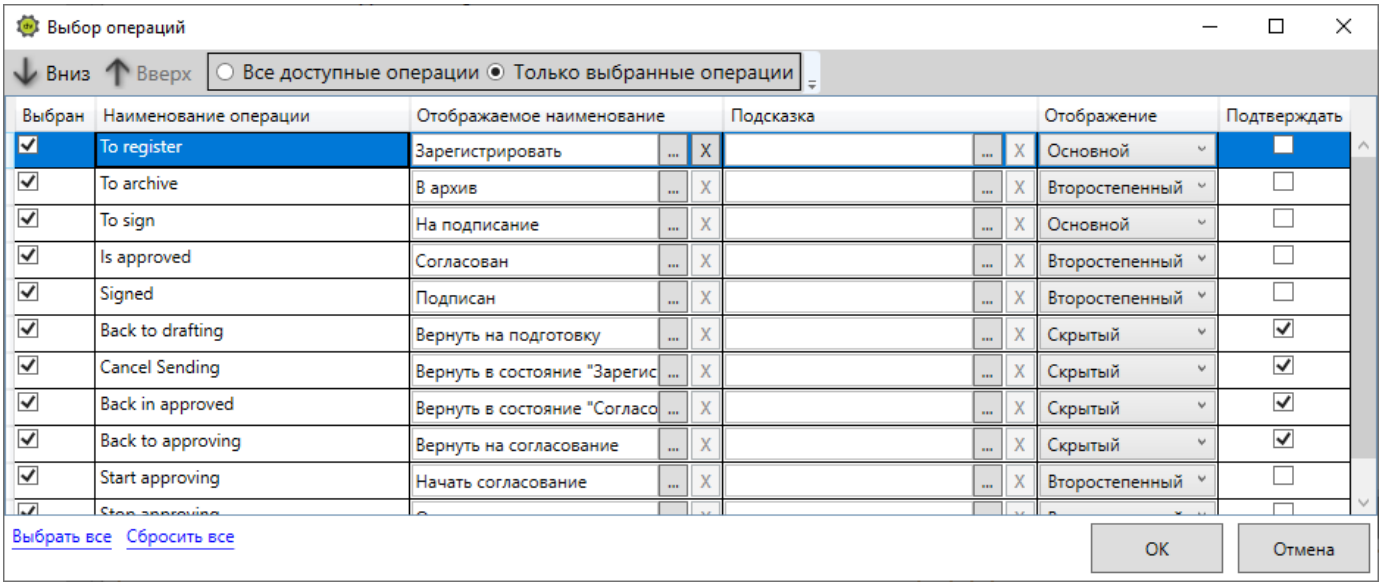


Рисунок 543. Окно "Выбор операций"

- Выбран. Включает или выключает показ данной операции в элементе управления. Когда флаг установлен, доступность операции пользователю будет зависеть от прав пользователя и текущего состояния карточки.
- *Отображаемое наименование* — название операции, с которым она будет отображаться в элементе управления. По умолчанию используются стандартные названия из *Конструктора состояний*.
- *Подсказка*. Текст, отображаемый при наведении курсора мыши на операцию.
- *Отображение*. Вариант отображения кнопки операции:
  - ***Основной***. Кнопка операции доступна по умолчанию. Цвет кнопки соответствует основному цвету типа текущей карточки.
  - ***Второстепенный***. Кнопка операции доступна по умолчанию; цвет кнопки — серый.
  - ***Скрытый***. Кнопка операции по умолчанию скрыта — для доступа пользователь должен нажать кнопку ⋮.
- Подтверждать. Включает/отключает подтверждение выполнения операции. Текст запроса может быть изменён на собственный с помощью функции, указанной в свойстве Функция генерации текста подтверждения.

  Переключатель **Все доступные операции** / **Только выбранные операции** включает выбор всех операций из списка или только операций с флагом Выбран.

  Изменить порядок отображения операций у пользователя можно с помощью

кнопок **Вверх** / **Вниз**.

Итоговый список операций перехода, отображаемых в элементе управления, определяется настройками автомата состояний в *Конструкторе состояний*, правами пользователя согласно ролевой модели и списком операций, выбранных в данной настройке.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Ограничение количества кнопок

Определяет максимальное количество одновременно отображаемых в ЭУ кнопок. Если число возможных (для карточки) операций перехода превышает установленное значение, последней будет добавлена кнопка ⋮. При нажатии этой кнопки будут показаны все кнопки перехода. Если ограничение отсутствует, отображаются все доступные операции.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Функция генерации текста подтверждения

Название клиентской функции, возвращающей текст подтверждения, отображаемый при выборе операции с установленным флагом `Подтверждать`.

Пример функции на TypeScript:

```
function createConfirmation(sender: StateButtons, operationDataModel: GenModels
.OperationDataModel){
return "Подтвердите выполнение операции " + operationDataModel.displayName;
}
```

События

Перед выполнением операции

Вызывается после подтверждения операции пользователем и настройки выполнения операции на боковой панели (если используется).

Перед изменением состояния карточки

Вызывается перед выполнением операции изменения состояния карточки.

Перед отображением диалога параметров

Вызывается после подтверждения операции пользователем, до отображения диалога настройки выполнения операции. Событие срабатывает только при наличии разметки настройки операции.

После изменения состояния карточки

Вызывается после выполнения операции изменения состояния карточки.

После нажатия на кнопку

Вызывается после нажатия на кнопку операции, до отображения окна подтверждения операции.

После отображения диалога параметров

Вызывается после показа диалога настройки выполнения операции. Событие срабатывает только при наличии разметки настройки операции.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Вид карточки

Элемент управления Вид карточки выводит название вида карточки.

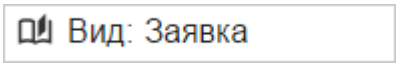


Рисунок 544. Пример элемента управления "Вид карточки"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Текст всплывающей подсказки. Когда значение не указано, во всплывающей подсказке отображается название вида с названиями родительских видов и типа.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Состояние

Элемент управления `Состояние` выводит локализованное название текущего состояния карточки.



Рисунок 545. Пример элемента управления "Состояние"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Управление карточкой

Элемент управления Управление карточкой предназначен для выполнения с карточкой стандартных действий:

- Добавить в избранные
- Открыть в режиме редактирования
- Обновить
- Удалить
- Открыть сообщения карточки



Рисунок 546. Элемент управления "Управление карточкой" в открытой карточке

Права на редактирование проверяются по доступности *операции редактирования*, выбранной в соответствующем свойстве элемента управления. Права на удаление проверяются по встроенной операции удаления. Значки недоступных операций не отображаются в элементе управления. Открытие сообщений карточек доступно только пользователям в группе **Администраторы**

Консоли управления.



Данный ЭУ может быть добавлен только в разметку, предназначенную для просмотра карточек.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Данные

Операция редактирования

Выбор операции для редактирования карточки. Если значение не задано, кнопка редактирования будет отображаться всем.

Если значение `Операции редактирования` наследуется от родительского Блока, название свойства меняется на `Операция редактирования (наследовано)`.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

`Адаптивный`

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Согласование

Web-клиент Конструктор Web-разметок

Согласование

Описание элементов управления группы "Согласование".

- Исполнение подчиненных заданий
- Лист согласования
- Отображение инициатора
- Другие элементы управления группы в навигационном меню слева…

Исполнение подчиненных заданий

После изменения разметок данной локации потребуется перезапуск службы Сервера Web-клиента, чтобы отобразить изменения.

Элемент управления Исполнение подчиненных заданий предназначен для копирования комментариев и файлов из подчиненных заданий в текущее задание **согласования**.

Элемент управления представляет собой кнопку, при нажатии которой открывается форма переноса комментариев. Кнопка отображается только при наличии подчиненных заданий.

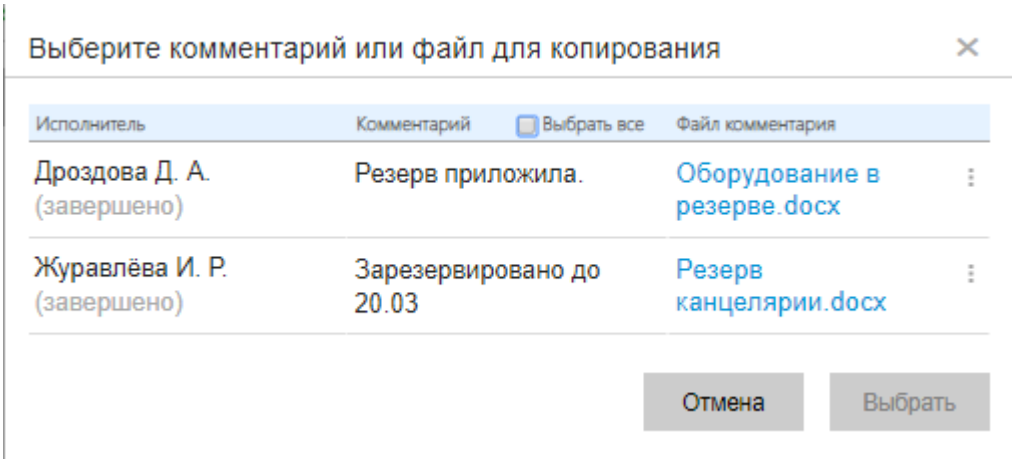


Рисунок 547. Форма переноса комментариев в элементе управления "Исполнение подчиненных заданий"

Элемент управления позволяет:

- Просматривать комментарии и файлы комментариев из подчиненных заданий.
- Переносить комментарии и файлы комментариев в текущее согласование.

Данный ЭУ может быть добавлен только в разметку просмотра карточки типа Задание.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст кнопки

Текст, отображаемый в кнопке. Если значение не установлено, будет использоваться значение по умолчанию: ***Скопировать из подчиненных заданий***.

Локализуемый

Данные комментария

Источник данных комментария

Секция карточки, содержащая данные элемента управления.

Поле комментария

Поле карточки, содержащее текст комментария (используется общая настройка для источника и получателя комментария).

Операция редактирования комментария

Операция редактирования, при доступности которой пользователю будет разрешено копировать комментарий.

Данные файла комментария

Источник данных файла комментария

Секция карточки, содержащая данные элемента управления.

Поле файла комментария

Поле карточки, содержащее файл комментария (используется общая настройка для источника и получателя комментария).

Операция редактирования файла комментария

Операция редактирования, при доступности которой пользователю будет разрешено копировать файл комментарий.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Уровень вложенности подчиненных

Определяет максимальный уровень вложенности заданий, комментарии к которым будут доступны для копирования.

События

Перед загрузкой данных подчиненных заданий

Вызывается перед загрузкой данных подчиненных заданий.

Перед закрытием диалога

Вызывается перед закрытием диалога переноса комментариев.

Перед изменением выделения

Вызывается перед изменением выбора комментариев.

Перед копированием комментариев в карточку

Вызывается перед копированием комментариев в карточку.

Перед открытием диалога

Вызывается перед открытием формы копирования комментариев.

Перед открытием превью файла

Вызывается перед открытием предварительного просмотра файла.

Перед открытием файла

Вызывается перед открытием файла комментария.

Перед перезагрузкой карточки

Вызывается перед перезагрузкой текущей разметки, выполняемой после копирования комментариев в карточку.

Перед скачиванием файла

Вызывается перед сохранением файла комментария на диск.

После загрузки данных подчиненных заданий

Вызывается после загрузки данных подчиненных заданий.

После закрытия диалога

Вызывается после закрытия диалога переноса комментариев.

После изменения выделения

Вызывается после изменения выбора комментариев.

После копирования комментариев в карточку

Вызывается после копирования комментариев в карточку.

После открытия диалога

Вызывается после открытия формы копирования комментариев.

После открытия превью файла

Вызывается после открытия предварительного просмотра файла.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Лист согласования

Элемент управления Лист согласования предназначен для открытия листа согласования.

Элемент управления добавляет в карточку кнопку, при нажатии которой открывается форма *Лист согласования*.

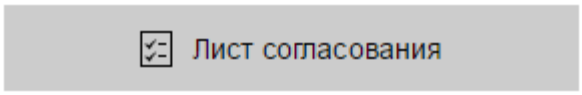


Рисунок 548. Пример отображения элемента управления "Лист согласования"

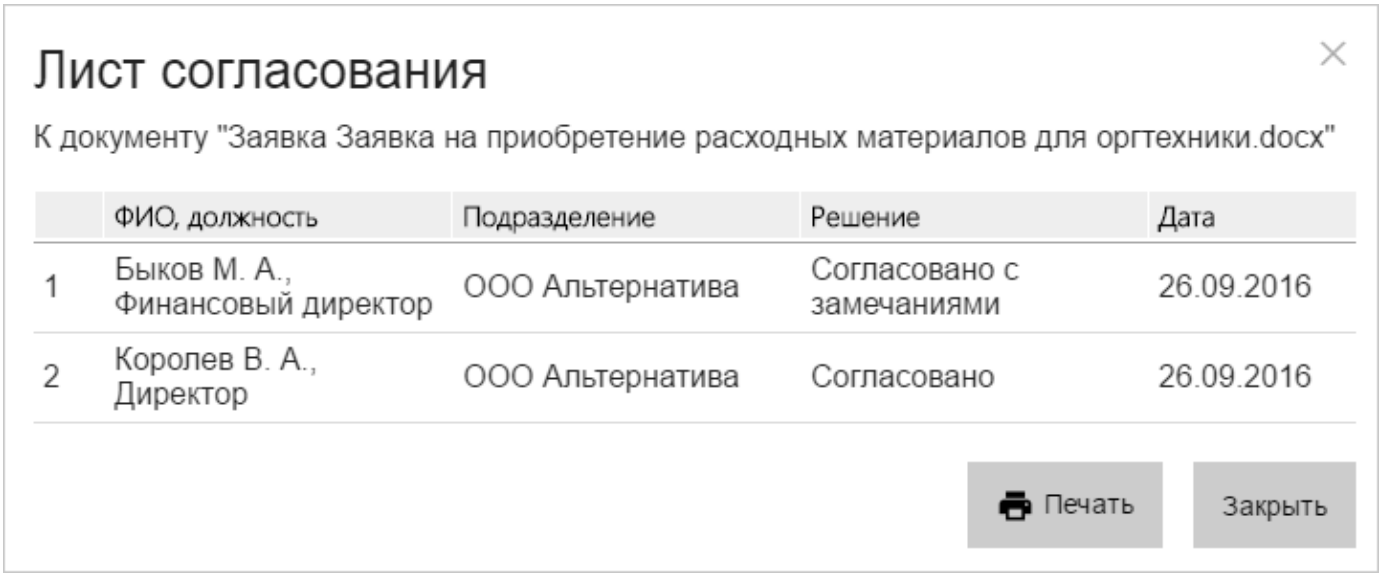


Рисунок 549. Пример открываемого листа согласования

На мобильных устройствах кнопка **Печать** не отображается, так как недоступна печать листа согласования.

Данный ЭУ может быть добавлен только в разметку карточек типа Документ.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст кнопки

Текст, отображаемый в кнопке. Если значение не установлено, отображается значение по умолчанию.

Локализуемый

Данные

Операция редактирования

Выбор операции, которая должна быть доступна пользователю для открытия листа согласования.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.

- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

После закрытия отчёта

Вызывается после закрытия окна листа согласования.

После открытия отчёта

Вызывается после отрытия окна листа согласования.

При закрытии отчёта

Вызывается при закрытии окна листа согласования.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При открытии отчёта

Вызывается при открытии окна листа согласования.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Отображение инициатора

Элемент управления Отображение инициатора предназначен для отображения инициатора согласования в карточке Задания. Не предназначен для изменения значения.

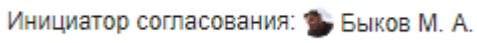


Рисунок 550. Пример отображения элемента управления "Отображение инициатора"

> В ЭУ отображаются данные поля Initiator из связанной карточки Согласование.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:

- Когда флаг установлен, текст метки отображается всегда.
- Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Подсказка к значению

Содержимое всплывающей подсказки, отображаемой если сотрудник выбран. Возможные варианты:

- ***ФИО***.
- ***ФИО+Должность***.
- ***Не используется***. Во всплывающей подсказке отображается текст из поля Подсказка

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Управление согласованием

Элемент управления Управление согласованием предназначен для отправки документа на согласование и управления ходом активного согласования.

Элемент управления добавляет в карточку документа кнопку На согласование / Управление (название может быть настроено), при нажатии которой открывается боковая панель отправки или управления согласованием. На панели отображаются маршруты, которые были настроены с использованием средств модуля *Конструктор согласований*.

Содержимое панели определяется текущим состоянием согласования:

- При создании согласования панель отображает настройки маршрута, а также кнопки отправки и отмены отправки согласования, временного исключения ☐

или удаления согласующего :

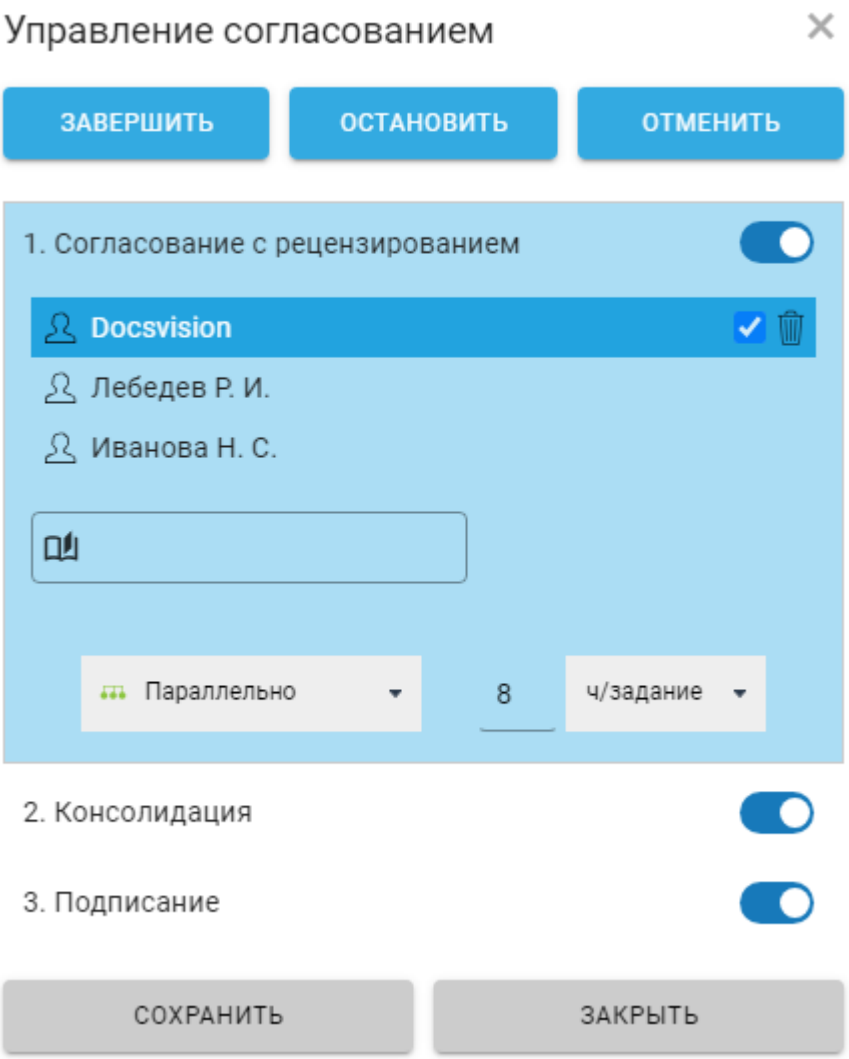


Рисунок 551. Панель согласования при создании согласования

- При управлении согласованием, когда есть активное согласование, панель отображает настройки этапа согласования, а также кнопки управления согласованием:

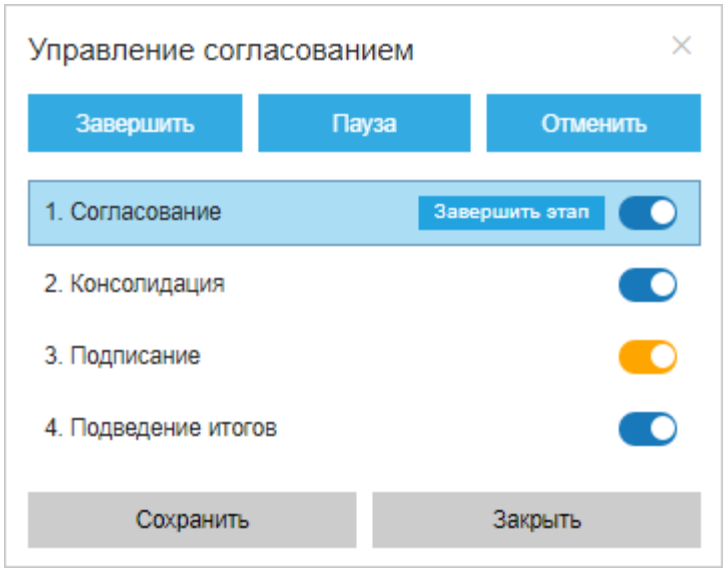


Рисунок 552. Панель согласования при наличии активного согласования

Фактический набор команд и настроек панели согласования, которые будут доступны пользователю, зависит от следующих факторов:

Факторы, влияющие на набор команд:

- Текущее состояние согласования.
- Доступность операции пользователю согласно автомату состояний и ролевой модели.
- Настройки маршрута согласования.
- Настройки элемента управления.
- Настройки этапа согласования:
  - Чтобы иметь возможность исключить этап из маршрута, необходимо установить флаг **Разрешить исключение этапа из маршрута** в настройках этапа.
  - Чтобы иметь возможность изменить длительность этапа и список согласующих, необходимо установить флаг **Разрешить изменение параметров этапа** в настройках этапа.
  - Чтобы иметь возможность изменить тип маршрута, необходимо установить флаги **Разрешить изменение параметров этапа** и **Разрешить редактирование типа маршрутизации** в настройках этапа.



Данный ЭУ может быть добавлен только в разметку, предназначенную для просмотра карточек типа Документ.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Названия кнопок

Настройка названий кнопок управления согласованием.

Локализованные названия кнопок настраиваются в форме, открываемой при нажатии кнопки ...:

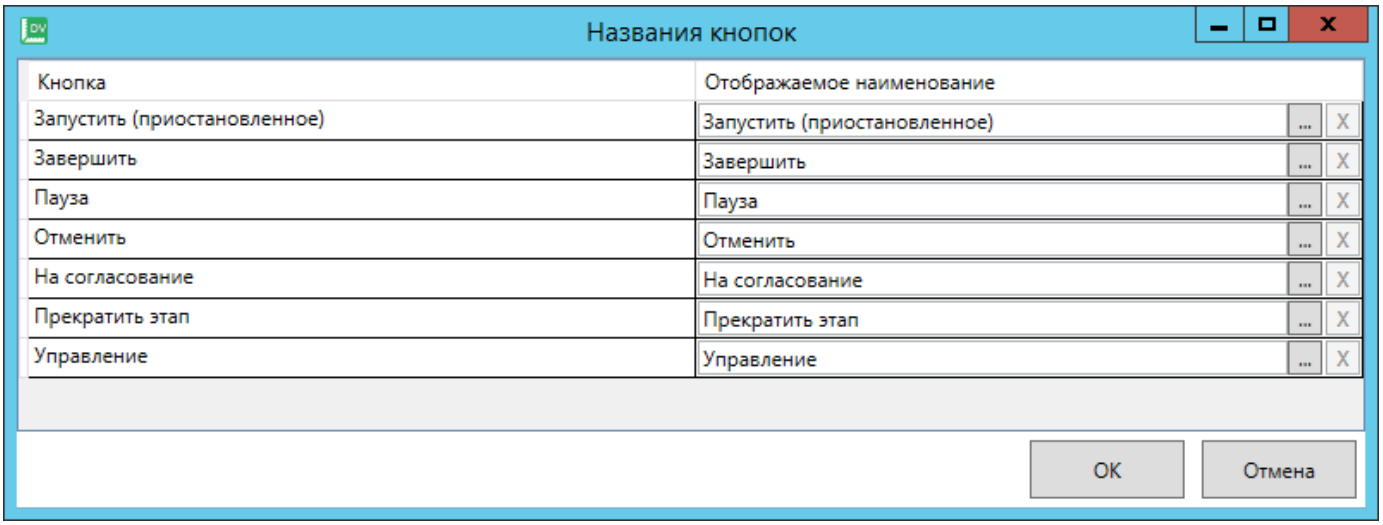


Рисунок 553. Форма настройки названий кнопок ЭУ "Управление согласованием"

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Запрос подтверждения при старте

Настройка подтверждения отправки на согласование:

- Когда флаг установлен, при отправке на согласование отображается запрос на подтверждение отправки.
- Когда флаг не установлен, согласование запускается без дополнительного подтверждения.

Использовать последних

Если флаг установлен, при добавлении участников согласования отображаются последние выбранные исполнители.

Если область выбора ограничена группой, свойство Использовать последних не работает.

Операция для прекращения этапа

Выбор операции, которая должна быть доступна пользователю для прекращения этапа согласования.

Операция для старта согласования

Выбор операции, которая должна быть доступна пользователю для запуска согласования.

Операция для управления согласованием

Выбор операции, которая должна быть доступна пользователю для управления согласованием.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности

пользователю выбранной операции редактирования.

- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим

Определяет режим отображения кнопок старта и управления согласованием. Возможные варианты:

- ***Старт и Управление***. Когда согласование запущено, отображаются кнопки управления согласованием. Когда согласование не запущено, отображается кнопка запуска согласования.
- ***Только Старт***. Отображается только кнопка запуска согласования, если оно не запущено.
- ***Только Управление***. Отображаются только кнопки управления согласованием, если оно запущено.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Управление согласованием

Открывает форму настройки списка отображаемых в ЭУ кнопок управления согласованием: **Остановить**, **Отменить** и **Завершить**.

В панели *Управление согласованием* отображается состав согласующих,

зафиксированный на момент старта этапа согласования. Если после запуска этапа фактические исполнители заданий изменились, например из-за делегирования, добавления согласующих или специальной логики, состав в окне автоматически не обновляется.

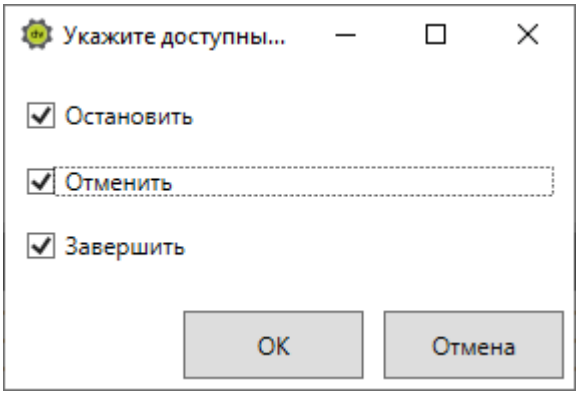


Рисунок 554. Форма настройки доступности кнопок управления согласованием ЭУ "Управление согласованием"

Кнопки управления, с которых снят флаг, будут скрыты с панели управления согласованием.

Кнопка **Управление согласованием** недоступна, если свойство Режим в значении ***только Старт***.

Формат отображения Согласующего

Определяет формат отображения имён согласующих на панели согласования:

- ***ФИО***.
- ***ФИО+Должность***.
- ***Строка отображения*** — используется формат отображения, настроенный в *Справочнике сотрудников*.

События

Перед сохранением согласования

Вызывается перед сохранением согласования.

После сохранения согласования

Вызывается после сохранения согласования.

Перед добавлением участника согласования

Вызывается перед добавлением участника согласования.

Перед закрытием панели согласования

Вызывается перед закрытием панели согласования.

Перед изменением пути согласования

Вызывается перед изменением пути согласования.

Перед остановкой согласования

Вызывается перед остановкой согласования.

Перед остановкой Этапа

Вызывается перед остановкой этапа согласования.

Перед открытием панели согласования

Вызывается перед открытием панели согласования.

Перед отменой согласования

Вызывается перед отменой согласования.

Перед отменой старта согласования (закрытием панели)

Вызывается перед отменой старта согласования, перед закрытием панели согласования.

Перед принудительным завершением согласования

Вызывается перед принудительным завершением согласования.

Перед продолжением согласования после паузы

Вызывается перед продолжением согласования после паузы.

Перед стартом согласования

Вызывается перед стартом согласования.

Перед удалением участника согласования

Вызывается перед удалением участника согласования.

После добавления участника согласования

Вызывается после добавления участника согласования.

После закрытия панели согласования

Вызывается после закрытия панели согласования.

После изменения пути согласования

Вызывается после изменения пути согласования.

После остановки Этапа

Вызывается после остановки этапа согласования.

После открытия панели согласования

Вызывается после открытия панели согласования.

После отмены старта согласования (закрытия панели)

Вызывается после отмены старта согласования, после закрытия панели согласования.

После удаления участника согласования

Вызывается после удаления участника согласования.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Файл замечаний

Элемент управления Файл замечаний предназначен для добавления файла с замечаниями к согласованию.

ЭУ представляет собой поле со ссылкой на файл замечаний и кнопкой ⋮ для открытия меню.

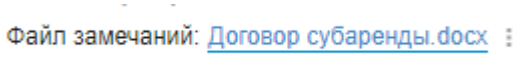


Рисунок 555. Элемент управления "Файл замечаний"

Меню, в зависимости от роли пользователя и режима работы элемента

управления, может содержать команды для открытия файла с замечаниями, загрузки файла на компьютер, замены и удаления.



Данный ЭУ может быть добавлен только в разметку, предназначенную для просмотра карточек.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать **Заполнитель**.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, имеющее тип `FileId`.

Операция редактирования

Выбор операции для редактирования значения ЭУ. Если операция недоступна пользователю, то выбор и изменение ссылки будут недоступны. Если операция не выбрана, то возможность изменения ссылки не проверяется.

Если значение `Операции редактирования` наследуется от родительского Блока, название свойства меняется на `Операция редактирования (наследовано)`.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента

управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется в отдельном окне, которое открывается нажатием на элемент управления. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра

карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления. Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед открытием диалога выбора файла

Вызывается перед открытием диалога выбора файла замечаний.

Перед открытием превью

Вызывается перед открытием предварительного просмотра файла.

Перед открытием файла через WebDAV

Вызывается перед открытием файла с использование WebDAV.

При сохранении файла в WebDAV

Вызывается при сохранении файла с использованием WebDAV.

Перед скачиванием файла

Вызывается перед скачиванием файла.

Перед удалением файла

Вызывается перед удалением файла.

После закрытия диалога выбора файла

Событие, возникающее после открытия окна выбора файла, но до загрузки файла на сервер и установления значения элемента управления.

После открытия превью

Вызывается после открытия предварительного просмотра файла.

После удаления файла

Вызывается после удаления файла.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Файловая панель согласования

Элемент управления Файловая панель согласования предназначен для работы с файлами согласования.

Одновременное размещение на разметке элементов управления "Файловая панель" и "Файловая панель согласования" в данный момент не поддерживается и может привести к ошибкам при открытии файлов на предпросмотр.

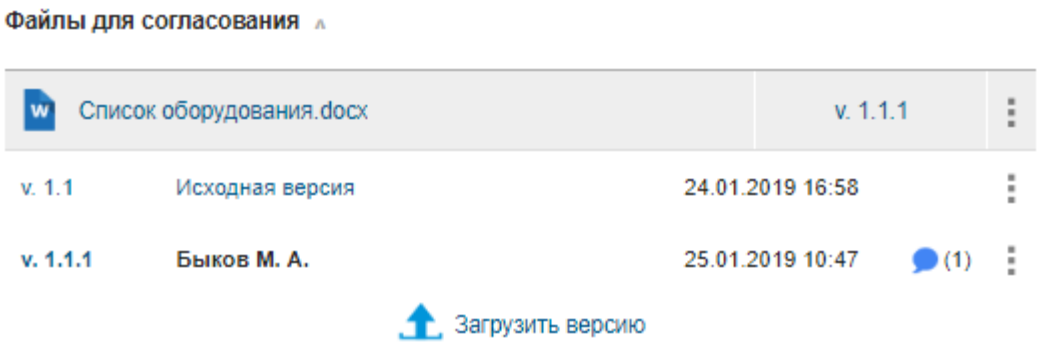


Рисунок 556. Элемент управления "Файловая панель согласования"

ЭУ предоставляет возможность:

- Работать с версиями файлов в режимах согласования и консолидации.
- Просматривать содержимое файла в окне предварительного просмотра: собственная реализация или с использованием компонента Предпросмотр файла.
- Комментировать версии файлов.

> Данный ЭУ может быть добавлен только в разметку карточек Задание.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Загружать синхронно (шт)

Определяет максимальное количество элементов списка, при котором должна использоваться синхронная загрузка данных. Если количество элементов больше указанного, то все данные будут загружаться асинхронно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

При сохранении файла в WebDAV

Вызывается при сохранении файла с использованием WebDAV.

Ход согласования

Элемент управления `Ход согласования` предоставляет возможность просматривать журнал согласования документа. Доступность элемента для пользователя зависит от настроек разметки и прав на просмотр согласования.

ЭУ поддерживает два режима работы:

1. В карточке отображается кнопка, при нажатии которой будет открыт журнал согласования.
2. В карточке сразу отображается журнал согласования:

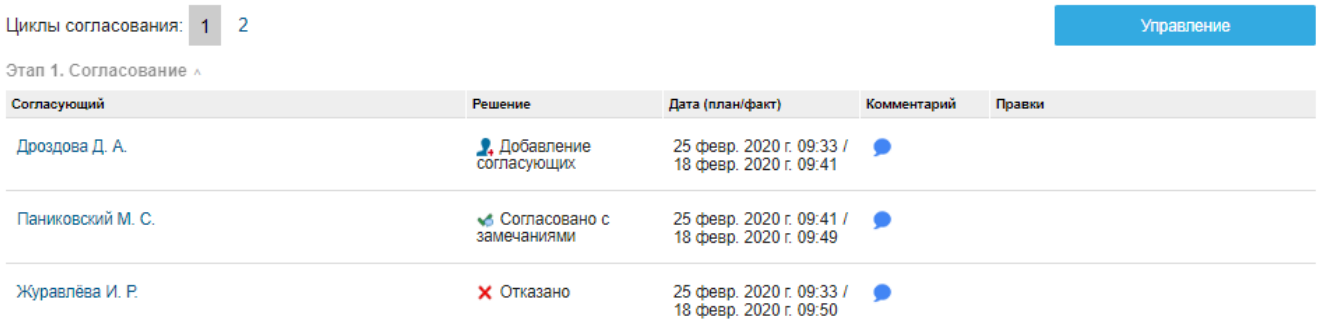


Рисунок 557. Пример журнала хода согласования

Может быть настроен способ отображения комментариев к решению по согласованию: отображать непосредственно в таблице или в отдельном окне (нажать на значок 💬).

Доступна возможность программного переопределения метода получения содержимого ячеек таблицы *Ход согласования* (см. Переопределение сервиса отрисовки элемента "Ход согласования").

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Комментарий в таблице

Определяет способ отображения комментариев к решению по согласованию в элементе управления:

- Когда флаг установлен, комментарий отображается непосредственно в таблице хода согласования, файл с замечанием доступен по отдельной ссылке.
- Когда флаг снят, в столбце *Комментарий к решению* отображается значок для открытия комментарий в отдельном окне.

Адаптивный

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст кнопки

Текст, отображаемый в кнопке. Если значение не установлено, отображается только значок.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим

Определяет вариант отображения журнала согласования:

1. Кнопка — в карточке отображается кнопка, при нажатии которой будет открыт журнал согласования.
2. Разметка — в карточке отображается журнал согласования.

Режим открытия

Выбор способа открытия ссылки на задание согласования из таблицы "Ход согласования": "Текущая вкладка", "Новая вкладка", "Новое окно браузера".

Режим отображения

Выбор способа отображения хода согласования. По умолчанию используется ***Стандартный*** способ, при необходимости можно выбрать ***Полный*** способ отображения. При ***Полном*** отображении согласования будут отображены все согласования карточки, включая отмененные.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента

управления.

События

Перед сменой выбора этапа

Вызывается перед сменой выбора этапа.

После закрытия отчёта

Вызывается после закрытия окна журнала хода согласования.

После обновления отчёта

Вызывается после обновления журнала.

После открытия отчёта

Вызывается после открытия окна журнала хода согласования.

После смены выбора этапа

Вызывается после смены выбора этапа.

При закрытии отчёта

Вызывается при закрытии окна журнала хода согласования.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При обновлении отчёта

Вызывается при обновлении журнала.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При открытии отчёта

Вызывается при открытии окна журнала хода согласования.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Цикл согласования

Элемент управления Цикл согласования предназначен для отображения текущего

цикла согласования.

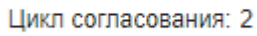


Рисунок 558. Элемент управления "Цикл согласования"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента

управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Этап согласования

Элемент управления Этап согласования предназначен для отображения текущего этапа согласования.

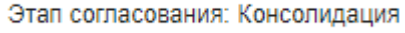


Рисунок 559. Элемент управления "Этап согласования"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности

пользователю выбранной операции редактирования.

- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Прочие

Web-клиент Конструктор Web-разметок

Прочие

Описание элементов управления группы "Прочие".

- Диаграмма

Диаграмма

Элемент управления добавляет отображение диаграмм на главной странице. Позволяет изменять вид и содержание диаграмм.

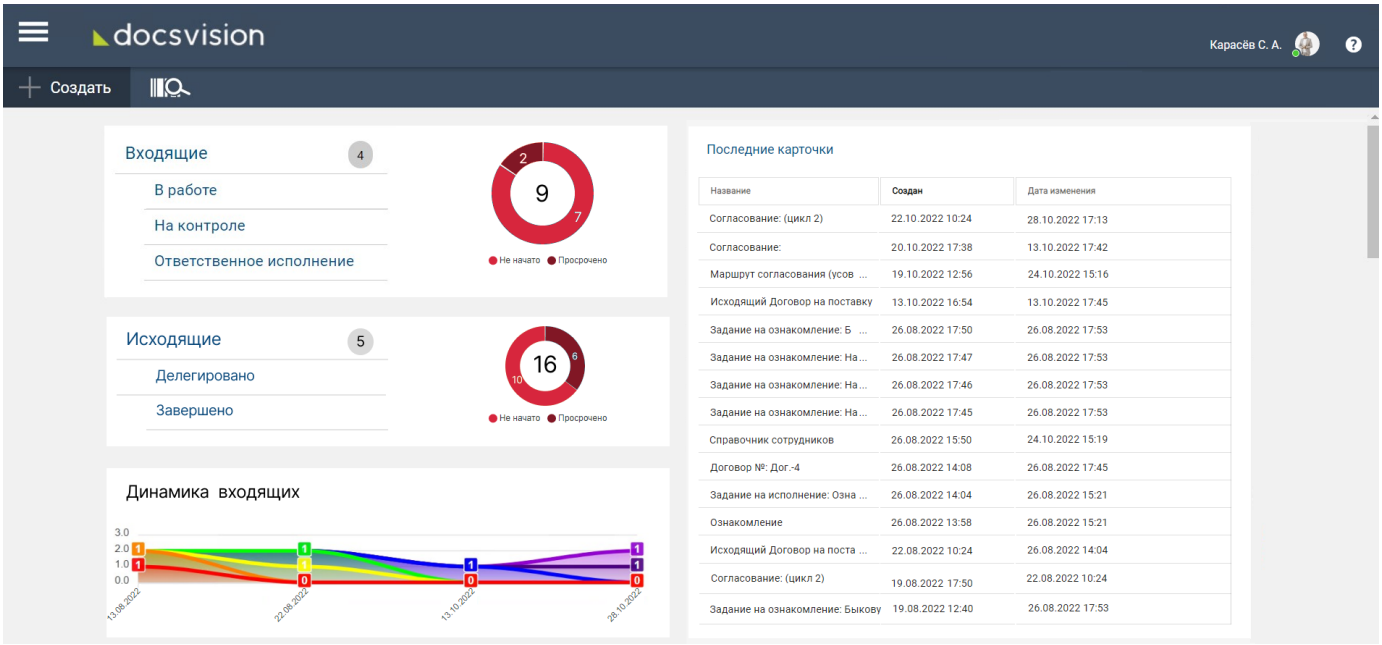


Рисунок 560. Элемент управления "Диаграмма" на главной странице

- При изменении свойств элемента управления рекомендуется очистка кэша браузера.
- ЭУ не работает в браузере Internet Explorer.
- ЭУ не предназначен для работы с большим количеством карточек.
- ЭУ не работает с представлением *Дайджест*.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Высота

Высота блока с диаграммой. Значение указывается в пикселях.

Чтобы высота устанавливалась автоматически, укажите значение `0`.

Заголовок

Текст, отображаемый над диаграммой.

Локализуемый

Показывать легенду

Свойство включает или отключает отображение легенды диаграммы.

Положение легенды

Позволяет выбрать местоположение легенды. Свойство доступно только, если флаг `Показывать легенду` установлен.

Доступные значения:

- Снизу
- Сверху
- Слева
- Справа

Цветовая схема

Определяет преобладающий цвет диаграммы. Для выбора доступны 9 цветов и разноцветная схема.

Цветовой режим

Определяет цветовой режим диаграммы: ***Одноцветный*** — будет использоваться один цвет или ***Монохромный*** — будут использоваться оттенки выбранного цвета.

Поведение

1-й столбец

Свойство определяет первый столбец из списка доступных в представлении для указанной папки или указанного представления.

2-й столбец

Свойство определяет второй столбец из списка доступных в представлении для указанной папки или указанного представления.

Свойство может быть не указано для других видов диаграмм.

Свойство не доступно для типа диаграмм: гистограмма и кольцевая диаграмма.

ID папки

Позволяет задать папку, из которой будут получены данные для диаграммы.

Свойство не доступно, если указан Поисковый запрос. Чтобы указать ID папки нужно очистить Поисковый запрос — указать значение ***<Не задано>***.

ID представления

Позволяет указать представление, которое будет служить источником данных для диаграммы.

Свойство доступно вне зависимости от других установленных свойств.

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.

- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Перейти в папку

Когда флаг установлен, нажатие на сектор диаграммы переводит в папку, указанную в свойстве ID папки. В открывшейся папке автоматически применяется фильтр, соответствующий выбранному сектору диаграммы.

Поисковый запрос

Позволяет выбрать один из предварительно настроенных поисковых запросов.

Свойство не доступно, если в ID папки указана папка. Чтобы снова выбрать ID папки, очистите поисковый запрос — выберите значение ***<Не задано>***.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Тип диаграммы

Выберите из раскрывающегося списка тип диаграммы, которая должна отображаться.

На выбор доступны 4 типа диаграмм:

- ***Гистограмма***.
- ***Гистограмма с накоплением***.
- ***Диаграмма с областями с накоплением***.
- ***Кольцевая***.

События

После загрузки данных

Вызывается после загрузки данных.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Специальные

Web-клиент Конструктор Web-разметок

Специальные

Описание элементов управления группы "Специальные".

- Html виджет
- Автоконсолидация
- Адрес
- Другие элементы управления группы в навигационном меню слева…

Html виджет

Элемент управления `Html виджет` предназначен для отображения HTML-страницы с заданным адресом.

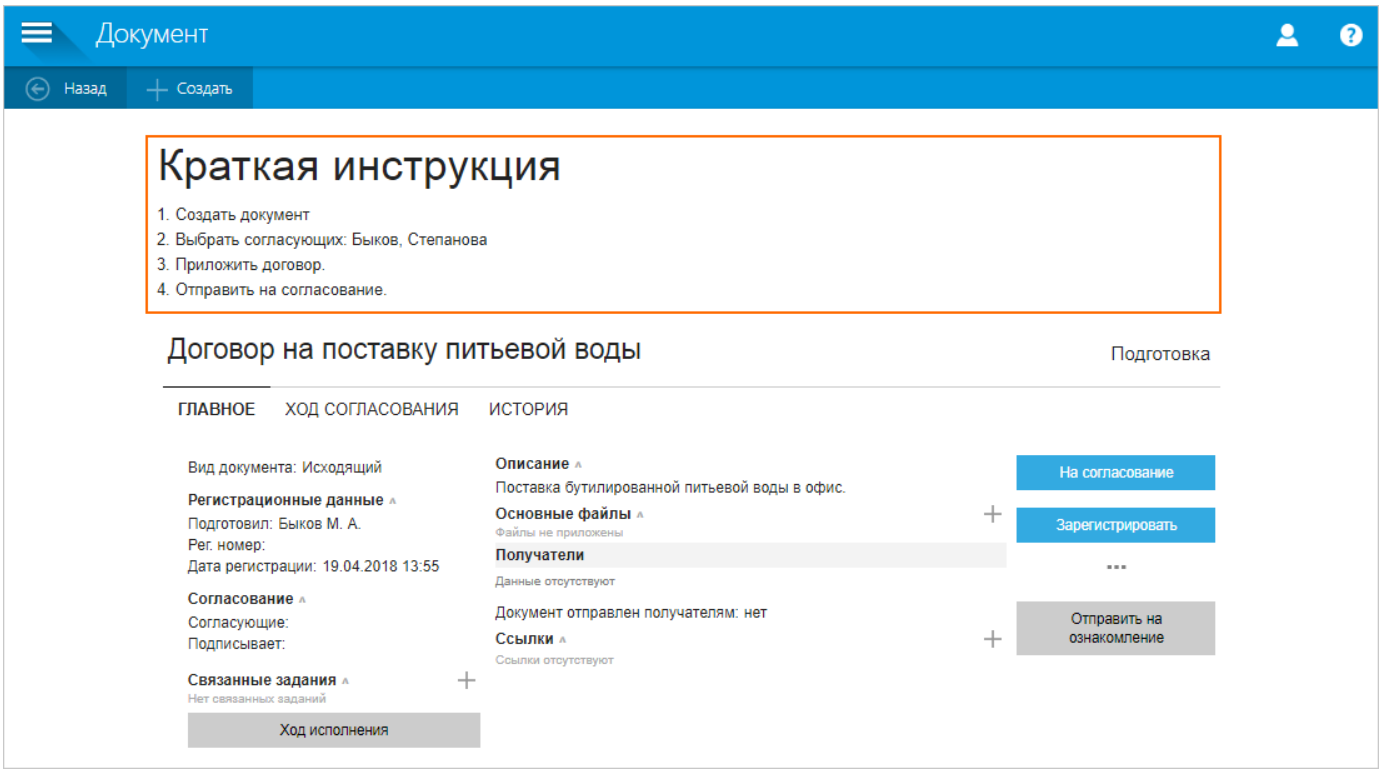


Рисунок 561. Элемент управления "Html виджет" с содержимым HTML-страницы в разметке исходящего документа

`Html виджет` может загружать HTML-страницу из указанного файла или ресурса. Дла загрузки HTML-страницы нужно указать значение свойства **Режим**. URL адрес можно указать непосредственно в настройках ЭУ или получить из данных карточки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Текстовое поле карточки, содержащее код отображаемой HTML-страницы.

Свойства `Поле данных` и `Источник данных` доступны, когда в настройке Режим выбрано значение `Из карточки`.

URL

Адрес HTML-страницы, который зависит от выбранного режима ЭУ (настройка `Режим`):

- В режиме ***Из файла*** следует указывать относительный путь к файлу HTML-страницы в каталоге установки модуля Web-клиент, например: `Content\Extensions\HtmlPages\manual.html`.

  > ⚠ Файл HTML-страницы следует размещать только в подкаталоге `Content/Extensions` или `Content/Solutions` каталога установки Web-клиента.

- В режиме ***Внешний ресурс*** следует указывать URL-адрес HTML-страницы, например: `http://web.domain.com/htmlpages/manual.html`.

  > ⚠ Внешний ресурс, с которого загружается HTML-страница, должен разрешать встраивание страниц в IFRAME (не отправлять заголовок `X-FRAME-OPTIONS` в ответе).

  Настройка "URL" отображается, когда выбран режим ***Из файла*** или ***Внешний ресурс***.

Поведение

Sandbox фрейма

Значение атрибута sandbox, добавляемого к тегу IFRAME. Должен быть установлен флаг Использовать sandbox фрейма.



Свойство Sandbox фрейма отображается, когда выбран режим Внешний ресурс.

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Обезопасить HTML

Флаг, указывающий на необходимость очистки HTML-кода внедряемой страницы от потенциально небезопасных элементов, через которые возможна XSS-атака (в частности, скрипты).

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Использовать sandbox фрейма

Флаг, указывающий на необходимость добавления к тегу IFRAME атрибута sandbox (см. developer.mozilla.org) со значением из свойства Sandbox фрейма.



Свойство Использовать sandbox фрейма отображается, когда выбран режим ***Внешний ресурс***.

Прокрутка фрейма

Режим показа полос прокрутки для содержимого, отображаемого во фрейме:

auto	Необходимость отображения полос прокрутки определяется автоматически.
yes	Полосы прокрутки отображаются всегда.
no	Полосы прокрутки всегда скрыты.



Свойство Прокрутка фрейма отображается, когда выбран режим ***Внешний ресурс***.

Режим

Определяет способ получения данных для отображения в виджете:

- ***Из карточки***. Должна отображаться HTML-страница, код которой хранится в поле карточки, указанном в настройке Поле данных.
- ***Из файла***. Должна отображаться HTML-страница из файла, путь к которому указан в настройке URL.
- ***Внешний ресурс***. Должна отображаться HTML-страница, адрес которой указан в настройке URL.

  В режиме ***Внешний ресурс*** HTML-страницы будут открываться во фрейме. В режимах ***Из файла*** и ***Из карточки*** HTML-страницы будут встраиваться в HTML-код страницы Web-клиента. Особенности размещения страницы во фрейме и встраивании в HTML-код приведены в конце раздела.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Размеры

Высота

Высота фрейма.

Ширина

Ширина фрейма.

Значения высоты и ширины фрейма указываются с единицей измерения (например, `100px`).



Свойства **Высота** и **Ширина** отображаются, когда выбран режим **Внешний ресурс**.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Отличительные особенности внедрения HTML-страницы от открытия страницы во фрейме

Если для отображения HTML-страницы используется фрейм (выбран режим ***Внешний ресурс***), страница отображается в отдельном окне браузера, размещенном внутри страницы Web-клиента, при этом:

- Стили, используемые Web-клиентом, не распространяются на содержимое фрейма.
- HTML-страница загружается асинхронно с загрузкой страницы Web-клиента.

Если HTML-страница внедряется (без использования фрейма), HTML-код страницы добавляется к HTML-коду страницы Web-клиента без преобразований, при этом:

- В коде внедряемой страницы не должно быть тегов `head`, `body` и `style`.
- Внедряемый код должен быть безопасным, т.к. данный код вставляется без изменений — иначе используйте фрейм, с установленным флагом **Использовать sandbox фрейма**.

Особенности использования изображений и стилей

Если для отображения HTML-страницы используется фрейм (режим **Внешний**

ресурс):

- Путь к ресурсам (изображение, скрипт) указывается относительно файла HTML-страницы, например:

```
<img src="images/pic.jpg" />
```

- Стиль подключается стандартным образом — в теге body.

Если для отображения HTML-страницы используется внедрение:

- Путь к ресурсам указывается относительно корня сайта, например:

```
<img src="Content\Extensions\HtmlPages\imgages\pic.jpg" />
```

- CSS файл со стилями размещается в папке Site\Content\Solutions\<Название решения>\Styles или Site\Content\Extensions\<Название расширения>\Styles каталога установки Web-клиента.

Автоконсолидация

Элемент управления Автоконсолидация предназначен для запуска функции автоматической консолидации версий согласуемого документа. Автоконсолидация выполняется с помощью "Р7-Офис. Сервер документов".

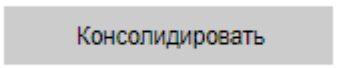


Рисунок 562. Пример отображения элемента управления "Автоконсолидация"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например,

JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на кнопке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
}
```

Растягивать на всю ширину

Элемент управления растягивается на всю ширину родительского элемента, когда флаг установлен. Ширина элемента управления соответствует содержимому, когда флаг снят.

Текст

Текст, отображаемый на кнопке.

Локализуемый

Данные

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Адрес

Элемент управления Адрес предназначен для отображения и редактирования секции, содержащей поля с данными адреса: индекс, город, адрес и страна.

Значение элемента управления отображается в специальном формате, соответствующем формату отображения адресных данных.



Рисунок 563. Элемент управления "Адрес"

Изменения значения элемента управления выполняется в отдельном окне с отдельными полями ввода индекса, города, адреса и страны.



Рисунок 564. Редактирование значения в элементе управления "Адрес"

Адрес является специализированным элементом управления для редактирования адреса в справочниках (сотрудника, контрагентов).

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Источник данных

Открывает окно выбора секции и полей карточки с данными элемента управления.

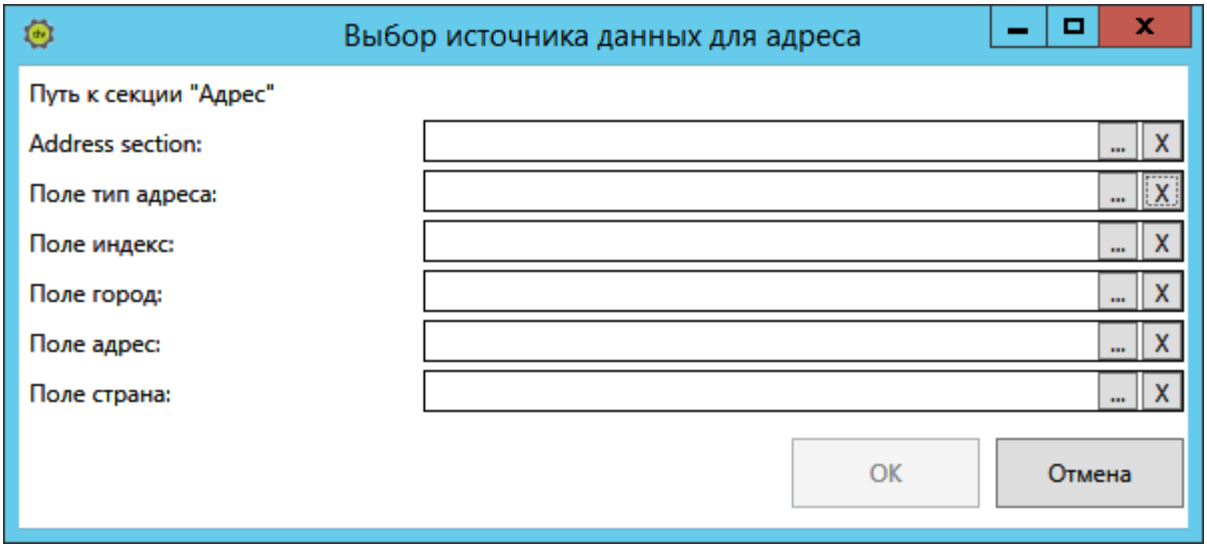


Рисунок 565. Окно с настройками источника данных

Последовательно выберите:

1. Секцию, содержащую поля с составляющими адреса.
2. Поле с типом адреса (контактный, юридический или почтовый).
3. Поле с индексом.
4. Поле с названием города.
5. Поле с адресом.
6. Поле с названием страны.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту*** — значение изменяется в отдельном окне, которое открывается нажатием на элемент управления. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.
- ***Редактирование*** — значение изменяется непосредственно в элементе управления. Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования*** — значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Тип адреса

Определяет тип адреса, устанавливаемого в элементе управления: Контактный, Почтовый или Юридический.

События

Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования ***По месту***.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования ***По месту***.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Дерево исполнения

Элемент управления Дерево исполнения предназначен для отображения дерева заданий.

ЭУ поддерживает два варианта отображения:

1. В карточке отображается кнопка, при нажатии которой будет открыто окно просмотра дерева исполнения.
2. В карточке сразу отображается дерево исполнения.



Рисунок 566. Элемент управления "Дерево исполнения" в карточке документа

При выборе задания или группы заданий будет открыт блок, содержащий информацию о данном элементе.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Режим отображения

Определяет вариант отображения ЭУ в карточке:

- ***Кнопка*** — в карточке отображается кнопка для открытия модального окна с деревом исполнения;
- ***Разметка*** — в карточке отображается непосредственно дерево исполнения.

Текст кнопки

Текст на кнопке открытия дерева исполнения, когда режим отображения в значении ***Кнопка***.

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее ссылку на карточку *Список ссылок на карточки заданий*.

Виды заданий для отображения

Выбор видов заданий, отображаемых в дереве исполнения. Если значение не указано, то в дереве исполнения отображаются задания и группы заданий всех видов.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Максимальное количество заданий Группы

Определяет количество заданий группы заданий, начиная с которого вся ветвь дерева исполнения с данной группой заданий будет отображаться свернутой. Свернутую под действием данного условия ветвь группы заданий можно раскрыть двойным кликом мыши или нажатием на кнопку **+**, отображаемую под названием свёрнутого узла. Если указано значение `0`, ветви групп заданий не будут сворачиваться.



Рисунок 567. Отображение свернутой группы заданий в дереве исполнения

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства `Видимость`:

- Когда флаг `Видимость` установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг `Видимость` установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг `Видимость` НЕ установлен, ЭУ всегда скрыт.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Просмотр полного дерева

Выбор операции, которая должна быть доступна пользователю для возможности просмотра полного дерева заданий.

По умолчанию в дереве исполнения пользователь видит только текущее задание и его подчиненные задания. Чтобы просмотреть дерево исполнения вместе с родительскими заданиями, нужно переключить режим отображения. Кнопка переключения режима отображения скрыта, если пользователю недоступна операция, указанная в свойстве **Просмотр полного дерева**. Если операция не указана в свойстве, кнопка переключения режима отображения будет отображаться всегда.

Режим отображения группы из 1 задания

Определяет вариант отображения в дереве исполнения группы заданий, содержащей единственное задание:

- ***Отображать оба*** — в дереве исполнения будут отображаться узлы группы заданий и задания.
- ***Группа*** — в дереве исполнения будет отображаться только узел группы заданий, при двойном клике мыши по которому будет выполнен переход к карточке данной группы.
- ***Задание*** — в дереве исполнения будет отображаться (если сформировано) только узел задания группы заданий, при двойном клике мыши по которому будет выполнен переход к карточке данного задания.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Задания

Элемент управления *Задания* предназначен для создания и отображения связанных заданий и групп заданий карточки.

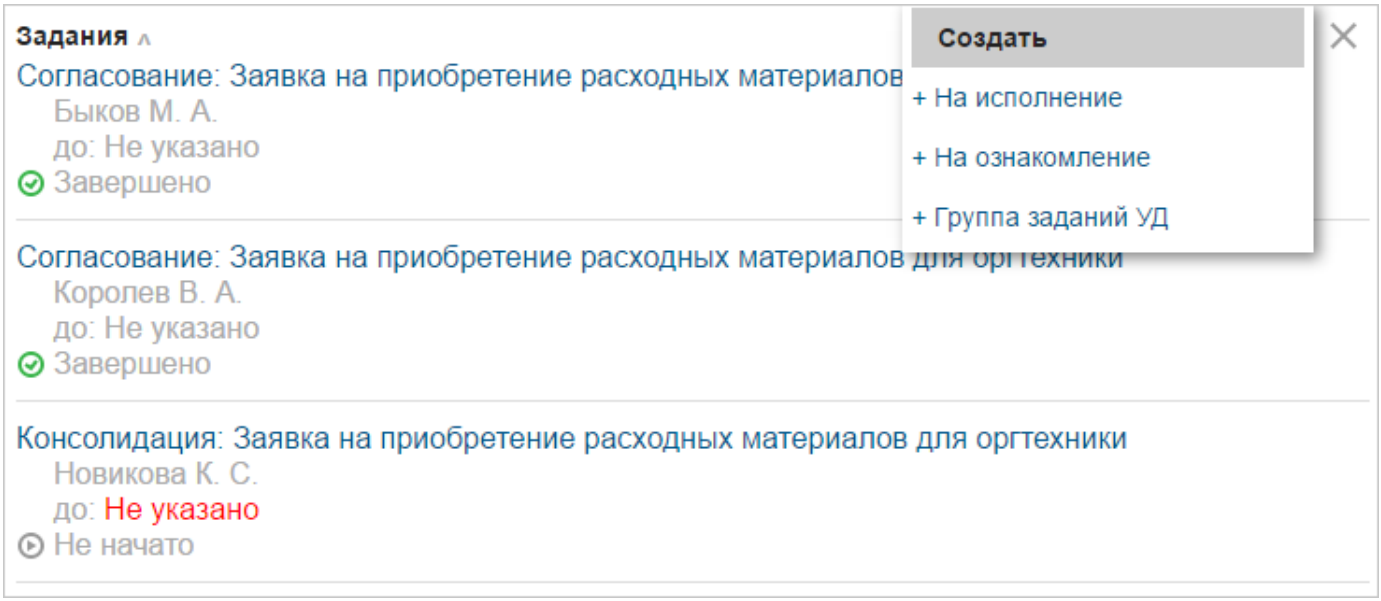


Рисунок 568. Пример элемента управления "Задания" со списком связанных заданий карточки

В списке заданий отображаются только задания и группы заданий непосредственно связанные с карточкой. Задания большей глубины вложенности не отображаются.

Данный ЭУ может быть добавлен только в разметку, предназначенную для просмотра карточек.

После изменения разметок с данным ЭУ потребуется перезапуск службы Сервера Web-клиента, чтобы отобразить изменения.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Дайджест представления

Определяет формат отображения списка заданий карточки.

- Когда флаг установлен, отображаются названия заданий и значки состояний.
- Когда флаг не установлен, отображается название задания, ФИО исполнителя, срок окончания и состояние задания или группы заданий

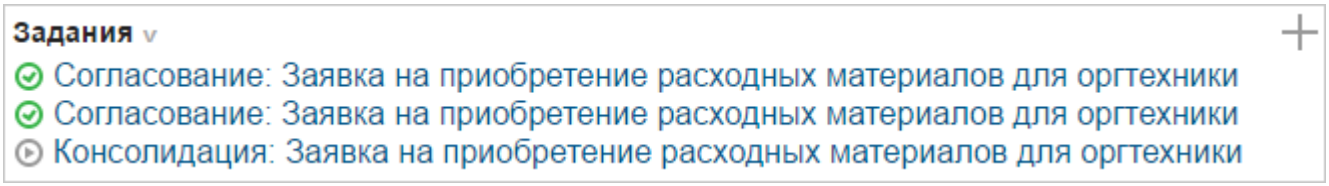


Рисунок 569. Список заданий в режиме "Дайджест"

Заголовок

Текст, который будет отображаться над блоком элемента управления. Заголовок выделяется жирным шрифтом.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее ссылку на карточку *Список ссылок на карточки заданий*.

Виды заданий для отображения

Выбор видов отображаемых в списке заданий. Если значение не указано, то в списке отображаются связанные задания всех видов.

Виды заданий для создания

Выбор видов заданий, которые могут быть созданы из карточки.

Для каждого вида может быть указано собственное отображаемое название.

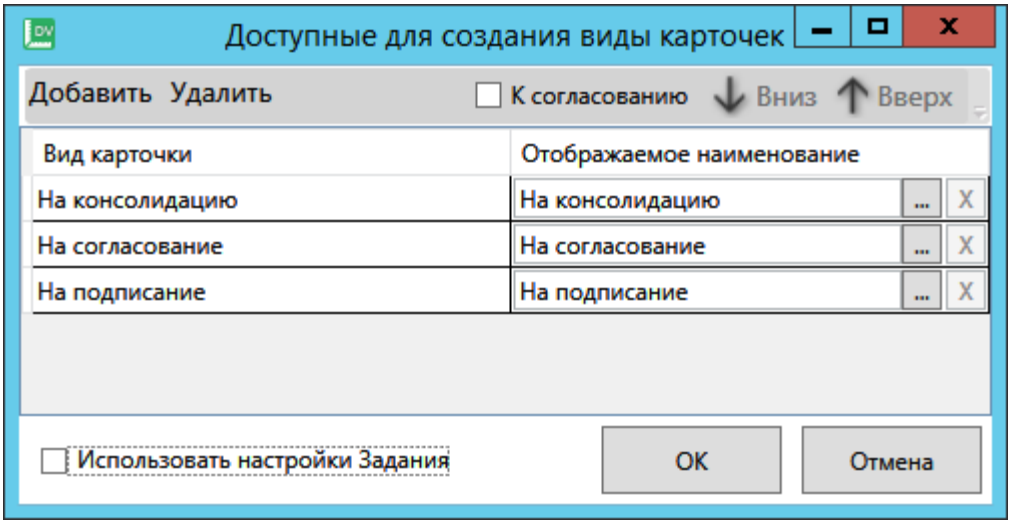


Рисунок 570. Форма для выбора видов карточек

Флаг **К согласованию** включает возможность создания в согласовании подчиненного задания согласования. Чтобы функция работала, в настройках этапа согласования должна быть включена настройка *Разрешить создание подчиненных заданий*.

При установленном флаге, в список видов карточек добавляется вид *Подчиненные к согласованию*.

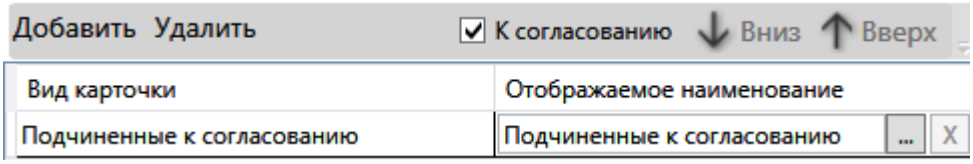


Рисунок 571. Форма для выбора видов карточек с установленным флагом К согласованию

Флаг **Использовать настройки Задания** включает режим, при котором список доступных для создания видов карточек будет получен из *Справочника видов карточек*. Данный режим предназначен для настройки разметок карточки *Задание*.

Если флаг установлен, виды подчиненных заданий и групп заданий будут получены из свойств **Виды подчиненного задания** и **Виды подчиненной группы заданий**.

При этом действуют следующие правила:

- Создание подчиненных заданий недоступно, если в *Справочнике видов карточек* используется режим ***Выбор из ограниченного списка***, но виды в списке не выбраны.
- Если в настройках вида задания разрешается создавать подчиненные задания единственного вида, то при создании будет выбран этот вид. Диалоговое

окно выбора вида не будет показано.

- При создании подчиненных заданий учитываются настройки переноса значений полей и ссылок из родительского задания.



Когда флаг Использовать настройки Задания установлен, возможность добавления видов карточек блокируется.

Операция редактирования

Выбор операции редактирования для добавления связанных заданий. Если значение не задано, добавление будет доступно всем (в случае, если добавление разрешено).

+ Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Создание групп заданий

Определяет возможность создания новой группы заданий из элемента управления. Создание групп заданий разрешено, если флаг установлен.

Создание заданий

Определяет возможность создавать новое задание из элемента управления. Создание заданий разрешено, если флаг установлен.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются,

когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Загружать синхронно (шт)

Определяет максимальное количество элементов списка, при котором должна использоваться синхронная загрузка данных. Если количество элементов больше указанного, все данные будут загружаться асинхронно.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим открытия

Определяет способ открытия ссылки:

- ***Текущая вкладка***. Ссылка будет открыта в текущей вкладке.
- ***Новая вкладка***. Ссылка будет открыта в новой вкладке.
- ***Новое окно браузера***. Ссылка будет открыта в новом окне веб-браузера.

Режим отображения

Определяет вариант отображения элемента управления в карточке:

- ***Список и создание*** — в ЭУ отображаются задания и кнопка создания нового задания или группы заданий. Задания отображаются в формате, определённом свойством ***Дайджест представления***.

- ***Только создание*** — в ЭУ отображается кнопка создания нового задания или группы заданий;
- ***Родитель*** — в элементе управления отображается ссылка на родительское задание. Для работы ЭУ в данном режиме в свойстве Поле данных должно быть указано поле карточки, содержащее ссылку на родительское задание.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед разворачиванием

Вызывается перед разворачиванием ЭУ кнопкой v.

Перед сворачиванием

Вызывается перед сворачиванием *Блока* кнопкой ^.

Перед созданием задания

Вызывается перед созданием задания.

После загрузки заданий

Вызывается после загрузки заданий.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При разворачивании

Вызывается после разворачивания ЭУ кнопкой v.

При сворачивании

Вызывается после сворачивания *Блока* кнопкой ^.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Информация о делегировании

Элемент управления Информация о делегировании предназначен для отображения в карточке делегированного задания информации о делегировании: комментария, указанного при делегировании, и делегировавшего. Отображает информацию о последнем делегировании.

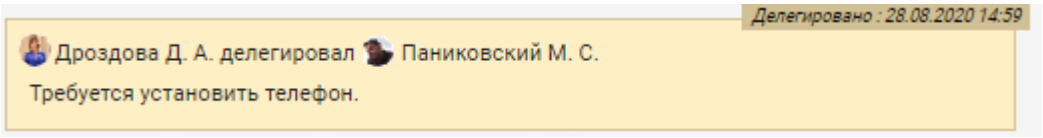


Рисунок 572. Пример элемента управления "Информация о делегировании"

ЭУ скрыт во всех случаях и отображается, только когда:

- Задание было делегировано с комментарием.
- При завершении задания был выбран собственный вариант завершения и задание было возвращено с делегирования. В этом случае в элементе управления будет отображаться вариант завершения.

Данный ЭУ может быть добавлен только в разметку карточек типа Задание.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Избранное

Элемент управления позволяет добавить карточку в папку *Избранное*.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Данные

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.

- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

История

Элемент управления История предназначен для отображения истории изменений в карточке.

ЭУ поддерживает два варианта отображения:

1. В карточке отображается кнопка, при нажатии которой будет открыто окно просмотра истории карточки.
2. В карточке отображаются последние изменения (количество отображаемых значений настраивается) и кнопка. При нажатии кнопки будет открыто окно просмотра истории карточки:

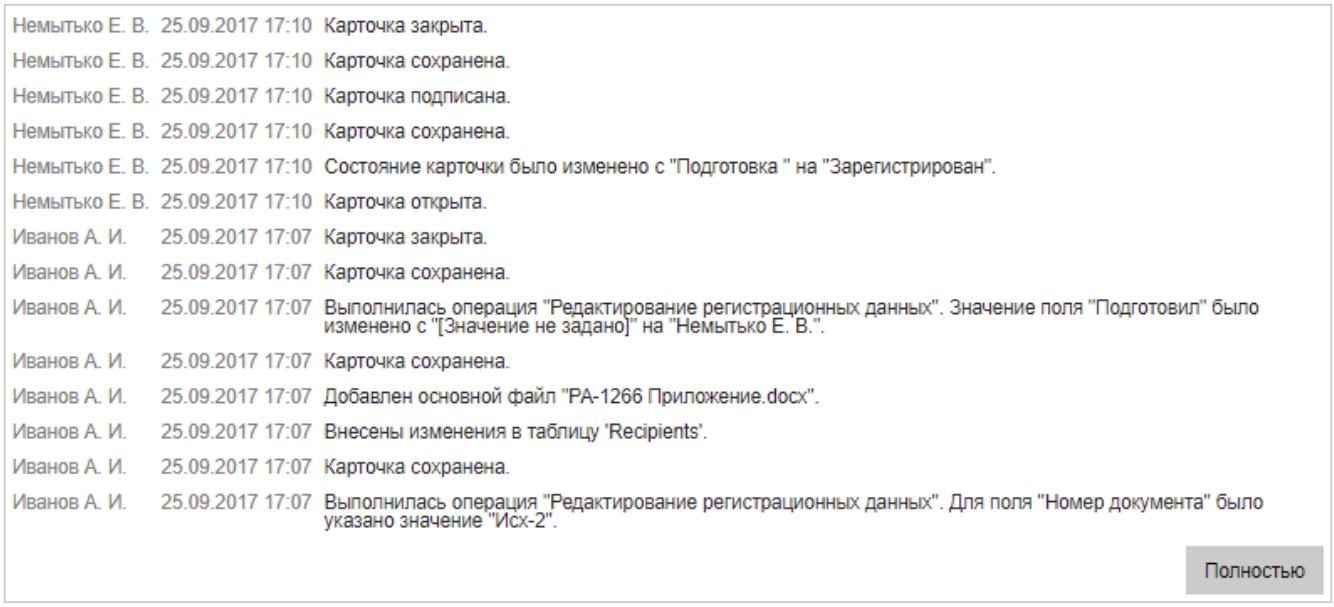


Рисунок 573. Элемент управления "История" с использованием просмотра в карточке

Окно просмотра истории изменений карточки предоставляет возможность фильтровать историю по автору, дате и описанию события карточки.

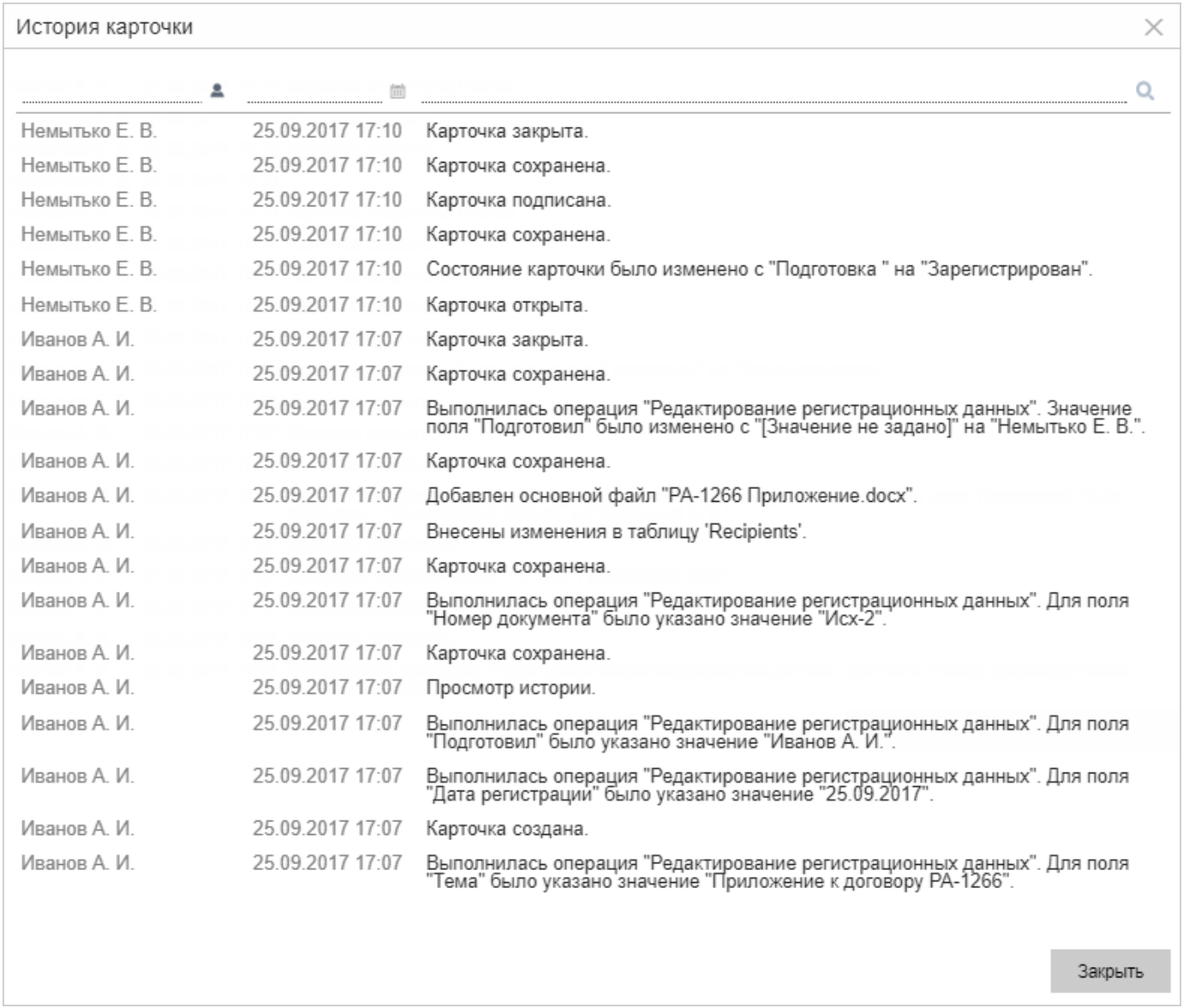


Рисунок 574. Пример журнала изменений карточки

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Записей в предпросмотре

Определяет количество отображаемых записей из истории карточки.

Использовать просмотр в карточке

Настройка отображения последних операций непосредственно в карточке. * Когда флаг установлен, в карточке отображаются несколько последних операций с карточкой и кнопка для открытия окна История карточки, в котором можно просмотреть всю историю. * Когда флаг не установлен, в карточке отображается кнопка для открытия окна История карточки.

Адаптивный

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст кнопки

Текст, отображаемый в кнопке. Если значение не установлено, отображается значение по умолчанию.

Локализуемый

Данные

Операция редактирования

Выбор операции, которая должна быть доступна пользователю для просмотра истории карточки.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Скрывать операции

Настройка списка видов операций, не отображаемых в ЭУ.

Настройка осуществляется в форме, открываемой при нажатии кнопки :

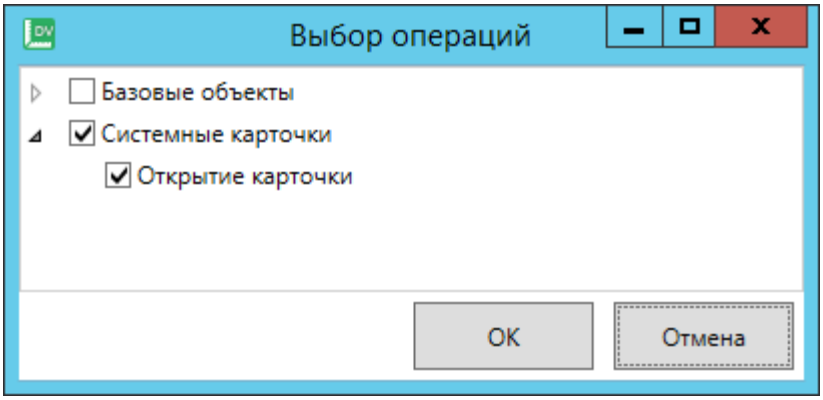


Рисунок 575. Окно настройки списка видов операций, не отображаемых в ЭУ

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед закрытием окна

Вызывается перед закрытием окна просмотра истории.

Перед открытием окна

Вызывается перед открытием окна просмотра истории.

После закрытия окна

Вызывается после закрытия окна просмотра истории.

После открытия окна

Вызывается после открытия окна просмотра истории.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка отправки письмом

Элемент управления позволяет отправлять карточку и файлы по почте. Для отправки файлов элемент управления может быть помещён только в ЭУ Список файлов. Элемент управления может быть добавлен только в карточки вида *Управление документами* в режиме просмотр.

HTML-форматирование:

Если в карточке присутствуют поля с HTML-кодом и для печати используются стандартные шаблоны, текст в таких полях будет выводиться на печать в отформатированном виде.

По умолчанию HTML-форматирование учитывается только в БД с версией Базовые объекты 5.5.5 и выше. Для БД с Базовые объекты более ранних версий необходимо вручную отредактировать шаблоны для карточки *Документ*, привязанные на вкладке *Экспорт* в *Справочнике видов карточек*. Стандартные шаблоны хранятся в папке `C:\Program Files\Docsvision\Platform\Tools\Database\BackOffice\Xslt`.

В справочнике видов можно указать актуальные шаблоны или вручную изменить стандартные. В шаблоне необходимо обернуть HTML-текст в тег `<EscapeHTML>`.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый в кнопке. Если значение не установлено, отображается

значение по умолчанию.

Локализуемый

Поведение

Использовать шаблон

Свойство активирует настройку шаблонов письма. Если свойство неактивно, тело письма останется пустым.

Окно *Доступные шаблоны письма* аналогично выбору шаблона печати. Однако, в шаблонах письма можно выбрать только один шаблон или не использовать шаблоны.

Прикладываемые файлы

Свойство позволяет выбрать, какие файлы по умолчанию будут включены в приложение к письму:

- ***Основные*** — к письму автоматически прикладываются только все основные файлы карточки.
- ***Дополнительные*** — к письму прикладываются только все дополнительные файлы.
- ***Все*** — к письму прикладываются все файлы карточки.
- ***Никакие*** — к письму не прикладываются файлы.

Изменение выбора файлов

Свойство определяет, может ли пользователь выбирать файлы, которые будут приложены к письму. Если свойство активно, при нажатии на кнопку пользователь увидит окно со списком файлов и возможностью их выбирать. Список файлов в окне зависит от значения свойства Прикладываемые файлы.

Ссылка на карточку

Свойство определяет, будет ли в тело письма добавлена ссылка на карточку в Web-клиенте в формате Ссылка на карточку в Веб Клиенте: ссылка. Ссылка добавляется в конце тела письма.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка электронной подписи

Элемент управления позволяет подписывать карточки без файлов (выполнять подписание атрибутов).



ЭУ можно разместить только в разметке карточки типа "Документ". Для разметок других типов ЭУ в библиотеке не отображается.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Данные

Операция редактирования Настроек подписи

Свойство регулирует доступность кнопки для вызова окна "Настройки подписи", в котором осуществляется настройка списка сертификатов.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии

кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Подписывать без метки

Флаг определяет, будет ли отображаться выпадающий список способов подписания. Если флаг установлен, будет выполняться подписание без метки и выпадающий список не будет отображаться.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед открытием диалога подписи

Перед открытием журнала подписей

Перед сохранением подписи

Перед формированием подписи

После открытия диалога подписи

После открытия журнала подписей

После подписания

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка Печати

Элемент управления Кнопка Печати добавляет возможность печати карточки с использованием XSLT-преобразования. Доступные шаблоны передаются модулем *Платформа* для каждого типа и вида карточки.

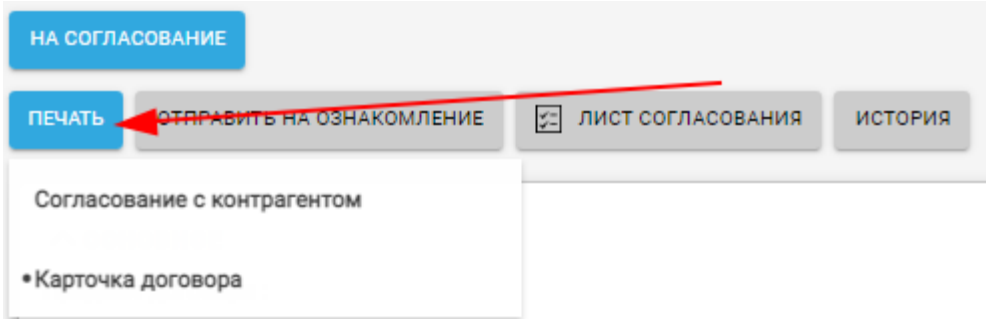


Рисунок 576. Пример отображения кнопки Печати в разметке

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый в кнопке. Если значение не установлено, отображается значение по умолчанию.

Локализуемый

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на кнопке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
  height: 24px;
```

```
}
```

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Доступные шаблоны печати

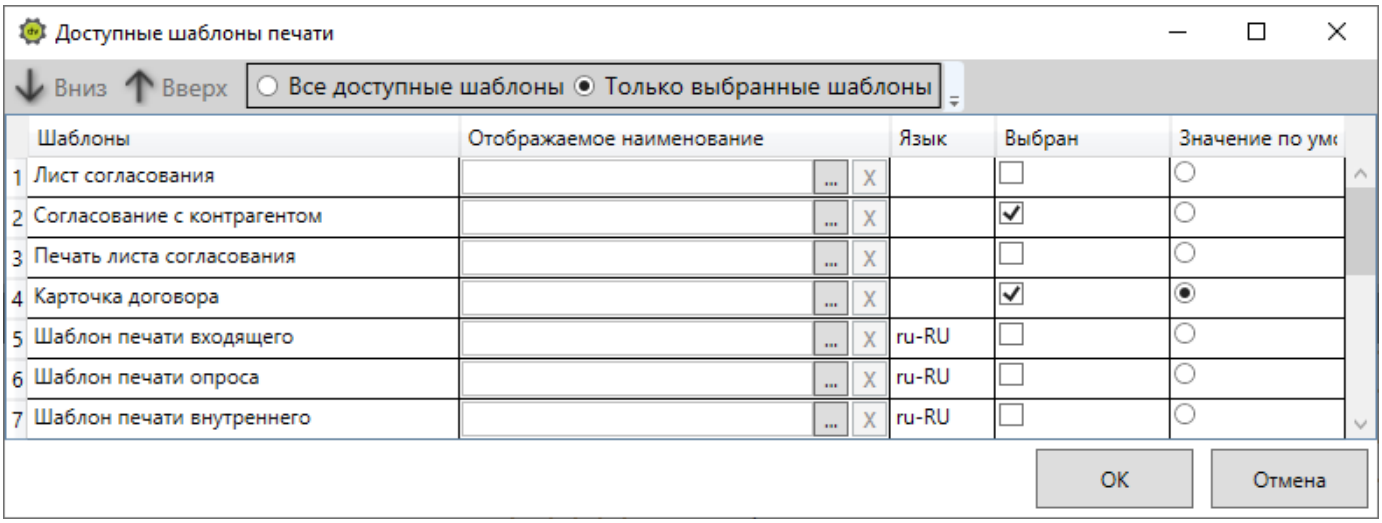


Рисунок 577. Выбор доступных шаблонов печати

Свойство позволяет выполнить следующие действия:

1. Установить количество доступных шаблонов (все или только выбранные).
2. Выбрать шаблон, который будет применяться по умолчанию.

Если из модуля *Платформа* передаётся два шаблона по умолчанию — один шаблон для типа, и один для вида карточки,

по умолчанию будет использован шаблон для вида.

3. Добавить локализуемое имя шаблона в колонке *Отображаемое имя*.
4. Узнать, к какому типу и виду принадлежит шаблон, если навести на него курсор.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка создания связанной карточки

Элемент управления Кнопка создания связанной карточки предназначен для создания карточек, связанных с текущей карточкой.

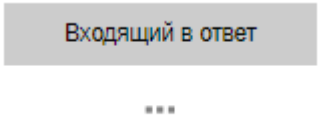


Рисунок 578. Пример элемента управления "Кнопка создания связанной карточки"

ЭУ представляет собой набор кнопок, каждая из которых соответствует режиму создания из Справочника видов карточек.

Особенности элемента управления "Кнопка создания связанной карточки"

- При создании связанной карточки используются настройки режима создания из *Справочника видов карточек*, включающие шаблон, тип ссылки. Свойство `Отображать диалог выбора папки` (размещение) игнорируется.
- В связанную карточку копируются значения полей текущей карточки, если это предусмотрено способом создания, который выбран в режиме создания.

В ЭУ отображаются только режимы создания, которые удовлетворяют следующим условиям:

1. В справочнике видов карточек у режима создания установлен флаг `Отображать команду`.
2. Если в режиме создания указан *Вид родительской карточки*, то вид текущей карточки должен ему соответствовать.
3. Если в режиме создания указана *Операция редактирования*, то она должна быть доступна пользователю.
4. Если в свойстве `Способ создания` выбрано значение, режим создания должен соответствовать ему.



После изменения настроек режима создания в *Справочнике видов карточек* нужно перезапустить службу Сервера Web-клиента, чтобы изменения стали доступны в Web-клиенте.



Данный ЭУ может быть добавлен только в разметку просмотра.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Вертикальная ориентация

Определяет формат отображения кнопок элемента управления.

+ * Когда флаг установлен, кнопки выстраиваются вертикально. * Когда флаг не установлен, кнопки выстраиваются горизонтально.

Адаптивный

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст

Текст, отображаемый на кнопке, если доступен **только один** режим создания. Если доступно несколько режимов создания, или текст не указан, на кнопке будет отображаться название режима создания.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Ограничение количества кнопок

Определяет максимальное количество кнопок режимов создания, отображаемых

пользователю.

+ * Если пользователю доступно больше режимов создания, чем указано в свойстве, будут показаны только первые `N` режимов и кнопка **...** для показа всех режимов.

+ * Если в настройке указано значение `0` (значение по умолчанию), отображаются все доступные режимы создания.

Адаптивный

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства `Видимость`:

- Когда флаг `Видимость` установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг `Видимость` установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг `Видимость` НЕ установлен, ЭУ всегда скрыт.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим открытия

Выбор способа открытия окна создания связанной карточки: ***Текущая вкладка***, ***Новая вкладка***, ***Новое окно браузера***.

Способ создания

Ограничивает список режимов создания, которые будут доступны пользователю.

В списке будут отображаться только режимы со способом создания, выбранным в свойстве `Способ создания`. Если в настройке выбрано значение `-` (значение по умолчанию), отображаются все доступные режимы создания.

Для выбора реализованы расширения для следующих способов:

- ***По ссылке***

- ***Ответ на Входящий***
- ***Ответ на Исходящий***
- ***Дополнительное соглашение***
- ***Акт***

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При запросе параметров

Вызывается при инициализации дополнительных параметров для запроса.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Лист ознакомления

Элемент управления доступен только для карточек типа *Документ*.

При работе с карточкой типа "Документ", содержащей этот ЭУ, пользователь получает возможность:

1. Настраивать выводимый лист ознакомления.
2. Выводить лист ознакомления с документом на отдельную вкладку в браузере.
3. Распечатывать лист ознакомления средствами браузера.

Элемент может отображать только:

- Непосредственные подчинённые задания и их подчинённые задания.
- Задания подчинённых групп заданий.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст кнопки

Текст, отображаемый на кнопке операции.

Локализуемый

Данные

Виды заданий для отображения

Регулирует, какие задания отображаются в листе ознакомления. По умолчанию — *На ознакомление*.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Шаблон печати

Выбрать можно только один шаблон.

Видимость подчинённых заданий

- Если флаг не установлен, в листе ознакомления отображаются только задания из списка заданий.
- Если флаг установлен, дополнительно отображаются подчинённые задания.

По умолчанию подчинённые задания не отображаются.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Сканирование

Элемент управления Сканирование добавляет в элемент Список файлов функцию сканирования документов с прикладыванием полученного файла к карточке.

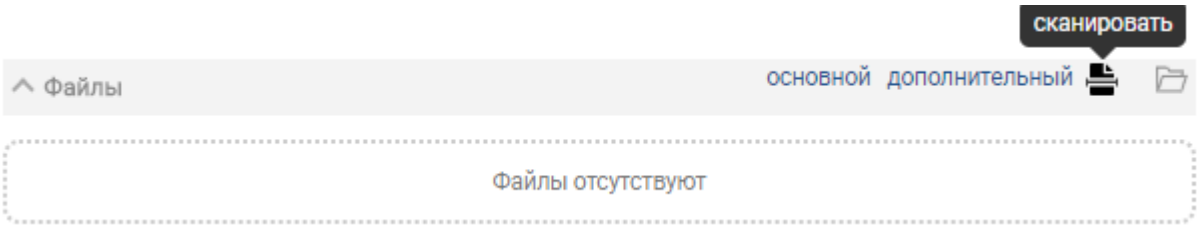


Рисунок 579. Пример элемента управления "Сканирование"

> Чтобы функция сканирования документов работала, на компьютере пользователя необходимо установить и запустить *Docsvision DVWebTool* или *Docsvision DVSupService*. Подробнее см. в разделах "Установка и запуск программы DVWebTool" и "Установка и запуск службы DVSupService".

Сканирование осуществляется в отдельном диалоговом окне, вид которого представлен на следующем рисунке.

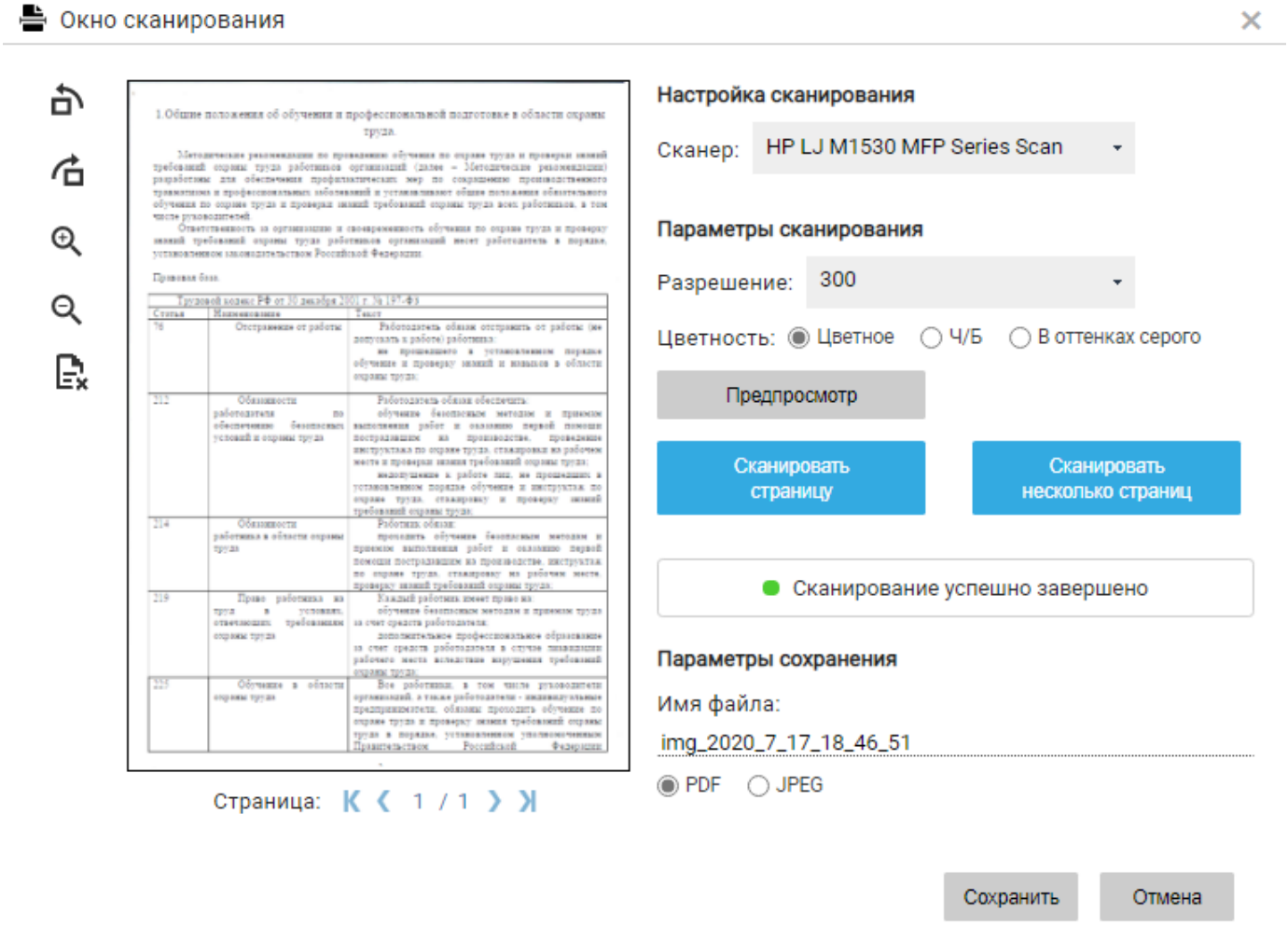


Рисунок 580. Диалоговое окно сканирования

Пользователь может настроить параметры сканирования:

- Разрешение из предопределённого списка.
- Цветность: цветное, черно-белое, оттенки серого.
- Формат файла: `.pdf`, `.jpeg`.
- Указать имя файла.



Доступно одиночное и потоковое сканирование (несколько страниц).

Разрешение на прикладывание файлов с помощью данного элемента управления определяется разрешениями родительского элемента *Список файлов*. Настройки элемента *Список файлов* также определяют возможность прикладывания основного и/или дополнительного файла.

Настройки сканирования сохраняются для текущего браузера пользователя.

Сбросить настройки можно из раздела Профиль пользователя Web-клиента.

Для включения функции сканирования добавьте элемент **Сканирование** в список операций (**fileListFileListOperations**) элемента Список файлов:

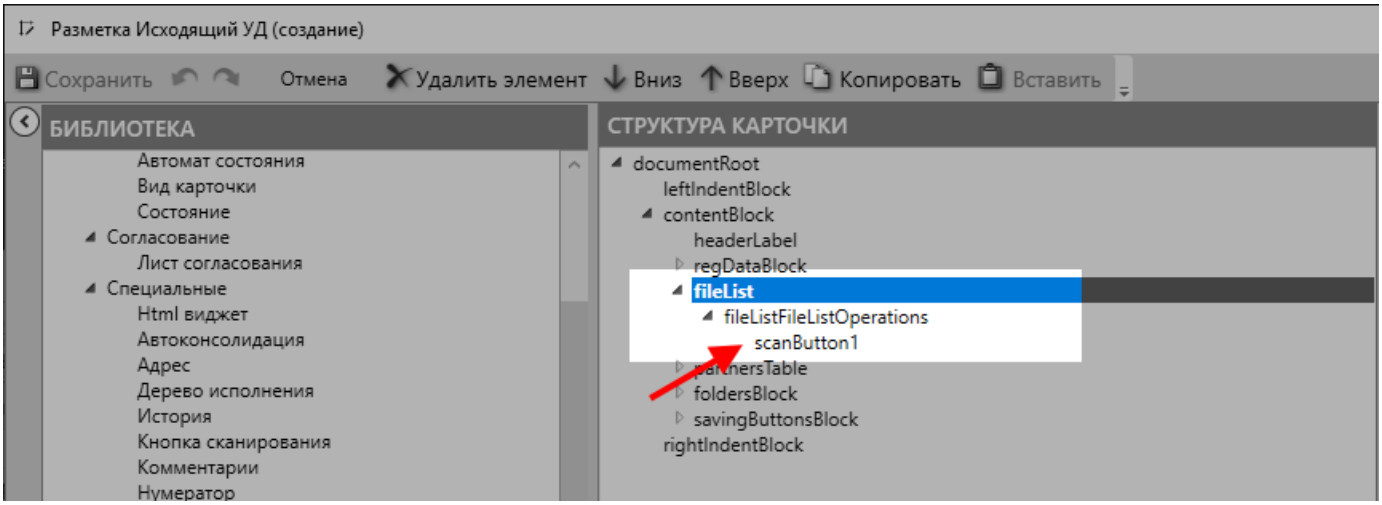


Рисунок 581. Список операций элемента "Список файлов" с элементом "Сканирование"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Комментарии

Элемент управления Комментарии предназначен для просмотра и добавления комментариев в карточке.



Рисунок 582. Элемент управления "Комментарии" в разметке карточки

ЭУ может быть добавлен в разметку просмотра или редактирования.

Пользователь может добавлять новые комментарии, изменять и удалять существующие (собственные и чужие) при наличии прав на данные операции по ролевой модели безопасности. Собственные комментарии отображаются с правой стороны, чужие — с левой.

При наличии прав на операции, над строкой комментария отображаются кнопки изменения и удаления. Если настроено ограничение на количество отображаемых комментариев, над списком может отображаться ссылка для показа всех комментариев с указанием количества скрытых комментариев.

Комментарии сортируются по порядку добавления. Автоматическое обновление списка комментариев не предусмотрено.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в поле для ввода комментария.

Локализуемый

Ширина блока комментария

Определяет ширину блока, в котором отображается отдельный комментарий, относительно ширины элемента управления. Используется для визуального разделения своих и чужих комментариев. Ширина по умолчанию — 80%.

Данные

Источник данных

Выбор секции и полей карточки с комментариями из данной или связанной карточки. Настройка осуществляется в форме, открываемой при нажатии кнопки :

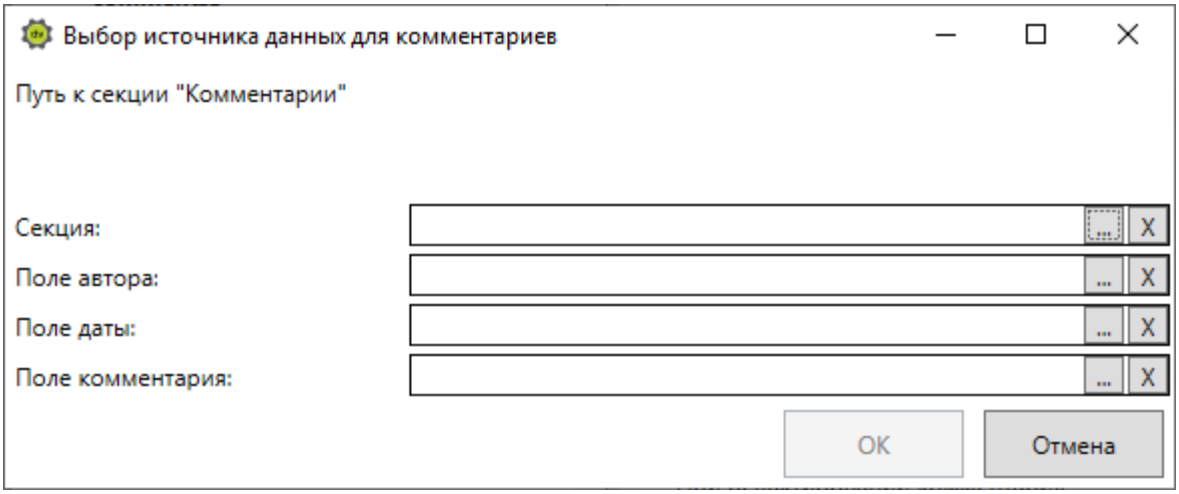


Рисунок 583. Окно настройки источника данных с комментариями

Настройки источника содержат следующие элементы:

- *Секция* — выбор секции с данными комментариев. Если к разметке подключены дополнительные источники данных, для выбора будут доступны секции карточки и дополнительных источников.
- *Поле автора*, *Поле даты* и *Поле комментария* — выбор полей из выбранной секции, содержащих соответствующие секции.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Добавление комментария

Выбор операции редактирования, которая должна быть доступна пользователю, для разрешения добавления комментариев. Если операция не выбрана, добавление комментариев разрешено.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Изменение всех комментариев

Операция редактирования, которая определяет разрешения пользователя на

изменение/удаление чужих и собственных комментариев. Если операция не выбрана, изменение/удаление чужих и собственных комментариев разрешено.

Изменение только своих комментариев

Операция редактирования, которая определяет разрешения пользователя на изменение/удаление собственных комментариев. Если операция не выбрана и/или доступна операция редактирования всех комментариев, изменение/удаление собственных комментариев разрешено.

Использовать HTML редактор

При включении данное свойство позволяет упоминать сотрудников в комментариях и форматировать текст комментария в процессе ввода.

Количество отображаемых комментариев

Максимальное количество (последних) отображаемых в списке комментариев. Если комментариев больше указанного числа, в ЭУ будет показана ссылка для показа всех комментариев.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед добавлением комментария

Вызывается перед сохранением нового комментария.

Перед редактированием комментария

Вызывается перед сохранением изменений комментария.

Перед удалением комментария

Вызывается перед удалением комментария.

После смены данных

Вызывается после изменения содержимого элемента управления.

При добавлении комментария

Вызывается после добавления комментария.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При редактировании комментария

Вызывается после изменения комментария.

При удалении комментария

Вызывается после удаления комментария.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Нумератор

Элемент управления Нумератор предназначен для присвоения и отображения регистрационного номера документа.

Вариант отображения ЭУ зависит от свойства **Режим редактирования**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

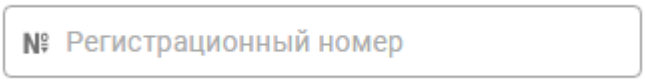


Рисунок 584. Элемент управления "Нумератор" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Новый номер может быть выдан при нажатии на ссылку в ЭУ или при наведении курсора на него.



Рисунок 585. Элемент управления "Нумератор" в режиме редактирования по месту с возможностью ручного ввода номера

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

ЭУ Нумератор позволяет присваивать следующий свободный номер автоматически (согласно выбранному правилу присвоения номеров) при нажатии кнопки № или вводить номер самостоятельно.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если номер не выдан.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее ссылку на номер в карточке *Нумератор*.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Генерировать номер после сохранения

Определяет необходимость автоматической генерации номера при сохранении карточки:

- Когда флаг установлен и номер не присвоен, после выполнения сохранения карточки номер будет сформирован автоматически.
- Когда флаг не установлен, автоматическое получение номера не производится.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Значение по умолчанию

Значение, присваиваемое ЭУ при создании карточки.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Показывать иконку

Свойство регулирует отображение иконки элемента управления. Когда флаг установлен (по умолчанию) иконка ЭУ отображается. Когда флаг снят, иконка не отображается.

Правило генерации номера

Раскрывающийся список с правилами нумерации *Конструктора правил нумерации*, зарегистрированными для данного типа карточки.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту*** — значение изменяется в отдельном окне, которое открывается нажатием на элемент управления. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.
- ***Редактирование*** — значение изменяется непосредственно в элементе управления. Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования*** — значение изменить нельзя.

Ручной ввод разрешён

Определяет возможность ручного ввода номера:

- Когда флаг установлен, номер может быть введён вручную или присвоен автоматически при нажатии кнопки №.
- Когда флаг не установлен, номер может быть присвоен только автоматически.



При включенном свойстве Ручной ввод разрешён проверка уникальности не выполняется.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед началом генерации номера

Вызывается перед получением номера, при нажатии кнопки №.

После окончания генерации номера

Вызывается после получения номера.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Отображение варианта завершения

Элемент управления Отображение варианта завершения предназначен для отображения варианта, с которым было завершено задание.

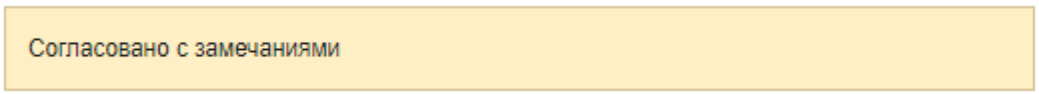


Рисунок 586. Элемент управления "Отображение варианта завершения" с вариантом завершения задания согласования

ЭУ может быть использован для отображения:

- Варианта завершения задания согласования.
- Варианта завершения задания другого вида, если в настройках вида указаны собственные варианты завершения и при завершении задания был выбран один из них.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Отображение исполнителей

Элемент управления Отображение исполнителей предназначен для показа списка выбранных и текущих исполнителей задания, и истории исполнения задания.

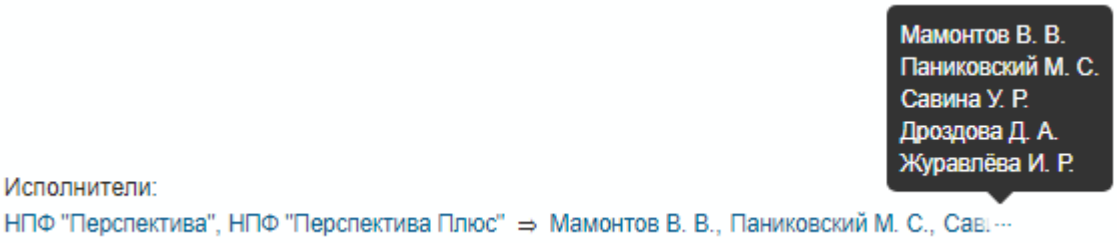


Рисунок 587. Элемент управления "Отображение исполнителей"

В зависимости от соответствия списка текущих и выбранных исполнителей, элемент управления может содержать:

Список выбранных исполнителей	Если выбранные исполнители соответствуют текущим.
Список выбранных исполнителей и, текущих исполнителей через символ =>	Если выбранные и текущие исполнители отличаются.
Журнал "История исполнения"	Если задание было делегировано или взято в работу сотрудником не из списка (заместителем). Журнал открывается по нажатию по списку исполнителей.

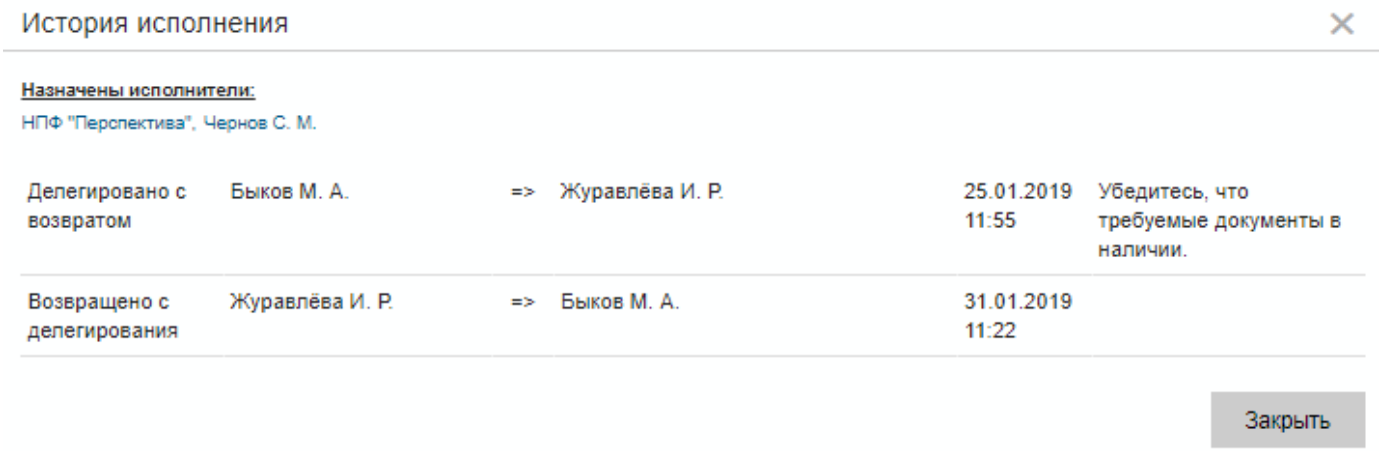


Рисунок 588. История исполнения

Если список исполнителей (текущих или выбранных) не помещается ЭУ, в конце списка будет добавлено многоточие. При наведении курсора мыши на многоточие будет показан полный список исполнителей.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:

- Когда флаг установлен, текст метки отображается всегда.
- Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Подсказка к значению (при наведении на исполнителя)

Содержимое всплывающей подсказки, отображаемой при наведении курсора мыши на исполнителя. Возможные варианты:

- ***ФИО***.
- ***ФИО + должность***.
- ***ФИО + подразделение + должность***.
- ***Строка отображения***. Используется формат отображения, установленный в *Справочнике сотрудников*.
- ***Не используется***. Во всплывающей подсказке отображается текст из поля Подсказка.

Подсказка к значению расширенная (при наведении на многоточие)

Содержимое всплывающей подсказки, отображаемой при наведении курсора мыши на многоточие. Возможные варианты см. в описании свойства Подсказка к значению (при наведении на исполнителя).

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Формат отображения

Формат отображения данных сотрудника в списке исполнителей. Возможные варианты:

- ***ФИО***.
- ***Строка отображения***.

Значение по умолчанию

Поле данных (для конечных исполнителей)

Выбор поля карточки, из которого будет получен список текущих исполнителей. Возможные варианты:

+ * ***CurrentPerformers*** (значение по умолчанию), * MainInfo.Performers.

Поле данных (для начальных исполнителей)

Выбор поля карточки, из которого будет получен список выбранных исполнителей. Возможные варианты:

- ***MainInfo.SelectedPerformers*** (значение по умолчанию) — может содержать группы, роли, поисковые слова и др.,
- ***MainInfo.Performers***.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения. Не может быть изменён.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.

- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед закрытием диалога

Вызывается при закрытии журнала *История исполнения*.

Перед открытием диалога

Вызывается при открытии журнала *История исполнения*.

После закрытия диалога

Вызывается после закрытия журнала *История исполнения*.

После открытия диалога

Вызывается после открытия журнала *История исполнения*.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Отправка на ознакомление

Элемент управления Отправка на ознакомление предназначен для отправки документа на ознакомление.

Элемент управления добавляет в карточку кнопку отправки, при нажатии которой отображается боковая панель отправки документа на ознакомление

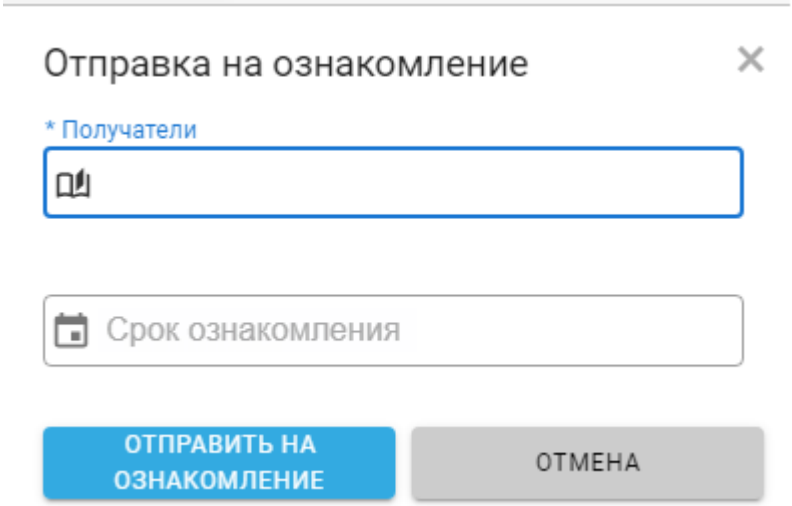


Рисунок 589. Панель отправки документа на ознакомление, предоставляемая элементом управления "Отправка на ознакомление"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый на кнопке отправки документа на ознакомление.

Локализуемый

Данные

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед отправкой

Вызывается при отправке документа на ознакомления. С возможностью отмены.

После закрытия

Вызывается после закрытия панели отправки на ознакомление.

После открытия

Вызывается после открытия панели отправки на ознакомление.

После отправки

Вызывается после отправки документа на ознакомления.

Перед закрытием

Вызывается перед закрытием панели отправки на ознакомление.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

Перед открытием

Вызывается перед открытием панели отправки на ознакомление.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Панель выбора исполнителей

Элемент управления Панель выбора исполнителей предназначен для выбора исполнителей и варианта исполнения группы заданий.

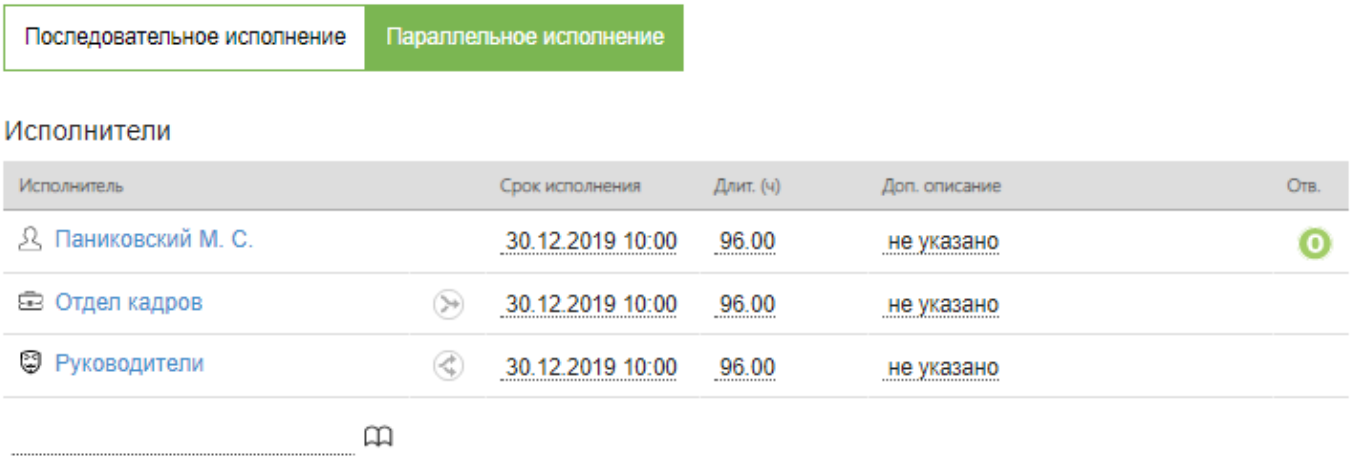


Рисунок 590. Элемент управления "Панель выбора исполнителей"

ЭУ предоставляет возможность:

- Выбирать вариант исполнения: последовательно или параллельно.
- Выбирать исполнителей из Справочника сотрудников: сотрудников, организации/подразделения, группы и роли.
- Назначать индивидуальные сроки исполнения.
- Выбирать ответственного исполнителя.

ЭУ может пересчитывать сроки исполнения участников при изменении общего и индивидуальных сроков исполнения, и оповещать пользователя, если сроки исполнения всех исполнителей больше общего срока исполнения.

ЭУ предназначен для разметки редактирования карточки Группа заданий.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Данные

Настройки выбора исполнителей

Данное свойство разрешает или запрещает выбирать исполнителей из сотрудников, подразделений, групп и ролей.

Нажмите кнопку ..., чтобы открыть панели настройки.

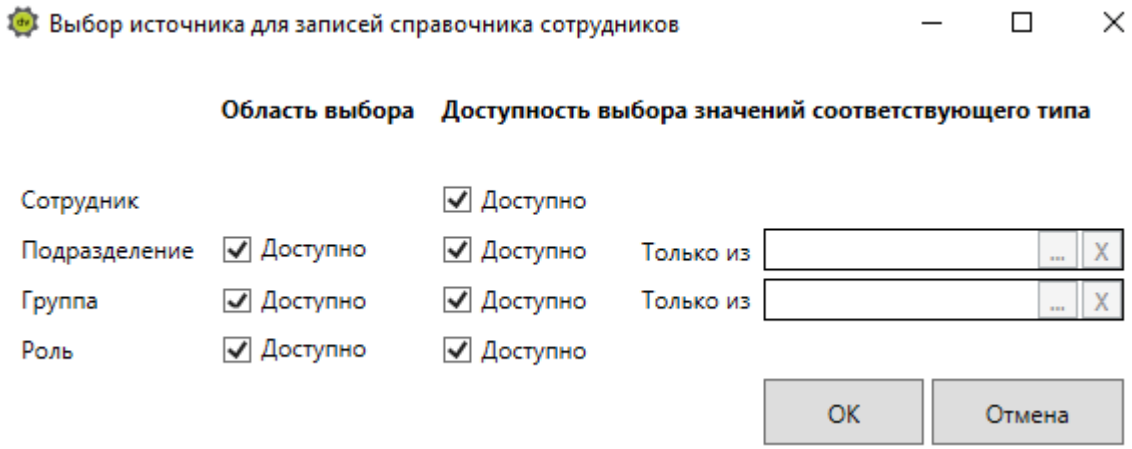


Рисунок 591. Панель настройки выбора исполнителей

Флаги в колонке *Область выбора* определяют возможность выбора элементов из определённого раздела Справочника сотрудников. Если флаг установлен, то пользователь сможет выбирать записи из подразделений, групп или ролей.

Флаги в колонке *Доступность выбора значений соответствующего типа* определяют возможность выбора соответствующего типа исполнителей: сотрудников, подразделений, групп или ролей.

Для подразделений и групп можно установить ограничение, выбрав в поле *Только из* определённое подразделение/группу, из которого будет разрешено выбирать исполнителей.

Операция редактирования

Выбор операции редактирования, которая должна быть доступна пользователю для возможности изменения списка исполнителей.

Если значение `Операции редактирования` наследуется от родительского Блока, название свойства меняется на `Операция редактирования (наследовано)`.

Поведение

Автоматический пересчет сроков

Определяет необходимость автоматического пересчета сроков исполнения при изменении общего и/или индивидуальных сроков исполнения:

- Когда флаг установлен, сроки исполнения для исполнителей будут пересчитаны пропорционально ранее установленным значениям в случае, если изменился общий срок исполнения. Если изменился индивидуальный срок исполнения, будет пересчитан срок исполнения следующего исполнителя или исполнителей.
- Когда флаг снят, автоматический пересчет сроков не выполняется.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Использовать бизнес-календарь

Включает или отключает использование бизнес-календаря при расчете сроков исполнения (или длительности).



Бизнес-календарь должен быть включен в *Справочнике видов карточек* (установлен флаг `Использовать бизнес-календарь` в настройках вида задания).

Использовать последних

Включает/отключает отображение последних выбранных исполнителей.

Если область выбора ограничена группой, свойство `Использовать последних` не работает.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки.

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства `Видимость`:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Распределение заданий по умолчанию

Определяет вариант рассылки заданий для исполнителей вида *Подразделение*, *Группа* и *Роль*, выбираемый при создании новой группы: *Отдельное задание для каждого сотрудника*, *Одно задание для всех сотрудников* или *Одно задание для руководителя*.

Режим исполнения по умолчанию

Определяет вариант исполнения группы заданий, выбираемый при создании новой группы: "Последовательное исполнение" или "Параллельное исполнение".

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед добавлением исполнителей

Вызывается перед добавлением исполнителя в список.

Перед применением описания задачи

Вызывается перед сохранением нового значения дополнительного описания задания.

Перед применением сроков задачи

Вызывается перед сохранением нового значения срока исполнения задания.

Перед сменой исполнителя задачи

Вызывается перед изменением исполнителя задания.

Перед сменой общего интервала исполнения ГЗ

Вызывается перед сохранением нового срока исполнения группы заданий.

Перед сменой порядка заданий

Вызывается перед изменением порядка исполнения задания.

Перед сменой режима распределение задачи

Вызывается перед применением нового варианта рассылки заданий для подразделений, групп, ролей.

Перед сменой режима типа исполнения

Вызывается перед применением нового варианта исполнения группы заданий.

Перед удалением задачи

Вызывается перед удалением исполнителя из списка исполнителей.

Перед установкой ответственного

Вызывается перед выбором ответственного исполнителя.

После открытия диалога редактирования длительности

Вызывается после открытия диалога настройки индивидуальных параметров исполнения.

После открытия диалога редактирования описания

Вызывается после открытия диалога редактирования дополнительного описания задачи.

После смены общего интервала исполнения ГЗ

Вызывается после сохранения нового срока исполнения группы заданий.

После смены данных

Вызывается после изменения содержимого элемента управления.

При загрузке данных контрола исполнителей

Вызывается при загрузке данных в элемент управления выбора исполнителей.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Панель отчёта

Элемент управления Панель отчёта предназначен для добавления в карточку задания файлов и комментариев отчёта.

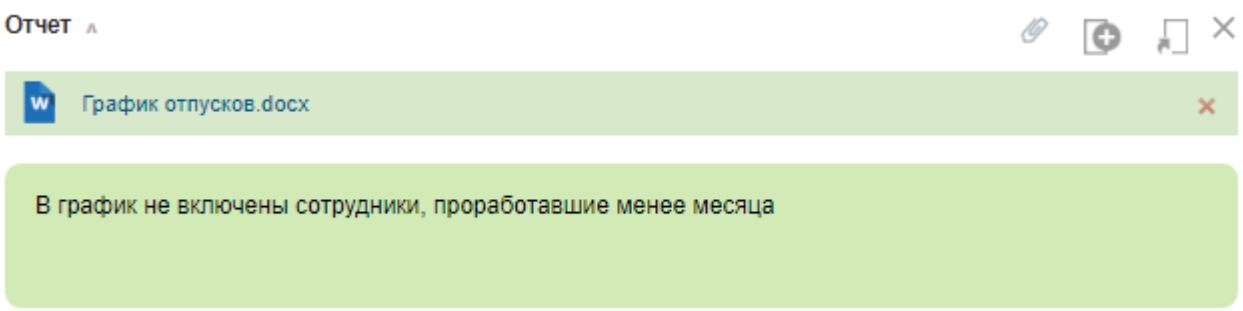


Рисунок 592. Элемент управления "Панель отчёта"

ЭУ предоставляет возможность добавлять отчёты: комментарии, файлы, существующие карточки и новые карточки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например,

JSON).

Поведение

Асинхронная загрузка

Определяет способ загрузки данных элемента управления.

Когда флаг установлен, данные загружаются после загрузки ЭУ. Когда флаг снят, данные загружаются вместе с ЭУ.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Параметры завершения задания

Элемент управления Параметры завершения задания предназначен для отображения дополнительных параметров завершения задания, настроенных в Справочнике видов карточек и в индивидуальных настройках задания.

Элемент управления может быть добавлен только в разметку локации **WebFrame › Разметки для элементов управления › Диалоги операций карточек › Завершить**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Предпросмотр файла

> По умолчанию предпросмотр документов в карточках отключен. Чтобы включить предпросмотр, необходимо настроить компоненты Р7-Офис Сервер документов/OnlyOffice.

Элемент управления Предпросмотр файла предназначен для отображения содержимого файла, выбранного в элементе Список файлов, Файловая панель, Файловая панель согласования.

> Если максимальный размер файла для предпросмотра ограничен настройками системы, вместо предпросмотра больших будет выводиться сообщение об ошибке Предпросмотр недоступен согласно настройкам системы. Для ознакомления с файлом скачайте его и откройте в соответствующей программе.

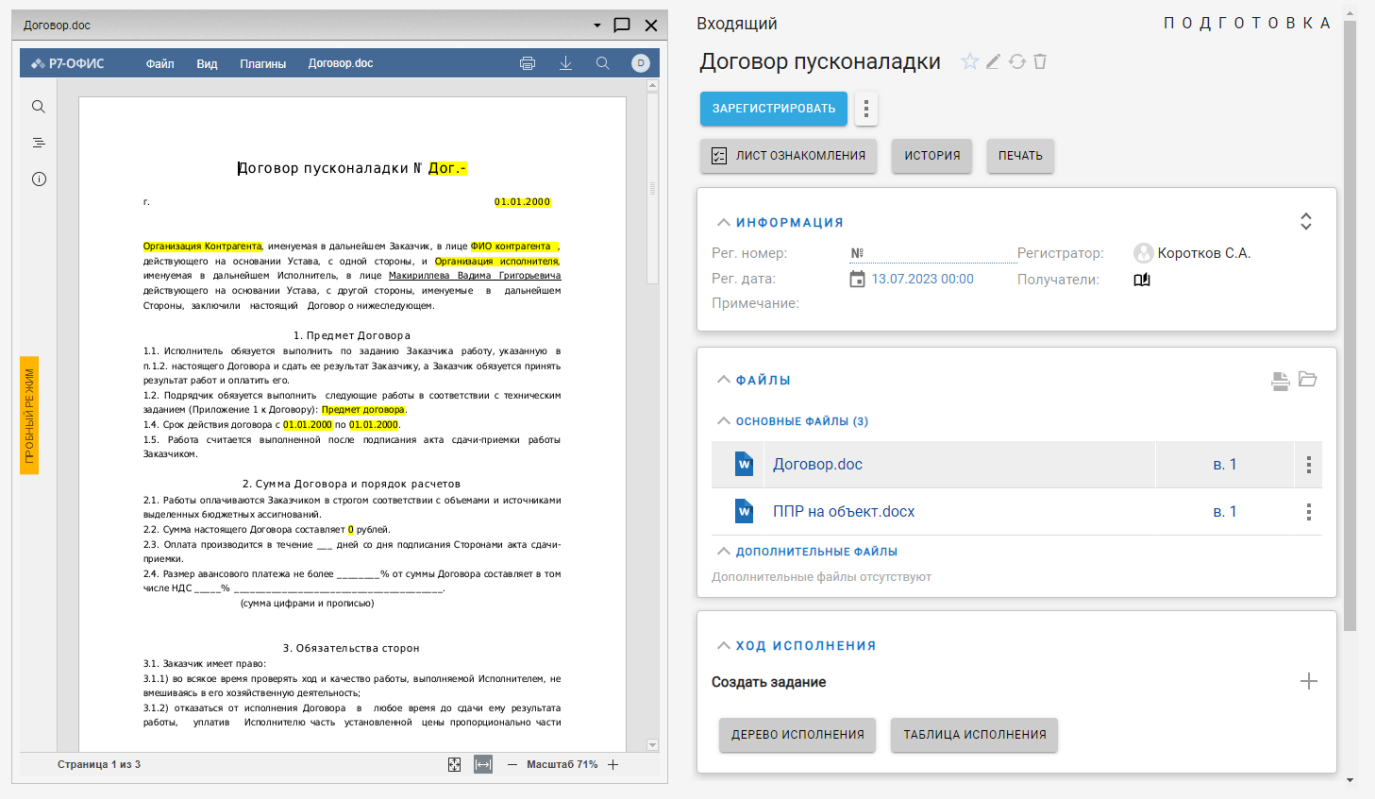


Рисунок 593. Пример элемента управления "Предпросмотр файла" в разметке карточки

Данный элемент управления позволяет:

- Отображать содержимое файла непосредственно в разметке просмотра и

редактирования карточки.

- Отображать содержимое файла при загрузке карточки без необходимости выбора файла.
- Переключаться между отображаемыми файлами.
- Работать с несколькими элементами Список файлов: если в разметке несколько данных элементов, ЭУ Предпросмотр файла будет показывать содержимое активного элемента Список файлов.

Важные особенности использования элемента "Предпросмотр файла"

- В разметку можно добавить только один элемент Предпросмотр файла.
- Элемент управления Предпросмотр файла должен быть добавлен в отдельный элемент Блок, в котором не должно быть других элементов кроме элемента Предпросмотр файла.
- Родительский блок элемента Предпросмотр файла должен размещаться в блоке, содержащем только два дочерних элемента (см. рисунок ниже): блок с элементом Предпросмотр файла и блок с остальными элементами разметки.

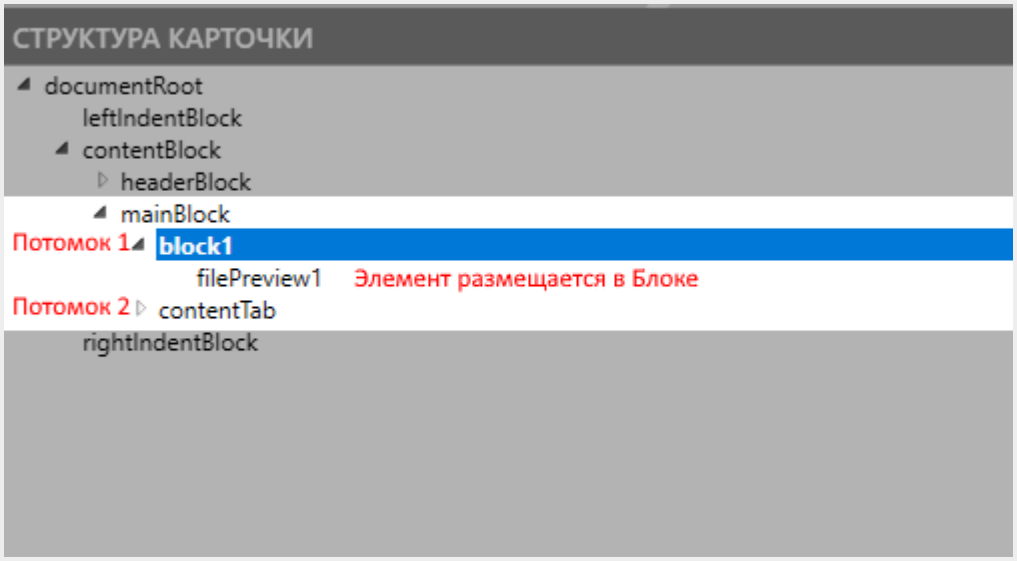


Рисунок 594. Пример структуры разметки, соответствующей требованиям

Типы файлов, для которых поддерживается предварительный просмотр:

- .pdf (в т.ч PDF/A-1a).
- Офисные: .docx, .pptx, .odt, .doc, .rtf, .dot, .dotx, .xlt, .xls, .xltx, .xlsx, .ods;

- Изображения: `.jpg`, `.jpeg`, `.gif`, `.png`, `.bmp`, `.tiff`, `.tif`;
- Прочие: `.htm`, `.adoc`, `.js`, `.xslt`, `.xsl`, `.xml`, `.asmx`, `.ascx`, `.aspx`, `.cpp`, `.h`, `.css`, `.idl`, `.ini`, `.java`, `.rc`, `.rc2`, `.shtm`, `.shtml`, `.txt`, `.log`, `.cs`, `.sql`, `.csv`.



Данный элемент управления может быть добавлен в разметки создания, просмотра и редактирования карточек *Документ*, а также **просмотра** и **редактирования** карточек *Задание* и *Группа заданий*.



Работа элемента управления в разметке создания карточек *Задание* и *Группа заданий* НЕ поддерживается.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Высота (в пикселях)

Устанавливает высоту элемента `Препросмотр файла` в пикселях.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента

управления. Перечисляются через пробел.

Комментарии

Включает или выключает возможность комментирования файла.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Раскрывать файл

Включает или выключает режим отображения содержимого файла при загрузке разметки. При выключенном свойстве содержимое файла отображается только после его выбора.

Если элемент `Список файлов` содержит несколько файлов, при включенном свойстве **`Раскрывать файл`** будет отображаться первый файл по списку.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина по умолчанию (в процентах)

Ширина элемента в процентах относительно родительского элемента.

События

Перед открытием/сокрытием блока комментариев

Вызывается перед открытием и закрытием блока комментариев.

Перед открытием/сокрытием окна

Вызывается перед закрытием окна предпросмотра.

Перед скачиванием файла

Вызывается при нажатии кнопки загрузки файла на компьютер.

Перед сменой файла

Вызывается перед сменой отображаемого файла.

После открытия/сокрытия блока комментариев

Вызывается после открытия и закрытия блока комментариев.

После открытия/сокрытия окна

Вызывается после закрытия окна предпросмотра.

После скачивания файла

Вызывается после нажатия кнопки загрузки файла на компьютер.

После смены файла

Вызывается после смены отображаемого файла.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Проверка на уникальность

Элемент управления Проверка на уникальность добавляет в карточку Документ функцию проверки уникальности атрибутов, критерии которой настроены в Справочнике видов карточек.

Проверка выполняется при изменении значения поля, сохранении карточки, а также при выполнении операций с ней. Неуникальные атрибуты карточки будут выделены цветом.

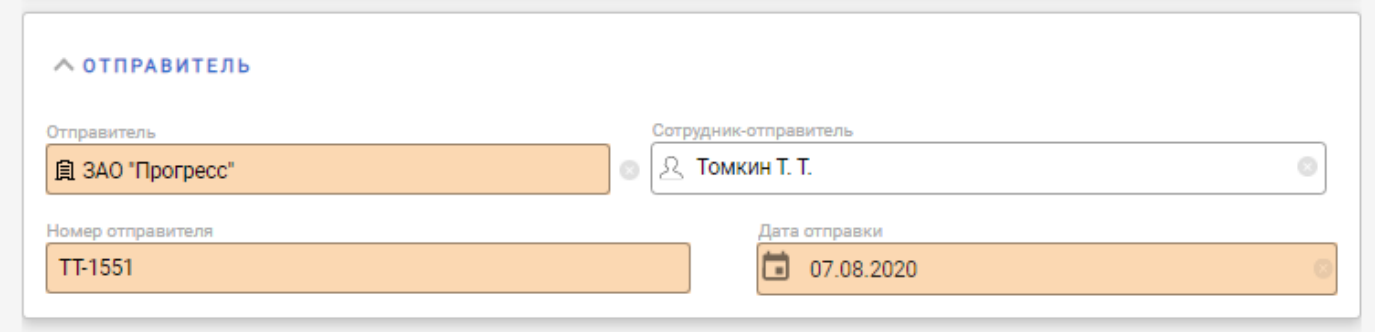


Рисунок 595. Выделение неуникальных атрибутов в карточке

Также элемент управления добавляет в карточки значок ⚠, указывающий на неуникальность карточки. При щелчке по значку будет открыто окно со списком нарушений, содержащем ссылку на карточку, с которой возник конфликт, а также список полей, нарушающих уникальность. В список полей также включаются неотображаемые в разметке поля, если уникальность нарушена в них.

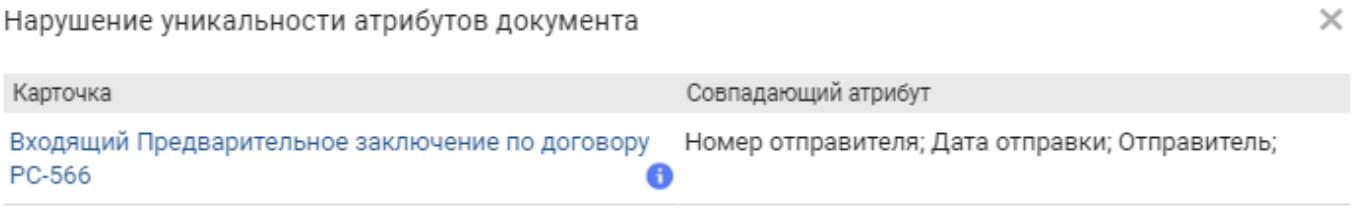


Рисунок 596. Результат проверки уникальности

Элемент управления может блокировать сохранение и/или изменение состояния карточки. Блокировка настраивается свойствами **Блокировать при смене состояния** и **Блокировать при сохранении**. В данном случае при сохранении или выполнении операции будет также показано окно со списком нарушений.

Элемент управления Проверка на уникальность предназначен для использования только в разметке карточки *Документ*.

⚠ Из-за особенности способа хранения данных проверка уникальности может не работать для значения элемента управления Текст в режиме HTML редактора.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Данные

Блокировать при смене состояния

Устанавливает список состояний, переход в которые должен блокироваться, если карточка не является уникальной.

Настройка списка осуществляется в диалоговом окне, открывающимся по нажатию ⋯.

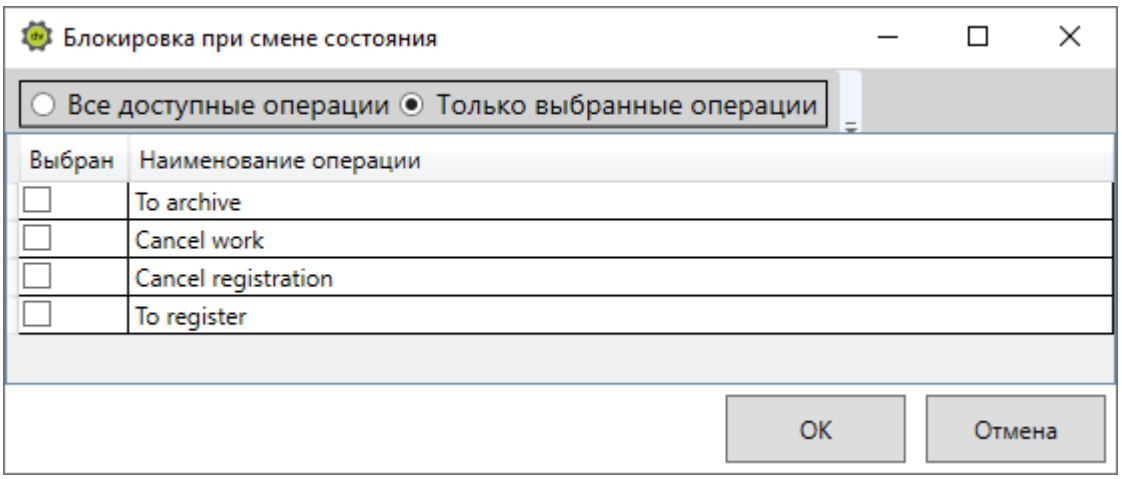


Рисунок 597. Диалоговое окно выбора состояний для блокировки

По умолчанию переходы в другие состояния не блокируются.

Настройка влияет только на изменение состояния карточки пользователем.

Блокировать при сохранении

Включает или выключает блокировку сохранения карточки, если карточка не является уникальной. По умолчанию блокировка включена.

Поведение

Css класс нарушения проверки уникальности

Название CSS класса со стилем, применяемым к атрибутам карточки, нарушающим уникальность. По умолчанию используется стиль **uniqueness-check-failed**, который может быть изменён на пользовательский.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Результаты поиска

Элемент управления Результаты поиска предназначен для отображения в разметке окна выбора ссылки (элемент "Ссылки") списка карточек, найденных поисковым запросом, или карточек выбранной папки.

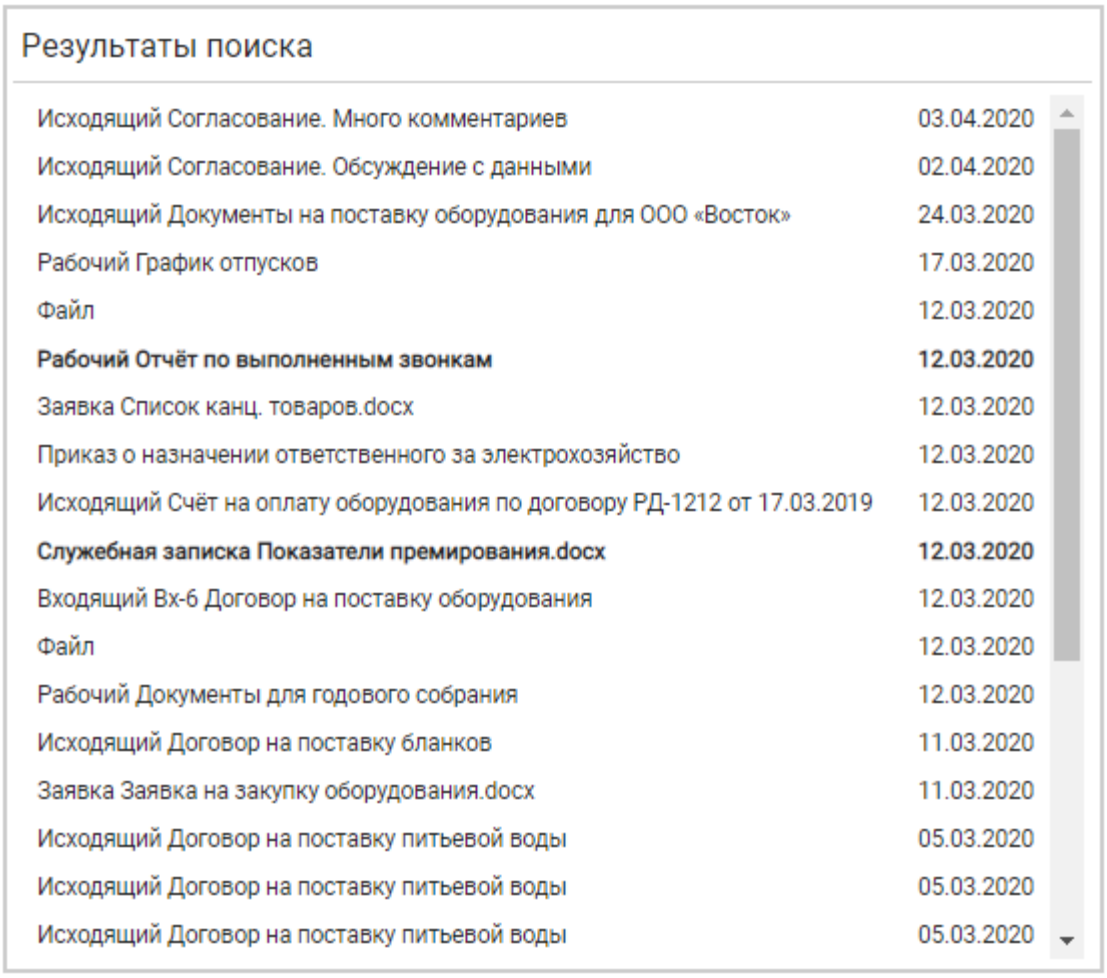


Рисунок 598. Элемент управления Результаты поиска с выбранными карточками

Элемент управления работает только совместно с другими элементами разметки `Поиск карточек для ссылок`: Таблица параметров запроса и "Выбор папки". Данная работа обеспечивается встроенными скриптами. Работа элемента в других ситуация не поддерживается.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед началом загрузки представлений

Вызывается перед началом загрузки представлений.

После окончания загрузки представлений

Вызывается после окончания загрузки представлений.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Родительская группа заданий

Элемент управления Родительская группа заданий предназначен для отображения родительской группы заданий в подчиненном задании.

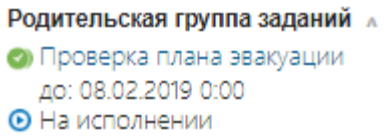


Рисунок 599. Элемент управления "Родительская группа заданий"

ЭУ содержит название группы заданий и срок выполнения.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства `Видимость`:

- Когда флаг `Видимость` установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг `Видимость` установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг `Видимость` НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента

управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Список файлов

Элемент управления Список файлов предназначен для работы со списком файлов карточки типа *Документ*.

> Данный ЭУ может быть добавлен только в разметку карточек типа Документ.

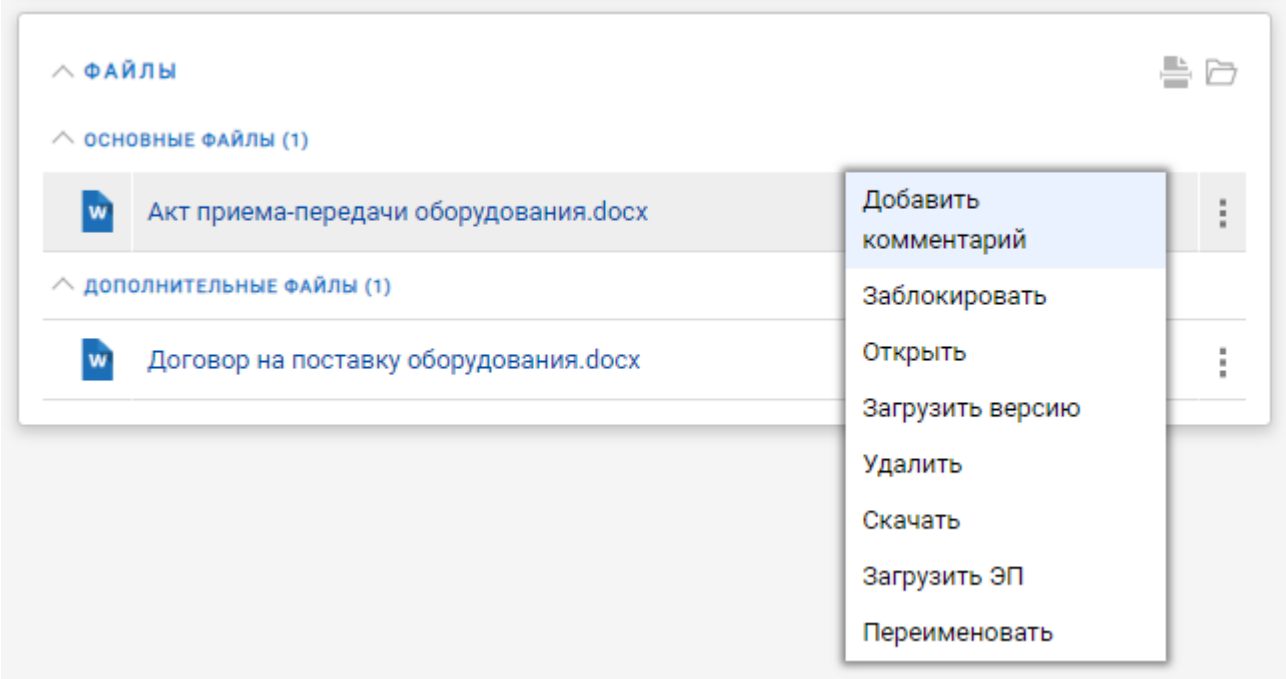


Рисунок 600. Пример элемента управления "Список файлов"

Данный элемент управления позволяет:

- Отображать основные и дополнительные файлы карточки.
- Добавлять или удалять основные и дополнительные файлы с файловой

системы и со сканера.



Для использования функции сканирования в элемент управления Список файлов требуется добавить элемент "Сканирование".

- Работать со списком версий.
- Изменять тип добавленного файла (основной, дополнительный) перетаскиванием.
- Переименовывать файлы.
- Просматривать содержимое файла в окне предварительного просмотра: собственная реализация или с использованием компонента "Предпросмотр файла".
- Подписывать файлы ЭП.
- Формировать печатную версию со штампом электронной подписи для основных файлов.
- Загружать отсоединённую электронную подпись.



Добавленные через ЭУ файлы сразу сохраняются на сервер. При отказе от сохранения карточки (нажатии кнопки **Отменить**) файл остаётся в карточке, так как уже был сохранён на сервере. Отменяются только те данные, которые не были отправлены на сервер.

Возможность выполнения определённой операции пользователем регулируется ролевой моделью.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Загружать синхронно (шт)

Определяет максимальное количество элементов списка, при котором должна использоваться синхронная загрузка данных. Если количество элементов больше указанного, то все данные будут загружаться асинхронно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Операция редактирования для добавления подписи

Определяет операцию, которая должна быть доступна пользователю для возможности загрузки файла отсоединённой электронной подписи.

Разрешить дополнительные файлы

Разрешает или запрещает добавление дополнительных файлов с помощью

данного элемента управления.

Сортировка файлов

Свойство позволяет сортировать файлы в списке, доступны несколько способов сортировки по имени или времени добавления:

- ***По имени по возрастанию***
- ***По имени по убыванию***
- ***По времени добавления по возрастанию***
- ***По времени добавления по убыванию***
- По умолчанию используется сортировка по имени по возрастанию.

Сортировка пользователем

Свойство разрешает или запрещает конечному пользователю выбирать собственный вариант сортировки на странице с элементом управления.

Когда свойство активно (флаг установлен), сортировка разрешена (значение по умолчанию). Когда свойство отключено (флаг снят), сортировка запрещена.

Скрыть кнопку подписи

Позволяет скрыть кнопку подписи в списке файлов. По умолчанию свойство активно — кнопка скрыта.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Текст метки

Заголовок

Тест, отображаемый в заголовке элемента управления. По умолчанию: `Файлы`.

Заполнитель

Текст, отображаемый в элементе управления, когда файлы не добавлены. По умолчанию: `Файлы отсутствуют`.

Заполнитель при перетаскивании

Текст, отображаемый в элементе управления при добавлении файла. По умолчанию: `Перетащите сюда файлы`.

Текст "Добавить дополнительный файл"

Текст отображаемый в ссылке, при нажатии которой вызывается добавление дополнительного файла. По умолчанию: + дополнительный.

Текст "Добавить основной файл"

Текст отображаемый в ссылке, при нажатии которой вызывается добавление основного файла. По умолчанию: + основной.

Текст для дополнительного файла

Текст, отображаемый в блоке добавления дополнительного файла перетаскиванием, и в заголовке списка дополнительных файлов. По умолчанию: Дополнительные файлы.

Текст для основного файла

Текст, отображаемый в блоке добавления основного файла перетаскиванием, и в заголовке списка основных файлов. По умолчанию: Основные файлы.

Свойство **Текст "Добавить дополнительный файл"**, **Текст "Добавить основной файл"**, **Текст для дополнительного файла** и **Текст для основного файла** доступны только при включенном свойстве **Разрешить дополнительные файлы**.

События

Клик по строке

Вызывается по нажатию на строку.

Перед визуализацией файла

Вызывается при формировании печатной версии файла со штампом электронной подписи.

Перед добавлением дополнительного файла

Вызывается перед добавлением дополнительного файла.

Перед добавлением комментария к версии файла

Вызывается перед добавлением комментария к версии файла.

Перед добавлением основного файла

Вызывается перед добавлением основного файла.

Перед загрузкой версии файла

Вызывается перед загрузкой версии файла.

Перед кликом по версии файла

Вызывается перед нажатием на версию файла.

Перед кликом по имени файла

Вызывается перед нажатием на имя файла.

Перед открытием диалога подписи

Вызывается перед открытием диалога подписания.

Перед открытием предпросмотра файла

Вызывается перед открытием предварительного просмотра файла.

Перед открытием списка версий

Вызывается перед открытием списка версий.

Перед открытием файла через WebDAV

Вызывается перед открытием файла с использованием WebDAV.

Перед открытием файлового меню

Вызывается перед открытием файлового меню.

Перед просмотром журнала подписей

Вызывается перед просмотром журнала подписей.

Перед разворачиванием

Вызывается перед разворачиванием списка файлов.

Перед сворачиванием

Вызывается перед сворачиванием списка файлов.

Перед скачиванием версии файла

Вызывается перед скачиванием версии файла.

Перед созданием цифровой подписи

Вызывается перед созданием цифровой подписи файла.

Перед удалением дополнительного файла

Вызывается перед удалением дополнительного файла.

Перед удалением комментария к версии файла

Вызывается перед удалением комментария к версии файла.

Перед удалением основного файла

Вызывается перед удалением основного файла.

После визуализации файла

Вызывается после формирования печатной версии файла со штампом электронной подписи.

После добавления дополнительного файла

Вызывается после добавления дополнительного файла.

После добавления комментария к версии файла

Вызывается после добавления комментария к версии файла.

После добавления основного файла

Вызывается после добавления основного файла.

После загрузки версии файла

Вызывается после загрузки версии файла.

После открытия предпросмотра файла

Вызывается после открытия предварительного просмотра файла.

После открытия файла через WebDAV

Вызывается после открытия файла с использованием WebDAV.

После разворачивания

Вызывается после разворачивания списка файлов.

После сворачивания

Вызывается после сворачивания списка файлов.

После скачивания версии файла

Вызывается после скачивания версии файла.

После создания цифровой подписи

Вызывается после создания цифровой подписи файла.

После удаления дополнительного файла

Вызывается после удаления дополнительного файла.

После удаления комментария к версии файла

Вызывается после удаления комментария к версии файла.

После удаления основного файла

Вызывается после удаления основного файла.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При просмотре журнала подписей

Вызывается после просмотра журнала подписей.

При сохранении файла в WebDAV

Вызывается при сохранении файла с использованием WebDAV.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Способы аутентификации

Элемент управления может быть размещён в справочнике сотрудников и профиле пользователя — там, где возможно получить информацию о сотруднике.

Чтобы определить набор отображаемых способов, в Web-клиенте создать экземпляр `AuthenticationExtensionFactory` и через него запрашивать доступные для определённой БД расширения аутентификации. Подробнее см. Настройка страницы авторизации.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Срок исполнения

Элемент управления Срок исполнения предназначен для установки дат начала и завершения, и длительности исполнения задания или группы заданий.



Рисунок 601. Элемент управления "Срок исполнения"

ЭУ является контейнером, в котором размещены три ЭУ:

- `startDate_timeOfPerformance` — дата и время начало задания; тип "Дата/время";
- `endDate_timeOfPerformance` — дата и время завершения задания; тип "Дата/время";
- `duration_timeOfPerformance` — длительность исполнения задания; тип "Число".

Для задания расчет длительности выполняется с учётом бизнес-календаря исполнителя, если это предусмотрено настройками вида карточки *Задание*.

ЭУ предназначен только для ввода значения.

Особенности настройки:

- Элемент управления предназначен для совместной работы с ЭУ "Панель выбора исполнителей", который должен быть добавлен на разметку.
- В настройках ЭУ `startDate_periodOfExecution`, `endDate_timeOfPerformance` и `duration_timeOfPerformance` нужно указать обработчик ***timeOfPerformanceToPerformersSync*** для события **После смены данных**. Обработчик предназначен только для синхронизации данных ЭУ "Панель выбора исполнителей" и "Срок исполнения", поэтому описанное актуально только для групп заданий. Обработчик установлен для событий **После получения даты завершения**, **После получения длительности**.
- **Необходимо самостоятельно настроить** вложенные ЭУ (`startDate_timeOfPerformance`, `endDate_timeOfPerformance`, `duration_timeOfPerformance`) для работы: как минимум, связать элементы управления с полями карточки *Задания* или *Группы заданий* (*дата начала*, *дата завершения*, *длительность*).

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Ориентация

Определяет способ отображения составляющих элемент управления.

- ***Вертикальная***. Входящие в ЭУ элементы будут размещаться параллельно.
- ***Горизонтальная***. Входящие в ЭУ элементы будут размещаться последовательно.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

После получения даты завершения

Вызывается после изменения значения даты завершения.

После получения длительности

Вызывается после изменения значения длительности.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении даты завершения

Вызывается при изменении значения даты завершения.

При получении длительности

Вызывается при изменении значения длительности.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Ссылка на карточку

Элемент управления `Ссылка на карточку` предназначен для выбора и отображения ссылки на определённую карточку.



После изменения разметок с данным ЭУ потребуется перезапуск службы Сервера Web-клиента, чтобы отобразить изменения.

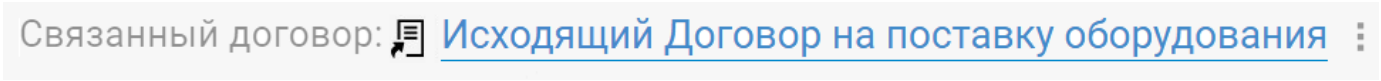


Рисунок 602. Элемент управления "Ссылка на карточку" в режиме просмотра

ЭУ позволяет:

- Открывать и изменять ссылку.
- Открывать файл карточки, на которую указывает ссылка.

Возможность добавления/удаления/изменения ссылки определяется доступностью данной операции по ролевой модели.



При удалении сильной ссылки удаляется сама карточка.

Выбор ссылки осуществляется в диалоговом окне, интерфейс по умолчанию которого представлен на следующем рисунке.

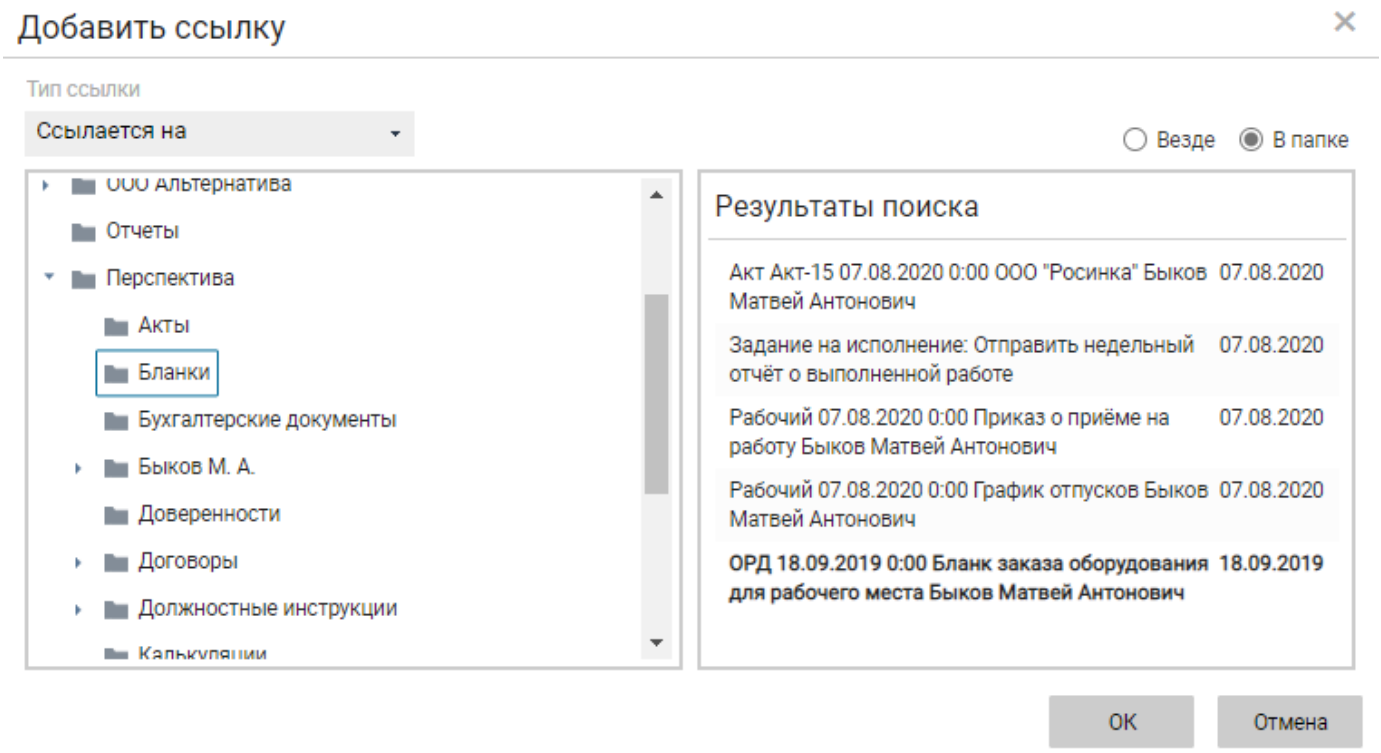


Рисунок 603. Выбор ссылки на карточку

Название карточки в ссылке по умолчанию формируется с помощью расширения Web-клиента. Если метод расширения для формирования не найден, в ссылке указывается дайджест карточки.



Ссылка может указывать на поисковый запрос. При поиске по карточке используется стандартный поисковый запрос *Ссылки*.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Показывать иконку

Включает или выключает отображение иконок у элемента управления.



Свойство доступно, только когда выбран поисковый параметр.

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, имеющее тип RefCardId.

Операция редактирования

Выбор операции для редактирования значения ЭУ. Если операция недоступна пользователю, то изменение ссылки будет недоступно. Если операция не выбрана, то возможность изменения ссылки не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Текст при отсутствии значения

Текст, отображаемый в ЭУ, когда ссылка не выбрана.

Локализуемый

Тип карточки для ссылки

Определяет типы карточек, ссылки на которые можно добавлять. Карточки других типов не отображаются при добавлении ссылки. Если значение не выбрано, для выбора будут доступны карточки всех типов.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Использовать список папок

Настройка списка папок, из которых пользователь сможет выбирать карточку для ссылки в режиме выбора "В текущей папке".

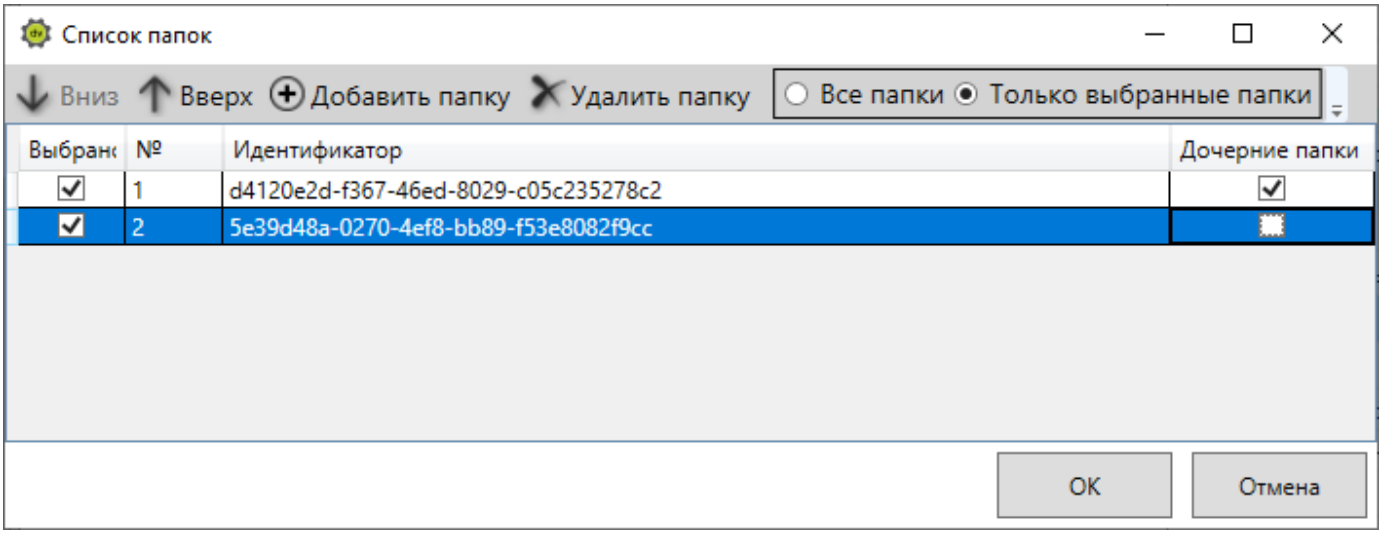


Рисунок 604. Список папок

Если список папок пуст, не отмечено ни одной добавленной папки или переключатель **Все папки / Только выбранные папки** в положении **Все папки**, пользователь сможет добавлять ссылки на карточки из любых доступных папок.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования.

Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Поисковый запрос

Открывает окно выбора поискового запроса, который будет использоваться при поиске карточки для ссылки в режиме выбора "Везде".

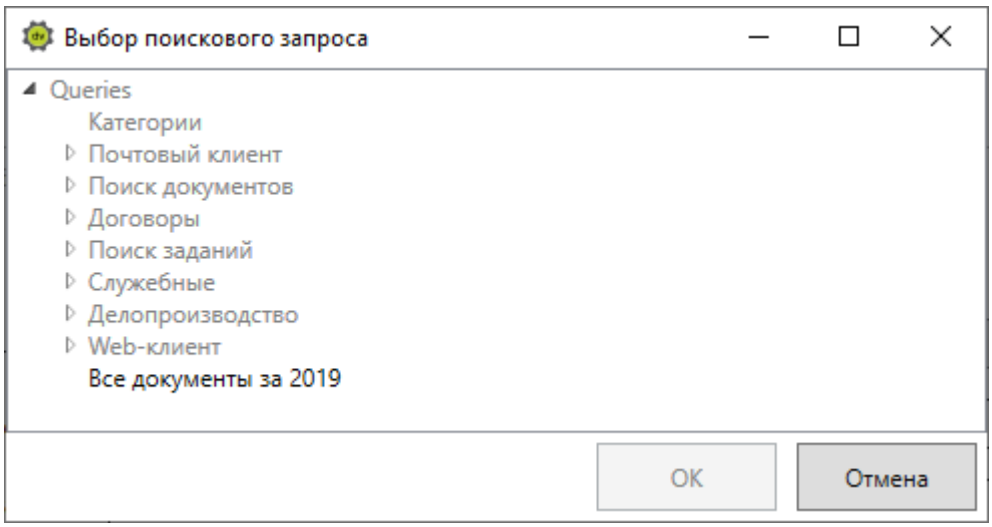


Рисунок 605. Выбор поискового запроса

По умолчанию используется собственный поисковый запрос Web-клиента: "Ссылки". Данный запрос позволяет искать карточки по дайджесту, автору, дате создания и изменения.

Поисковый параметр

Позволяет выбрать параметр поискового запроса, указанного в предыдущем свойстве.

Когда свойство настроено, при выборе карточки отображается специальное окно выбора с возможностью поиска карточки.

При формировании поиска для виртуальной папки появляется возможность искать по конкретной карточке.

Задержка поиска

Свойство доступно только при активации свойства Поисковый параметр.

Позволяет настроить задержку поиска после ввода поисковых символов. Значение по умолчанию: 500 мс.

Минимальное количество символов

Свойство доступно только при активации свойства Поисковый параметр.

Задаёт минимальное количество символов для поиска Значение по умолчанию: 3.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту*** — значение изменяется в отдельном окне, которое открывается нажатием на элемент управления. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.
- ***Редактирование*** — значение изменяется непосредственно в элементе управления. Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования*** — значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед закрытием окна

Вызывается перед закрытием окна выбора ссылки.

Перед открытием карточки

Вызывается при использовании команды открытия карточки по ссылке.

Перед открытием окна

Вызывается перед открытием окна выбора ссылки.

Перед открытием предпросмотра файла

Вызывается при использовании команды открытия окна предварительного просмотра файла карточки, на которую ведёт ссылка.

Перед удалением ссылки

Вызывается перед удалением ссылки.

После закрытия окна

Вызывается после закрытия окна выбора ссылки.

После открытия окна

Вызывается после открытия окна выбора ссылки.

После открытия предпросмотра файла

Вызывается после использования команды открытия окна предварительного просмотра файла карточки, на которую ведёт ссылка.

После смены данных

Вызывается после изменения содержимого элемента управления.

После удаления ссылки

Вызывается после удаления ссылки.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Ссылки

Элемент управления Ссылки предназначен для работы со списком ссылок карточки.



Элемент управления всегда работает в режиме редактирования по месту как в разметке редактирования, так и в разметке просмотра. Это значит, что добавленная ссылка сохранится всегда, даже если изменения в карточке не были сохранены.

При работе с элементом Ссылки в новой карточке и при редактировании существующей карточки функция добавления ссылки на новую карточку недоступна.

Вариант отображения ЭУ, а также его функциональные возможности зависят от свойства Режим отображения, которое может иметь значение ***по умолчанию*** или ***простой***.

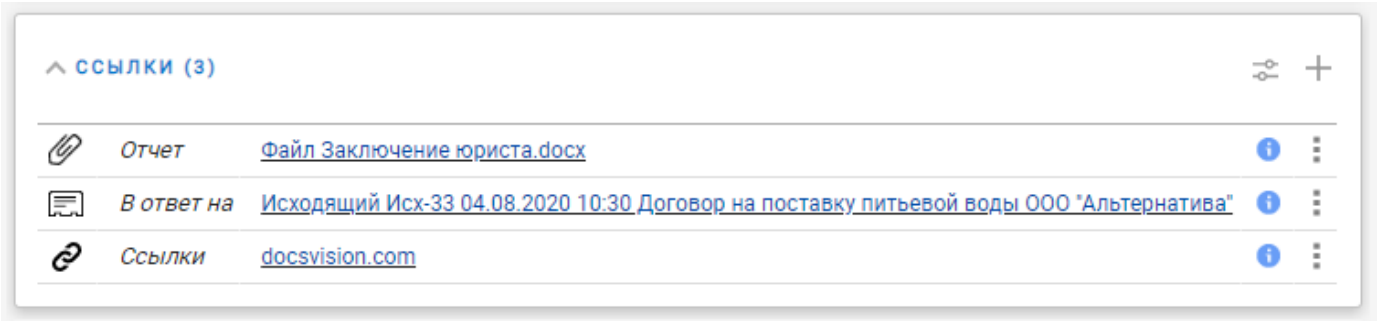


Рисунок 606. Пример отображения элемента управления "Ссылки" в режиме По умолчанию

ЭУ в режиме ***По умолчанию*** позволяет:

- Просматривать список существующих ссылок.
- Добавлять ссылки на существующие карточки с использованием формы поиска и выбора карточки.

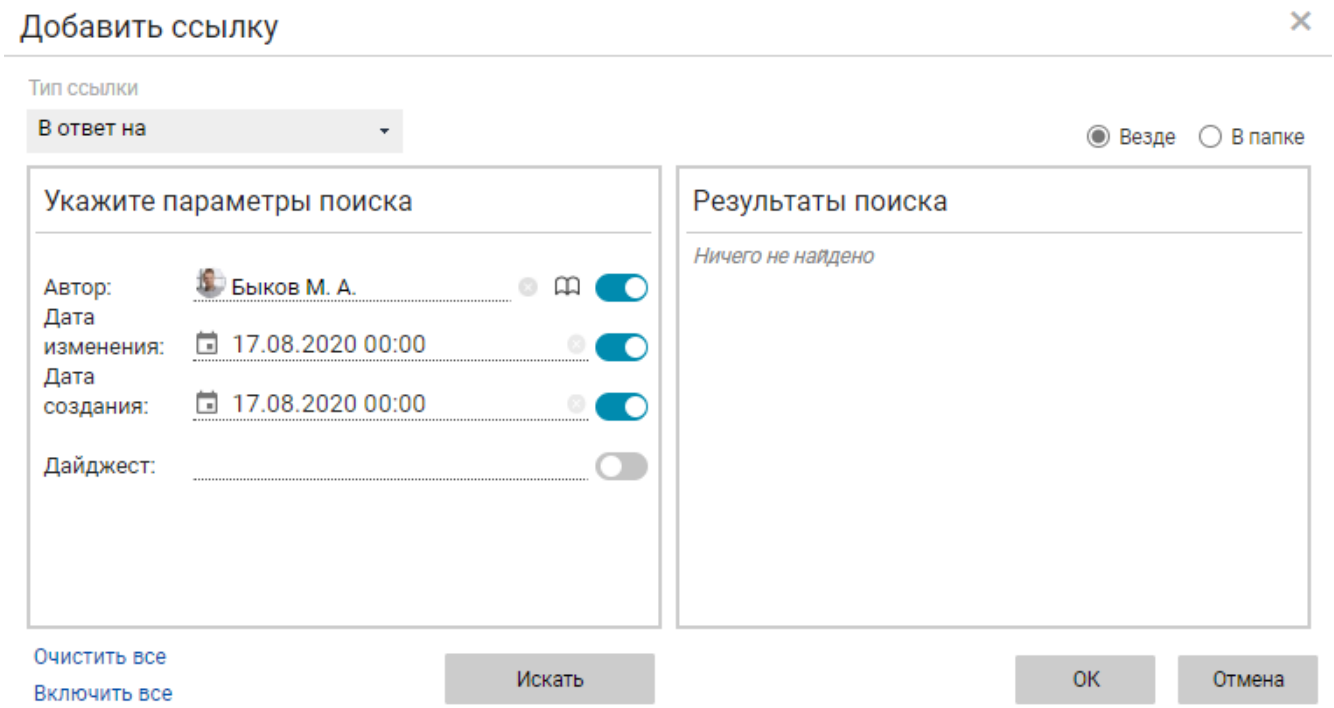


Рисунок 607. Выбор ссылки на карточку

- Создавать новые с добавлением ссылок.
- Добавлять гиперссылки.
- Добавлять файлы.
- Ограничивать возможность добавления ссылок.
- Просматривать информацию о ссылке и добавлять описание к ней.

Возможность выполнения определённой операции со списком ссылок определяется доступностью соответствующей операции согласно ролевой модели.

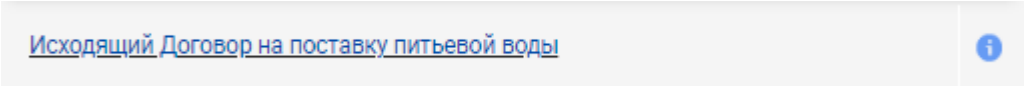


Рисунок 608. Пример отображения элемента управления "Ссылки" в режиме "Простой"

ЭУ в режиме ***Простой*** позволяет:

- Просматривать список существующих ссылок.
- Просматривать информацию о ссылке и добавлять описание к ней.



Описание ссылки в таблице ссылок, а также справка о ссылке могут быть настроены. См. пункт Настройка отображения информации о ссылке в элементе управления Ссылки.

> К карточкам, ссылки на которые содержатся в ЭУ, могут применяться групповые операции. См. пункт Настройка групповых операций в элементе управления Ссылки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Вид дополнительной информации

Определяет способ формирования справки о ссылке (открывается при нажатии кнопки):

- ***Только о ссылке***. В справке будут отображаться: автор ссылки, тип ссылки и описание.
- ***Только о карточке***. В справке будет отображаться информация о карточке, на которую указывает ссылка. Формат отображения информация о карточке определяется разметкой с режимом "Справка".
- ***О карточке и о ссылке***. В справку будут включены информация о ссылке и о карточке (настраивается разметкой), на которую указывает ссылка.
- ***Не показывать***. Не показывать кнопку для открытия справки.

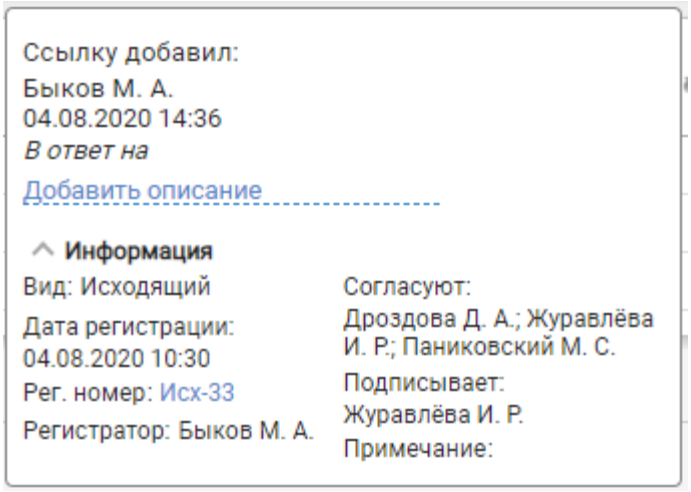


Рисунок 609. Отображение информации о ссылке и о карточке в окне справки

Заголовок

Текст, который отображается над ссылками. Заголовок выделяется жирным шрифтом. Рядом с заголовком отображается число ссылок.

Локализуемый

Колонки

Определяет набор колонок отображаемых по умолчанию в таблице ссылок. Если флаг в свойстве **Разрешить настройку столбцов** активен, пользователям будет предоставлена возможность изменить набор отображаемых столбцов.

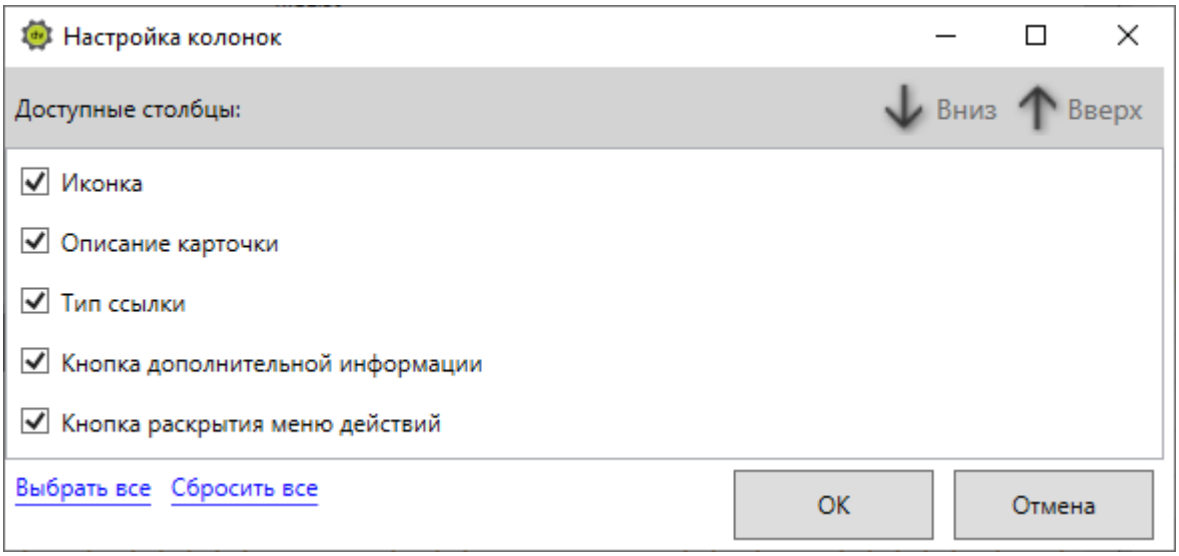


Рисунок 610. Настройки отображения колонок в таблице ссылок

Метод для описания

Определяет способ формирования информации о ссылке в таблице ссылок:

- ***Стандарт***. В таблице будет отображаться дайджест карточки.
- ***Расширение***. В таблице будет отображаться информация о карточке, возвращаемая серверным расширением Web-клиента (в свойстве `Наименование метода для описания` должно быть указано название метода, предоставляемого расширением).
- ***Разметка***. В таблице будет отображаться информация о карточке, формируемая разметкой с режимом выбора ***Описание***. Если у карточки, на которую указывает ссылка, отсутствует разметка с режимом выбора ***Описание***, в ссылке будет использоваться дайджест карточки.



Настройка применяется только к ссылкам на карточки.

Наименование метода для описания

Название метода расширения Web-клиента, возвращающего описание ссылки.

Для карточек *Исходящий*, *Входящий*, *Заявка*, *ОРД*, *Рабочий*, *Служебная записка*, *Договор*, *Акт* и *Доп. соглашение* предоставляется стандартный метод для описания: `DocumentManagement_Default`.

Разрешить настройку столбцов

Разрешает или запрещает пользователям самостоятельно формировать список столбцов, отображаемых таблице ссылок.

Режим отображения

Определяет способ работы с элементом управления:

- ***По умолчанию***. Обычный способ работы с предоставлением полного набор команд управления ссылками, в соответствии с правами пользователя.
- ***Простой***. Режим работы, предназначенный только для отображения ссылок: команды редактирования будут недоступны, а тип и значок ссылки скрыты.

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее ссылку на карточку *Список ссылок на карточки*.

Операция редактирования

Выбор операции для редактирования значения элемента управления. Если операция недоступна пользователю, то изменения списка ссылок или добавления комментария к ссылке будет недоступно. Если операция не выбрана, то возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Хранить сильную ссылку

Определяется тип ссылки на карточку. Если флаг установлен, то используется сильная ссылка.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Добавить гиперссылку

Настройка параметров добавления или просмотра ссылки на URL-адрес.

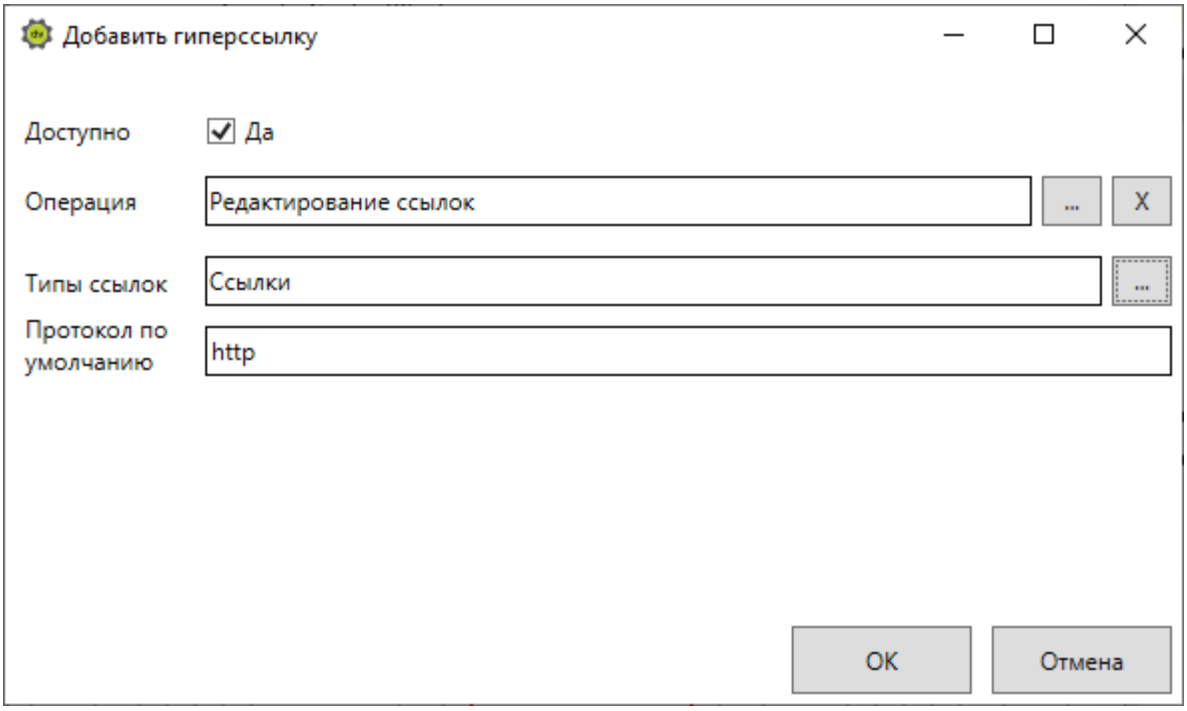


Рисунок 611. Настройки добавления гиперссылки

Можно включить или отключить возможность добавления новой гиперссылки, при этом соответствующая кнопка будет показана или скрыта. Также можно указать операцию редактирования, при которой добавление гиперссылок будет разрешено. В поле *Протокол по умолчанию* нужно указать протокол, который будет добавляться к адресу, введённому пользователем, если в адресе не указан конкретный протокол (`http`, `https`, `ftp` и т.п.). Добавление ссылок будет доступно всем, если операция редактирования не указана.

Добавить ссылку

Настройка параметров добавления или просмотра ссылки на существующую карточку.

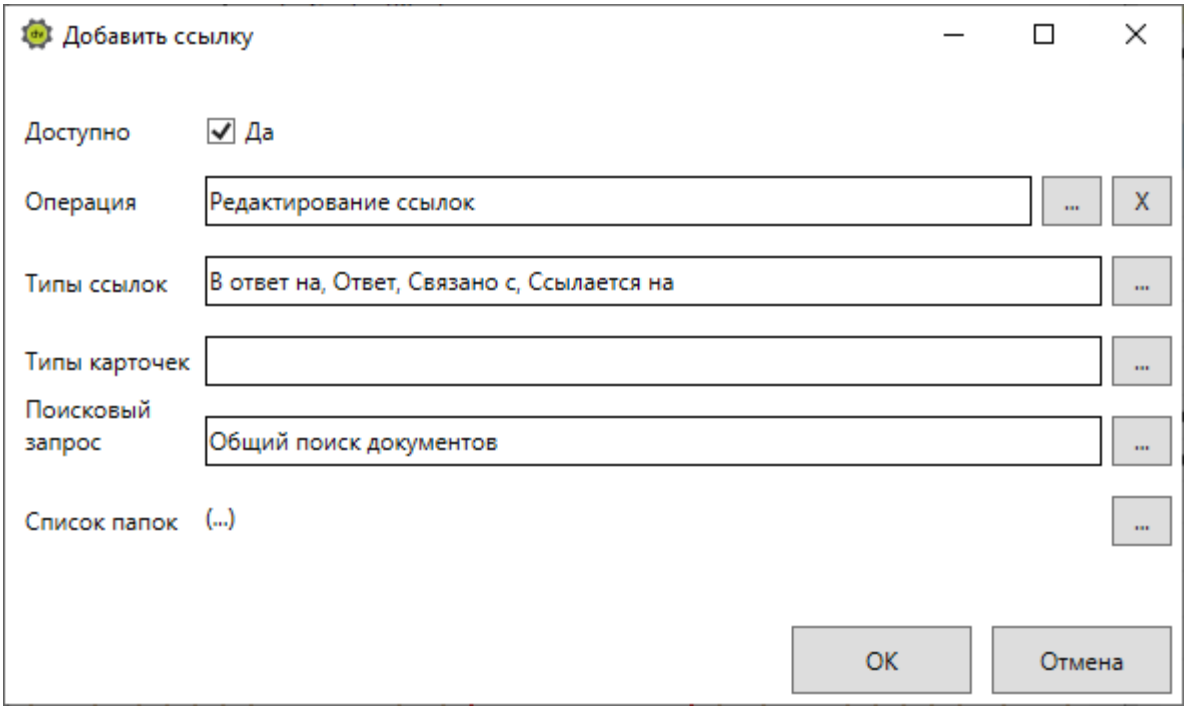


Рисунок 612. Настройки добавления ссылки

- Флаг **Доступно**. Позволяет включить или отключить возможность добавления новой ссылки, при этом соответствующая кнопка будет показана или скрыта.
- Можно также указать *Операцию* редактирования, при которой добавление ссылок будет разрешено.

  Если операция редактирования не указана, добавление ссылок будет доступно всем.

- А также возможные типы ссылок.

  В поле *Типы ссылок* отображаются только те ссылки, которые добавлены в области *Выбрано*. Чтобы выбрать разрешённые типы, нужно нажать на кнопку ... в появившемся диалоговом окне перенести их из колонки *Доступно* в колонку *Выбрано*.

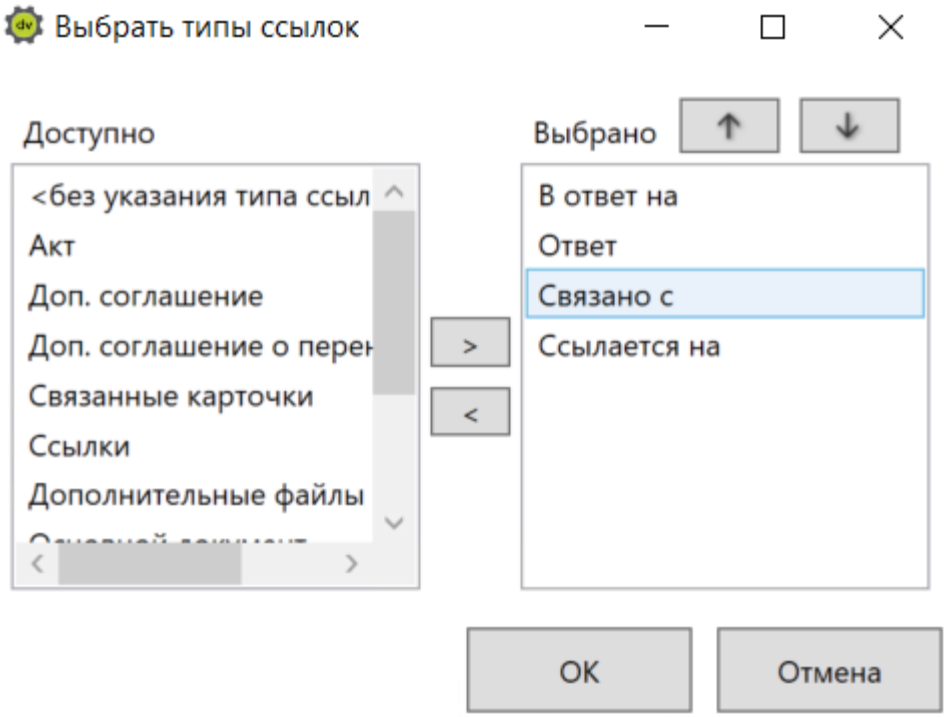


Рисунок 613. Выбор разрешённых типов ссылок

Диалоговое окно такого типа открывается в полях *Типы ссылок* и *Типы карточек* при редактировании свойства `Добавить ссылку`, а также в поле *Типы ссылок* при редактировании свойства `Добавить гиперссылку`.

- Настройка `Типы карточек` позволяет установить ограничение по типам карточек, на которые можно ссылаться.

  Можно выбрать один или несколько типов карточек, на которые разрешено ссылаться.

  Если типы карточек не выбраны, допускается добавление ссылок на карточки любых типов.

- *Поисковый запрос*. Открывает окно выбора поискового запроса. Выбранный поисковый запрос будет использоваться в режиме поиска карточек, при добавлении ссылок в режиме выбора ***Везде***.

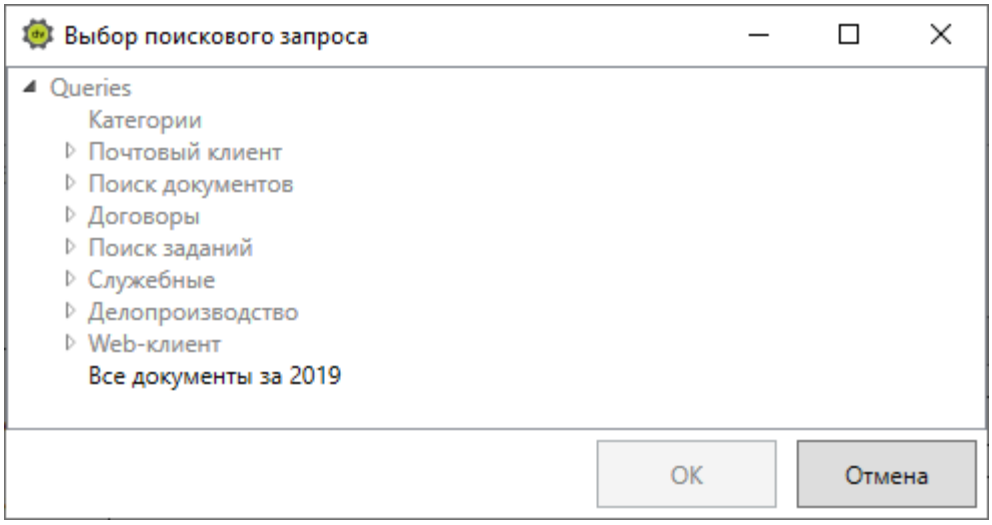


Рисунок 614. Выбор поискового запроса

По умолчанию используется собственный поисковый запрос Web-клиента: "Ссылки". Данный запрос позволяет искать карточки по дайджесту, автору, дате создания и изменения.

- *Список папок*. При нажатии открывается окно добавления папок (по идентификаторам). Из выбранных папок пользователь сможет выбирать карточки для добавления ссылок в режиме выбора ***В текущей папке***.

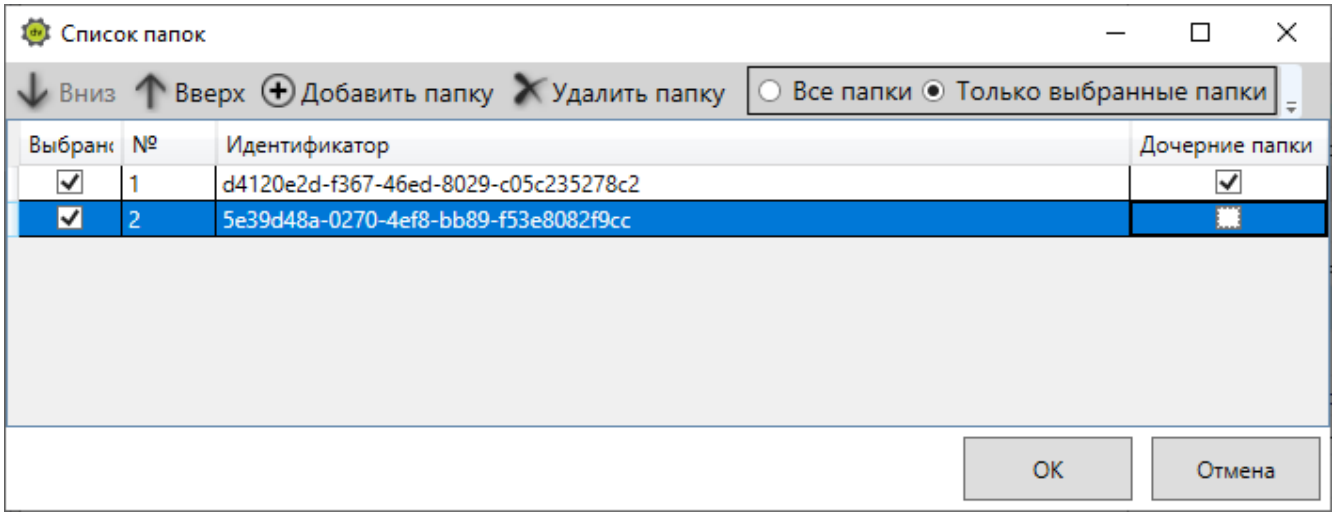


Рисунок 615. Список папок

Если список папок пуст, не отмечено ни одной добавленной папки или переключатель **Все папки / Только выбранные папки** в положении **Все папки**, пользователь сможет добавлять ссылки на карточки из любых доступных папок.

Добавить файл

Настройка параметров добавления или просмотра ссылки на файл.

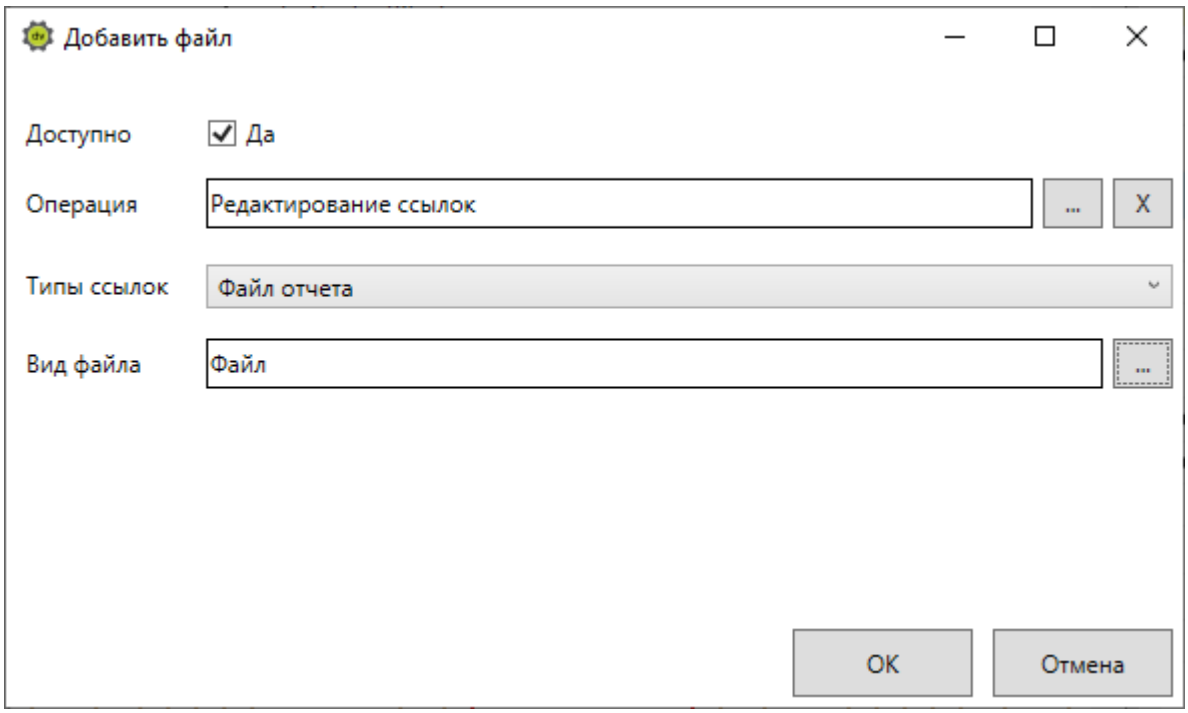


Рисунок 616. Настройки добавления файла

Добавленные через ЭУ файлы сразу сохраняются на сервер. При отказе от сохранения карточки (нажатии кнопки **Отменить**) файл остаётся в карточке, так как уже был сохранён на сервере. Отменяются только те данные, которые не были отправлены на сервер.

Диалоговое окно настройки добавления файла содержит:

- Флаг `Доступно`. Включает или выключает возможность добавления файла с помощью элемента управления "Ссылки".
- Поле *Операция*. Операция редактирования, которая должна быть доступна пользователю для разрешения добавления файла.
- Поле *Типы ссылок*. Тип ссылки, с которой будут добавляться файлы.
- Поле *Вид файла*. Вид карточки, которая будет создаваться для добавляемого файла.

  Файл (или несколько) можно добавить с помощью кнопки или "перетаскиванием" из файловой системы. При добавлении файлов, для каждого файла будет создана карточка вида *Вид файла*.

Пользователь может открыть файл, щелкнув по названию файла в списке ссылок или карточку файла из меню ⋮. Пользователь также может удалить ссылку на файл или сам файл.

Файл может быть удалён, если на него ссылается только данная карточка.

- Если карточка файла содержит только один файл, то будет удалена его карточка и ссылка на неё,
- Если карточка файла содержит несколько файлов, из карточки файла будет удалён только выбранный файл.



Когда задано свойство **Добавить файл**, настройки из свойства **Показывать файлы для ссылок** не будут действовать, т.е. файлы для ссылок отображаться не будут, см. подробнее Показывать файлы для ссылок.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Загружать синхронно (шт)

Определяет максимальное количество элементов списка, при котором должна использоваться синхронная загрузка данных. Если количество элементов больше указанного, то все данные будут загружаться асинхронно.

Изменение

Выбор операции редактирования, при которой возможно изменение списка ссылок.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.

- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Отображать раскрытым

Определяет начальное состояние блока элемента управления. Когда флаг установлен, список ссылок отображается. Когда флаг не установлен — список ссылок скрыт, отображается только заголовок и количество ссылок.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Показывать файлы для ссылок

Настройка типов ссылок, для которых должен действовать особый режим отображения. При особом режиме редактирования в ЭУ отображается не карточка, на которую ссылка, а её основные и дополнительные файлы.

Пользователь может открыть файл, щелкнув по названию файла в списке ссылок или карточку файла из меню ⋮. Пользователь также может удалить ссылку на карточку или файл из карточки, на которую указывает ссылка.

- Если настройка **Показывать файлы для ссылок** не задана для карточки, то возможность скачать файл из такой карточки будет недоступна.
- Настройка **Показывать файлы для ссылок** блокируется свойством **Добавить файл**. При заданном свойстве **Добавить файл** отображаться будут отображаются только ссылки, добавленные с типом, указанным в свойствах **Создать ссылку**, **Добавить ссылку**, **Добавить гиперссылку** или **Добавить файл**.

Создать ссылку

Настройка параметров создания карточки с добавлением ссылки.

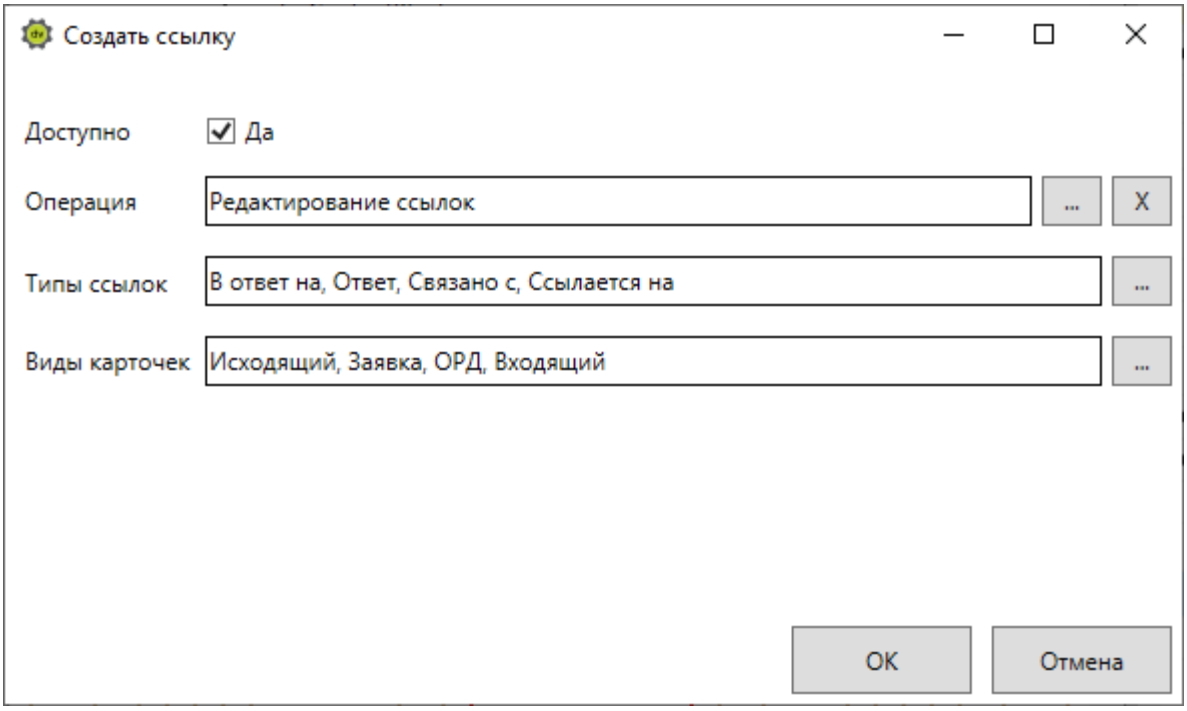


Рисунок 617. Настройки создания карточки с добавлением ссылки

Можно включить или отключить возможность создания ссылки с созданием карточки, при этом соответствующая кнопка будет показана или скрыта.

Также можно указать операцию редактирования, при которой создание ссылок будет разрешено, а также доступные для выбора типы ссылок. Если операция редактирования не указана, то добавление ссылок будет доступно всем.

Настройка `Виды карточек` позволяет указать виды карточек, которые можно создавать по ссылке:

- Можно выбрать один или несколько видов карточек, на которые разрешено ссылаться.
- Если вид карточек содержит подвиды, то при установленном в данной настройке флаге `С дочерними видами`, они также будут доступны для создания карточек по ссылке.

  Если вид или виды карточек не выбраны, то добавление ссылки с созданием карточки будет невозможно.

  Функция создания карточки с добавлением ссылки недоступна в разметке создания карточки.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента

управления.

Удаление

Выбор операции редактирования, при которой возможно удаление ссылки.

События

Данные загружены

Вызывается, когда данные ссылки на файл загружены.

Клик по строке

Вызывается при нажатии на строку ссылки на файл.

При открытии диалога добавления ссылки на новую карточку

Вызывается при открытии диалога выбора вида создаваемой карточки.

При открытии диалога выбора существующей карточки

Вызывается при открытии диалога поиска и выбора существующей карточки.

При подтверждении выбора существующей карточки

Вызывается при подтверждении добавления карточки в диалоге выбора, перед фактическим добавлением ссылки.

Перед загрузкой ссылки на файл

Вызывается перед загрузкой ссылки на файл.

Перед закрытием тултипа с информацией о карточке

Вызывается перед закрытием окна (открытого кнопкой) с информацией о карточке.

Перед кликом по имени ссылки

Вызывается перед нажатием на имя ссылки на файл.

Перед открытием карточки

Вызывается перед открытием карточки из списка ссылок.

Перед открытием меню ссылки

Вызывается перед открытием меню ссылки.

Перед открытием предпросмотра карточки

Вызывается перед открытием предварительного просмотра карточки.

Перед открытием тултипа с информацией о карточке

Вызывается перед открытием (кнопкой ) окна с информацией о карточке, на которую указывает ссылка.

Перед разворачиванием

Вызывается перед разворачиванием ЭУ кнопкой ˅.

Перед редактированием комментария к ссылке

Вызывается перед редактированием комментария к ссылке в окне информации о карточке.

Перед сворачиванием

Вызывается перед сворачиванием *Блока* кнопкой ^.

Перед удалением ссылки

Вызывается перед удалением ссылки.

Перед удалением файла

Вызывается перед удалением ссылки на файл.

После добавления ссылки на существующую карточку

Вызывается после успешного добавления ссылки и закрытия диалога выбора.

После загрузки ссылки на файл

Вызывается после добавления ссылки на файл.

После изменения выделения

Вызывается после изменения выделения.

После открытия меню ссылки

Вызывается после открытия меню ссылки.

После переключения режима групповых операций

Вызывается после переключения режима групповых операций.

После разворачивания

Вызывается после разворачивания ЭУ кнопкой ∨.

После редактирования комментария к ссылке

Вызывается после сохранения комментария (в окне информации о карточке) к ссылке.

После сворачивания

Вызывается после сворачиванием ЭУ кнопкой ∧.

После удаления ссылки

Вызывается после удаления ссылки.

После удаления файла

Вызывается после удаления файла.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Настройка отображения информации о ссылке

Элемент управления Ссылки состоит из двух блоков: **строки меню** и **таблицы ссылок**.

В **таблице ссылок** отображаются:

- Значок вида ссылки.
- Информация о карточке.
- Тип ссылки.
- Кнопка информации о ссылке.
- Кнопка меню команд.

В столбце *Информация о карточке* по умолчанию отображается дайджест

карточки. Если требуется, вместо дайджеста можно вывести другую информацию о карточке.



Рисунок 618. Элемент управления "Ссылки" со стандартным методом получения описания ссылок

За выбор способа формирования строки с информацией о карточке отвечает свойство **Метод для описания** элемента управления Ссылки. Данная настройка может принимать значения:

- ***Стандарт***. В информации о карточке отображается дайджест.
- ***Расширение***. В информации о карточке отображается значение, предоставляемое специальным расширением Web-клиента.
- ***Разметка***. В информации о карточке отображается значение, формируемое специальной разметкой. Если разметка для формирования ссылки отсутствует, в ссылке будет отображаться дайджест карточки.

Использование дайджеста карточки в ссылке

1. Создайте новую разметку просмотра в локации, соответствующей виду/типу карточек, на которые есть ссылки в таблице ссылок.

   Разметка настраивается именно для карточек, на которые указывают ссылки. Если у вида/типа карточек не настроена такая разметка, в таблице ссылок строки с информацией о карточках данного вида будут пустыми (если свойство **Метод для описания** в значении ***Разметка***).

2. Добавьте в разметку требуемые элементы управления и настройте их на получение данных из карточки.

   Поддерживаются следующие типы дочерних элементов управления: Строка, Метка, Дата/время, Флаг, Раскрывающийся список, Число, Партнер, Подразделение, Подразделение контрагента, Сотрудник, Сотрудники, Строка конструктора справочников, Контейнер строка.

Например, добавьте ЭУ Контейнер строка с дочерними элементами управления:

Строка, связанный с секцией *Основная информация*, полем *Название*.

Сотрудник, связанный с секцией *Основная информация*, полем *Автор*.

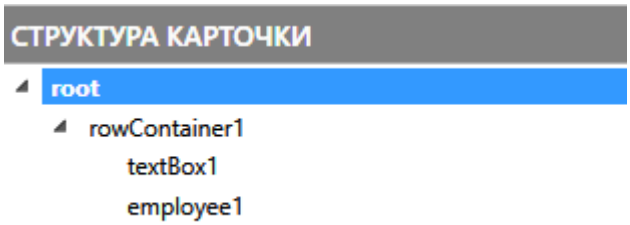


Рисунок 619. Пример структуры разметки блока информации о карточке

3. Сохраните разметку.
4. Добавьте в данную локацию новое условие выбора со следующими свойствами:
   - Режим: ***Описание***.
   - Состояние: ***Любое***.
   - Роль: ***Любое***".
   - Тип устройства: ***Любое***.
   - Разметка: созданная разметка.
5. Откройте разметку просмотра карточки с элементом управления Ссылки, в котором информации о ссылках должны формироваться разметкой.
6. В настройках ЭУ Ссылки измените значение свойства **Метод для описания** на ***Разметка***.
7. Сохраните разметку.

Откройте для просмотра карточку, разметка которой была настроена ранее. Описание ссылок в карточке будет содержать название и автора карточки.

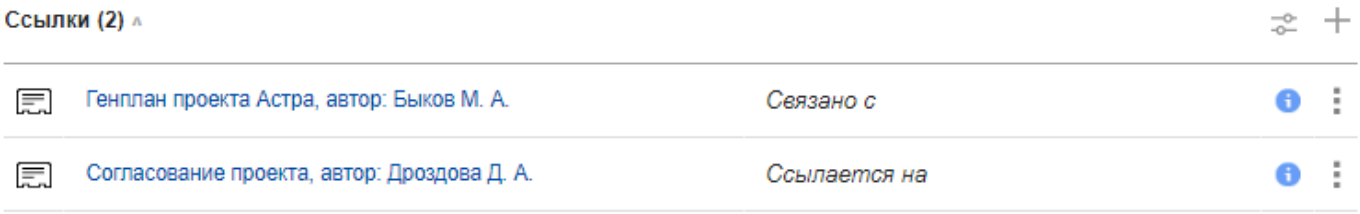


Рисунок 620. Элемент управления Ссылки с методом получения описания ссылок "Разметка"

Настройка блока информации о карточке в справке о ссылке

Аналогичным образом может быть настроена разметка блока информации о карточке в справке о ссылке (открывается при нажатии кнопки), со следующими отличиями:

- Разметка блока информации о карточке может быть более сложной: не требуется помещать все элементы внутрь ЭУ `*Контейнер строка*`.
- При добавлении условия выбора разметки следует указать Режим: ***Справка***;
- В ЭУ `Ссылки` нужно настроить свойство **`Вид дополнительной информации`**: должно быть выбрано значение ***О карточке и о ссылке*** или ***Только о карточке***.

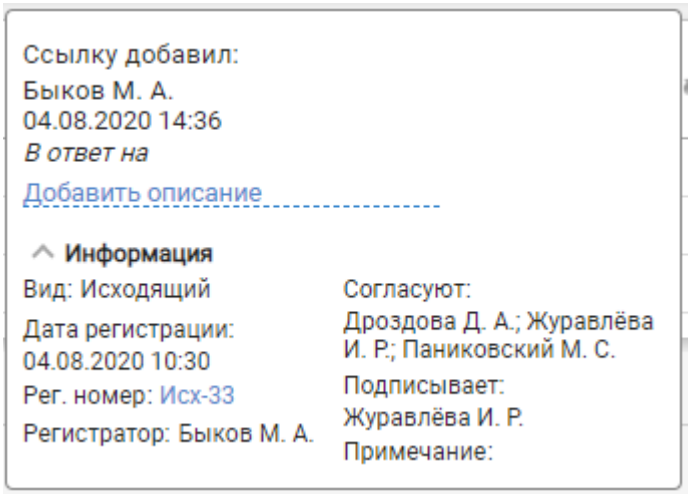


Рисунок 621. Отображение информации о ссылке и о карточке в окне справки

Настройка групповых операций в элементе управления "Ссылки"

К карточкам, ссылки на которые отображаются в ЭУ `Ссылки`, могут применяться групповые операции[0].

Чтобы применить групповые операции к карточкам по ссылке:

1. Откройте разметку с ЭУ `Ссылки`.
2. Добавьте в список операций (элемент `linksBatchOperations`, вложенный в элемент управления `Ссылки`) следующие элементы:

- "Сообщение об отсутствии доступных операций",
- "Состояние выделения карточек папки",
- "Групповая операция / Делегирование".

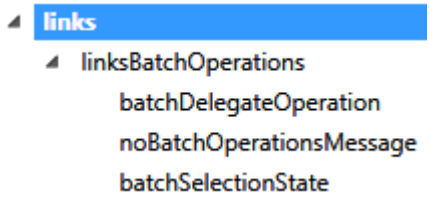


Рисунок 622. Элемент управления "Ссылки" со списком операций в редакторе разметок

3. Сохраните разметку.

Откройте для просмотра карточку, разметка которой была настроена.

Элемент управления Ссылки будет содержать кнопку переключения в режим групповых операций: .

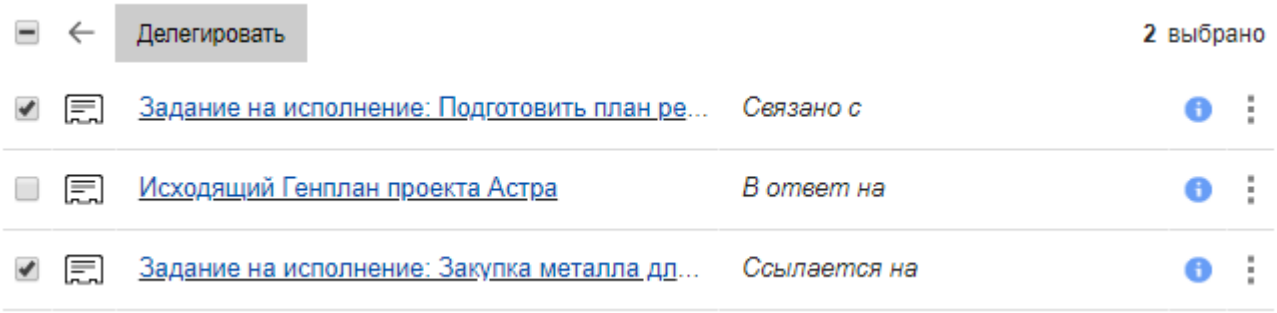


Рисунок 623. Элемент управления Ссылки в режиме групповых операций

Таблица исполнения

Элемент управления Таблица исполнения предназначен для отображения дерева заданий в виде списка.

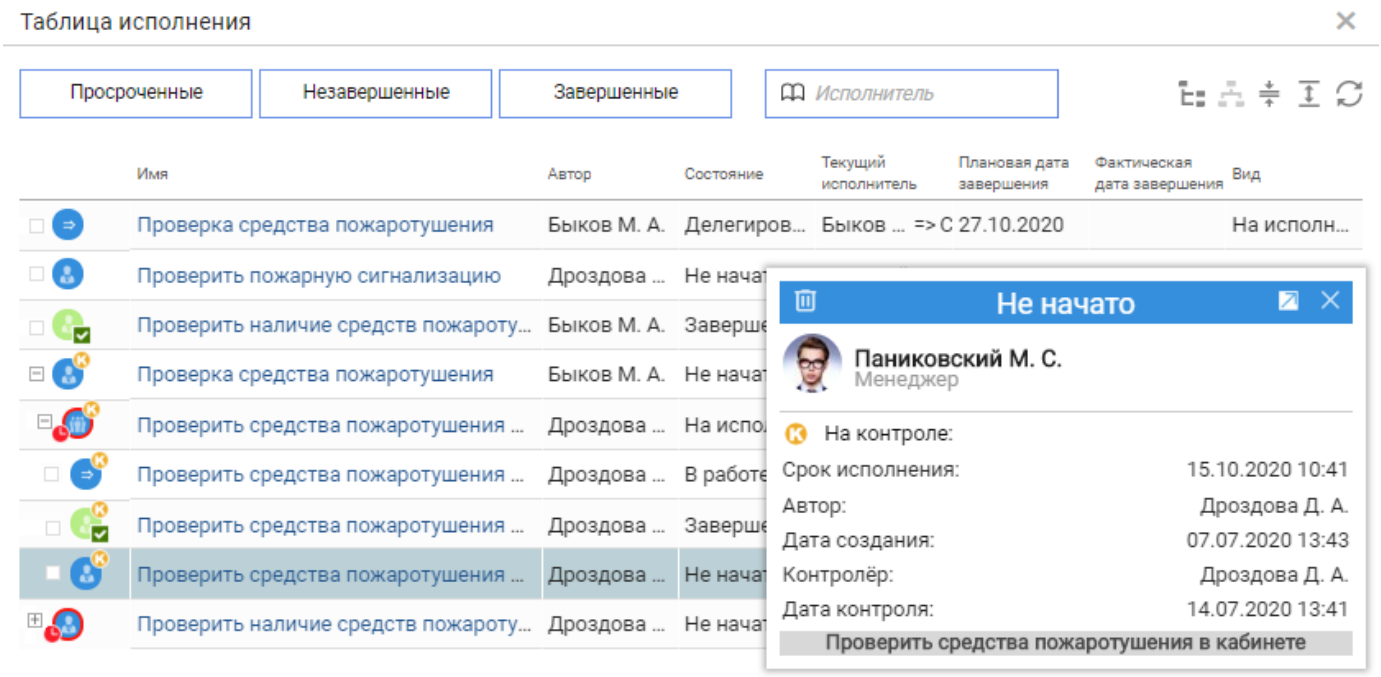


Рисунок 624. Элемент управления "Таблица исполнения" в карточке документа

Элемент управления поддерживает два варианта отображения:

1. В карточке отображается кнопка, при нажатии которой будет открыто окно просмотра таблицы исполнения.
2. В карточке сразу отображается таблица исполнения.

Элемент управления предоставляет функции:

- Фильтрация списка по текущему статусу задания, и поиск заданий по исполнителю.
- Переход к заданию/группе заданий из списка по нажатию на ссылку с названием задания/ГЗ.
- Просмотр информации по отдельному заданию/группе заданий по нажатию на иконку задания/ГЗ.
- Удаление задания/группы заданий из окна просмотра информации по заданию при наличии прав.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Показывать развернутым

Определяет уровень узлов дерева заданий, отображаемых по умолчанию:

`0` Показывать все узлы (значение по умолчанию).

`1` Показывать только узлы первого уровня (вложенности).

`2` Показывать узлы первого и второго уровня.

`3` Показывать узлы первого, второго и третьего уровня, и т. д.

Режим отображения

Определяет вариант отображения ЭУ в карточке:

- ***Кнопка***. В карточку добавляется кнопка для открытия модального окна со списком исполнения.
- ***Разметка***. В карточку добавляется непосредственно список исполнения.

Текст кнопки

Текст кнопки открытия таблицы исполнения. Используется в режиме отображения ***Кнопка***.

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее ссылку на карточку *Список ссылок на карточки заданий*.

Виды заданий для отображения

Выбор видов заданий, отображаемых в дереве исполнения. Если значение не указано, то в дереве исполнения отображаются задания и группы заданий всех видов.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Сортировка

Выбор сортировки по умолчанию для элемента управления.

Доступные виды сортировки:

- Дата отправки (сортировка по умолчанию).
- Срок исполнения.
- Имя.
- Автор.
- Состояние.
- Текущий исполнитель.
- Вид.

Видимые колонки таблицы

Список колонок с данными карточек, которые должны отображаться в таблице исполнения на устройствах с большим экраном (компьютер, ноутбук, планшет). На мобильных устройствах отображается только столбец Имя.

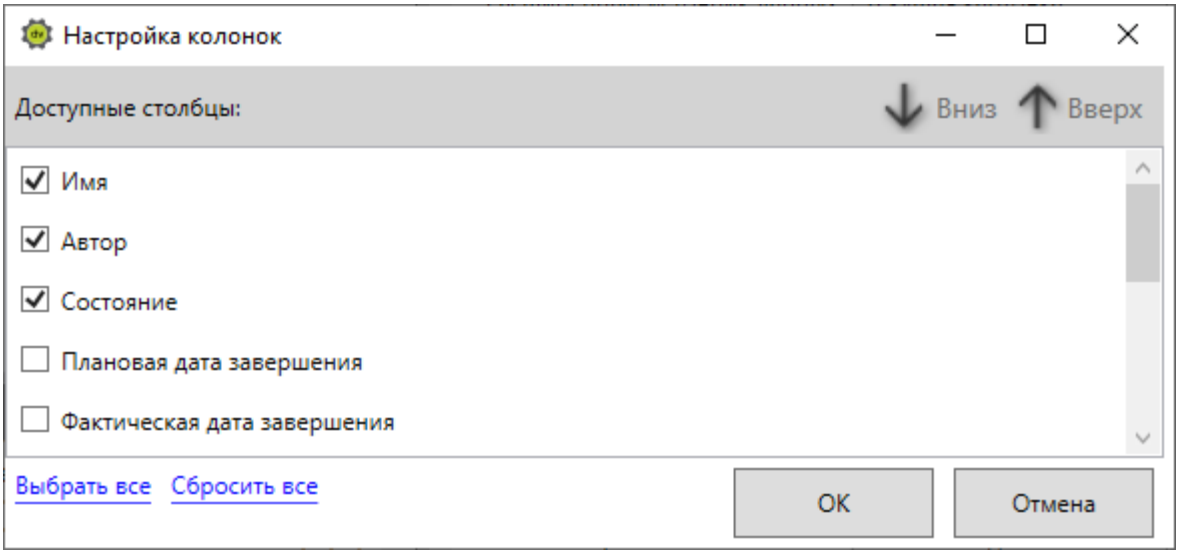


Рисунок 625. Выбор отображаемых колонок таблицы исполнения

Список колонок:

- *Имя*.
- *Автор*.
- *Состояние*.
- *Плановая дата завершения*.
- *Фактическая дата завершения*.
- *Назначенный исполнитель*.
- *Текущий исполнитель*.
- *Контролёр*.
- *Ответственный исполнитель*.
- *Дата отправки*.
- *Описание*.
- *Вид*.
- *Комментарий / Отчёт*.

Текущий исполнитель не отображается, если задание не отправлено.

В колонке *Комментарий / Отчёт* будет выводиться отчёт к завершению задания или последний комментарий, если задание не завершено. Выводится только текстовый комментарий/отчет. Файлы отчета в колонке не отображаются.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимальная ширина таблицы.

Адаптивный

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Просмотр полного дерева

Выбор операции, которая должна быть доступна пользователю для возможности просмотра полного дерева заданий.

По умолчанию у пользователя в заданиях и группах заданий в таблице исполнения отображаются только текущее задание и его подчиненные задания. Чтобы просмотреть полное дерево исполнения (с родительскими заданиями) пользователю нужно нажать кнопку переключения режима, которая скрыта, если пользователю недоступна операция, указанная в данной настройке. Если операция не указана, то кнопка переключения режима отображения таблицы исполнения будет доступна всегда.

Режим открытия

Определяет способ открытия ссылки:

- ***Текущая вкладка***. Ссылка будет открыта в текущей вкладке.
- ***Новая вкладка***. Ссылка будет открыта в новой вкладке.
- ***Новое окно браузера***. Ссылка будет открыта в новом окне веб-браузера.

Режим отображения группы из 1 задания

Определяет вариант отображения группы заданий, содержащей единственное задание в таблице исполнения:

- ***Отображать оба***. Должна быть показаны группа заданий и её единственное задание.
- ***Группа***. Должна отображаться только группа заданий.
- ***Задание***. Должно отображаться только задание группы.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед удалением строки

Вызывается при использовании функции удаления задания/группы заданий.

После удаления строки

Вызывается после удаления задания/группы заданий.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Файловая панель

Элемент управления Файловая панель предназначен для работы с файлами задания и группы заданий.

Одновременное размещение на разметке элементов управления "Файловая панель" и "Файловая панель согласования" в данный момент не поддерживается и может привести к ошибкам при открытии файлов на предпросмотр.

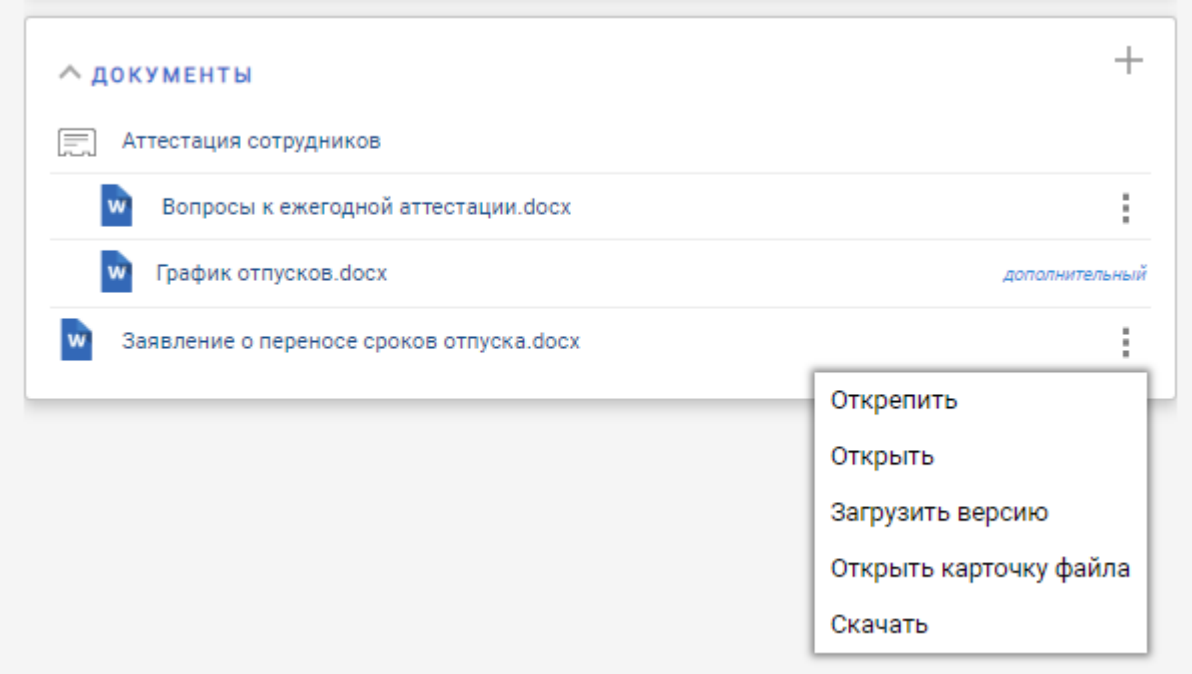


Рисунок 626. Элемент управления "Файловая панель"

ЭУ предоставляет следующие возможности:

- Добавлять новые файлы и новые версии файлов.
- Просматривать файлы родительского документа.
- Открывать файлы офисного формата во внешнем приложении.
- Просматривать содержимое файла в окне предварительного просмотра: собственная реализация или с использованием компонента Предпросмотр файла.

Возможности ЭУ зависят от типа/вида карточки, её состояния и роли пользователя.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Загружать синхронно (шт)

Определяет максимальное количество элементов списка, при котором должна использоваться синхронная загрузка данных. Если количество элементов больше указанного, то все данные будут загружаться асинхронно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При загрузке версии файла

Вызывается при загрузке версии файла.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

При сохранении файла в WebDAV

Вызывается при сохранении файла с использованием WebDAV.

Ход исполнения

Элемент управления Ход исполнения предназначен для отображения информации об исполнении подчиненных заданий в карточке *Группы заданий*.

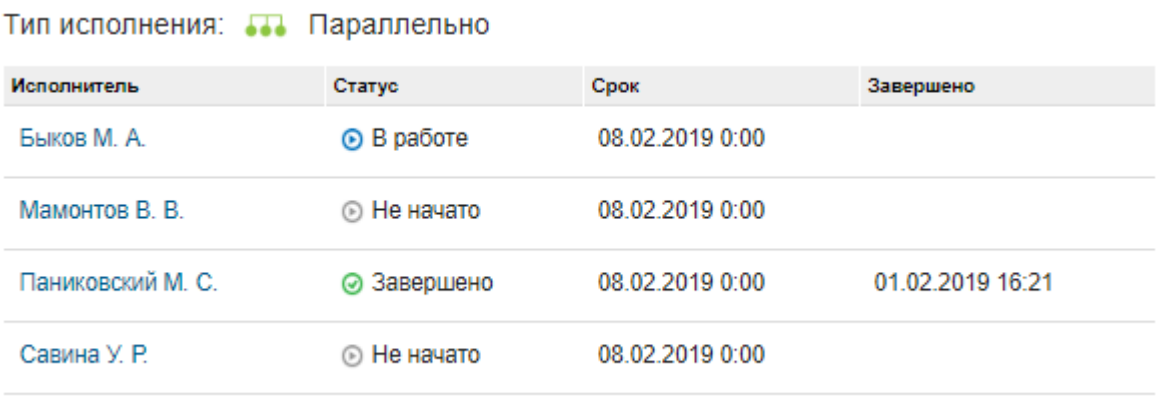


Рисунок 627. Элемент управления "Ход исполнения"

ЭУ содержит следующую информацию:

- Тип исполнения группы заданий.
- Список заданий данной группы с указанием исполнителя, статуса, срока завершения и даты завершения задания.

Реализация элемента управления переведена на React в Web-клиенте, начиная с версии 6.1, сборка 6478. Для работы ЭУ в стандартных разметках изменений не требуется. Если элемент

управления использовался в собственных разметках, его необходимо удалить и добавить повторно.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Экспорт подписей

Элемент управления `Экспорт подписей` добавляет в элемент "Список файлов" функцию выгрузки подписанных файлов с электронными подписями. При использовании данной функции на устройство пользователя будет сохранён архив, содержащий последние версии подписанных **основных** файлов с файлами их отсоединённых действительных электронных подписей в одном из форматов: `.p7s`, `.sig`, `.sgn`, `.sign`.



Элемент управления отображается, если у пользователя есть право на скачивание основных файлов и хотя бы один основной файл подписан действительной электронной подписью с сертификатом. При отсутствии подписей или наличии только простых подписей элемент управления не отображается.

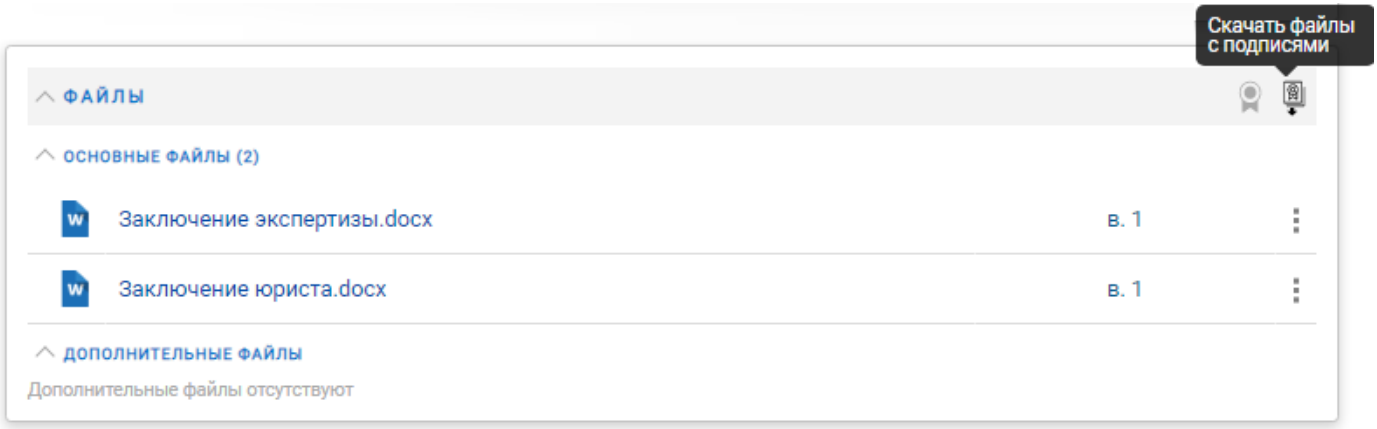


Рисунок 628. Пример элемента управления "Экспорт подписей"

Для включения функции выгрузки файлов с подписями добавьте элемент `Экспорт подписей` в список операций ("fileListFileListOperations") элемента "Список файлов":

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Экспорт с МЧД

Когда флаг установлен, при экспортировании подписи также экспортируется МЧД.

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Экспорт с визуализацией ЭП

Когда свойство активно, в архив с файлами и откреплеными подписями, формируемый по кнопке **Скачать файлы с подписями**, добавляется документ с отображением штампа электронной подписи. По умолчанию свойство отключено, файлы экспортируются без штампа ЭП. Название файла с ЭП будет сформировано по принципу `Имя-файла.stamp.pdf`.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Формат подписи

Свойство определяет расширение экспортируемой подписи. Доступен выбор из четырёх расширений: `.P7S` (формат экспорта подписи по умолчанию), `.SIG`, `.SGN`, `.SIGN`.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Запрос изменения срока исполнения

ЭУ получает и отображает данные запроса на изменение срока исполнения заданий. ЭУ предназначен для работы с запросами на изменение срока исполнения заданий в заданиях "Управление документами". Активный запрос на изменение срока в карточке отображается специальным блоком с жёлтым фоном, кнопками действия, автором запроса и комментарием.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки

Текст, отображаемый в кнопке. Если значение не установлено, отображается значение по умолчанию.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Справочники

Web-клиент Конструктор Web-разметок

Справочники

Описание элементов управления группы "Справочники".

- Выбор папки
- Записи справочника сотрудников
- Отображение организации сотрудника
- Другие элементы управления группы в навигационном меню слева…

Выбор папки

Элемент управления Выбор папки предназначен для выбора папки.

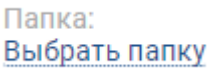


Рисунок 629. Пример отображения элемента управления "Выбор папки"

В зависимости от значения свойства **Отображать дерево папок** в элементе управления может отображаться ссылка. При нажатии на ссылку будет открыто дерево папок для выбора папки, или сразу определённая папка.



Данный ЭУ может быть добавлен в разметки редактирования карточек, а также в разметки просмотра локации *Поиск карточек*

для ссылок.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если папка не выбрана.

Локализуемый

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Значение по умолчанию

Определяет папку, выбранную по умолчанию:

- ***Не указано***. По умолчанию не выбирается никакая папка.
- ***Текущая папка***. По умолчанию выбирается активная папка.



Если в текущей папке создание карточек запрещено (например, папка является виртуальной, или установлено ограничение на создание карточек), то к предложению выбрать папку будет добавлено сообщение: "(в текущей запрещено)".

Использовать список папок

Устанавливает список папок, из которых можно выбирать папку. Если список не установлен, пользователю будут доступны все папки.

Список настраивается с помощью формы, открываемой по нажатию кнопки ...

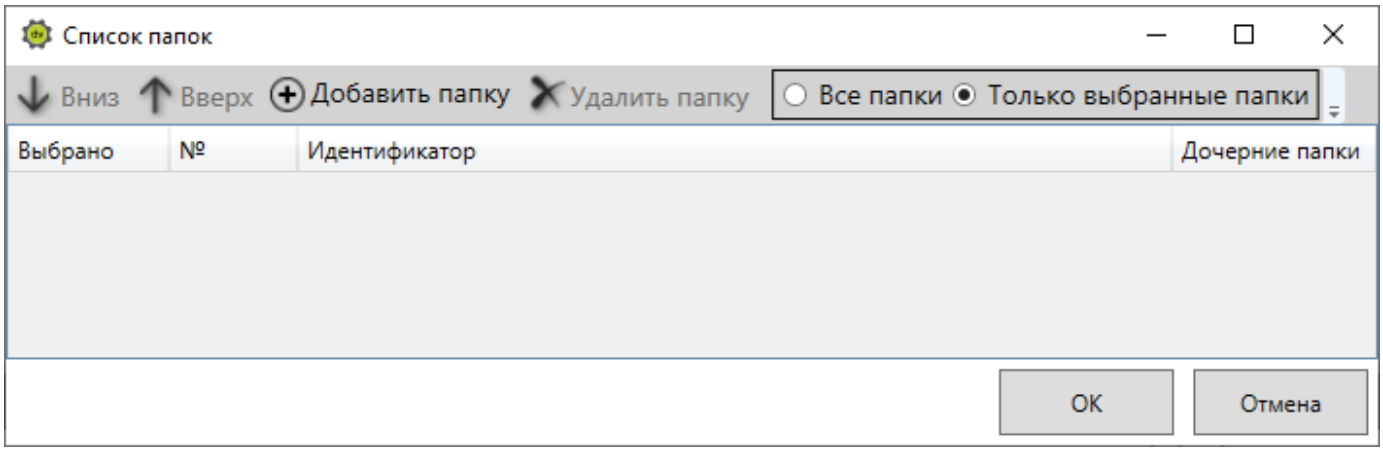


Рисунок 630. Настройка списка папок

Чтобы добавить папку:

1. Нажмите кнопку **Добавить папку**.
2. В поле Идентификатор введите идентификатор (GUID) папки.
3. Если для выбора должны быть также доступны подпапки указанной папки, установите флаг Дочерние папки.

Чтобы изменить порядок отображения папок у пользователя, используйте

кнопки **Вниз**/**Вверх**.

Если требуется, некоторые папки могут быть временно исключены из списка доступных для выбора. Для этого снимите с них флаг Выбрано — на панели инструментов должен быть включён режим ***Только выбранные папки***. В режиме ***Все папки*** пользователям будут доступны все папки из списка независимо от значения флага Выбрано.

Если в Значение по умолчанию установлена ***Текущая папка*** и данная папка есть в списке, у пользователя она будет отображаться первой в списке. Если ***Текущая папка отсутствует*** в списке, она также будет добавлена в список папок, доступных пользователю. Если нужно полностью ограничить возможность выбора папок установленным списком, выберите ***Не указано*** в Значение по умолчанию.

Обязательное

Когда флаг установлен, папка должна быть выбрана перед сохранением карточки. Когда флаг не установлен, выбирать папку не требуется.

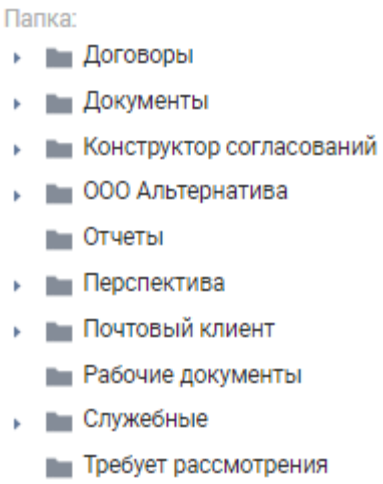
Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отображать дерево папок

При установленном флаге в элементе управления сразу отображается дерево папок, а не кнопка для его открытия.



Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Записи справочника сотрудников

Элемент управления `Записи справочника сотрудников` предназначен для выбора записей из *Справочника сотрудников*. В зависимости от настроек пользователь может выбирать сотрудников, организации, подразделения, группы, роли и должности, а также поисковые слова.

Вариант отображения ЭУ зависит от свойства `Режим редактирования`:

- ***Редактирование***. В данном режиме ЭУ предоставляет возможность добавлять и удалять записи.

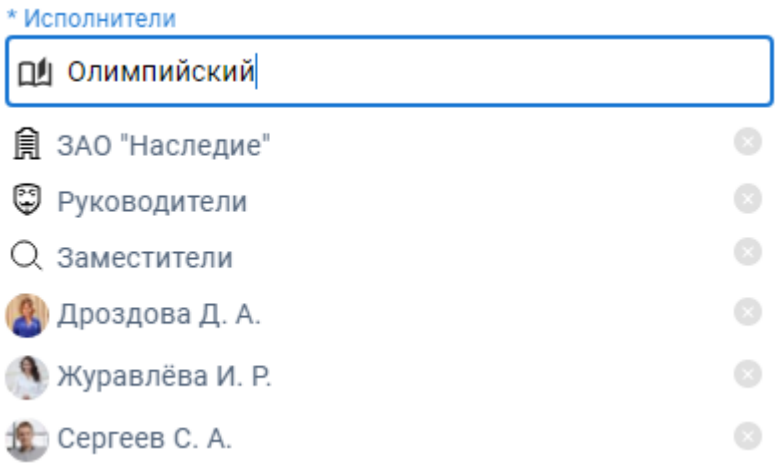


Рисунок 631. Элемент управления "Сотрудники" в режиме редактирования

Пользователь может выполнять быстрый поиск и выбирать записи из справочника.

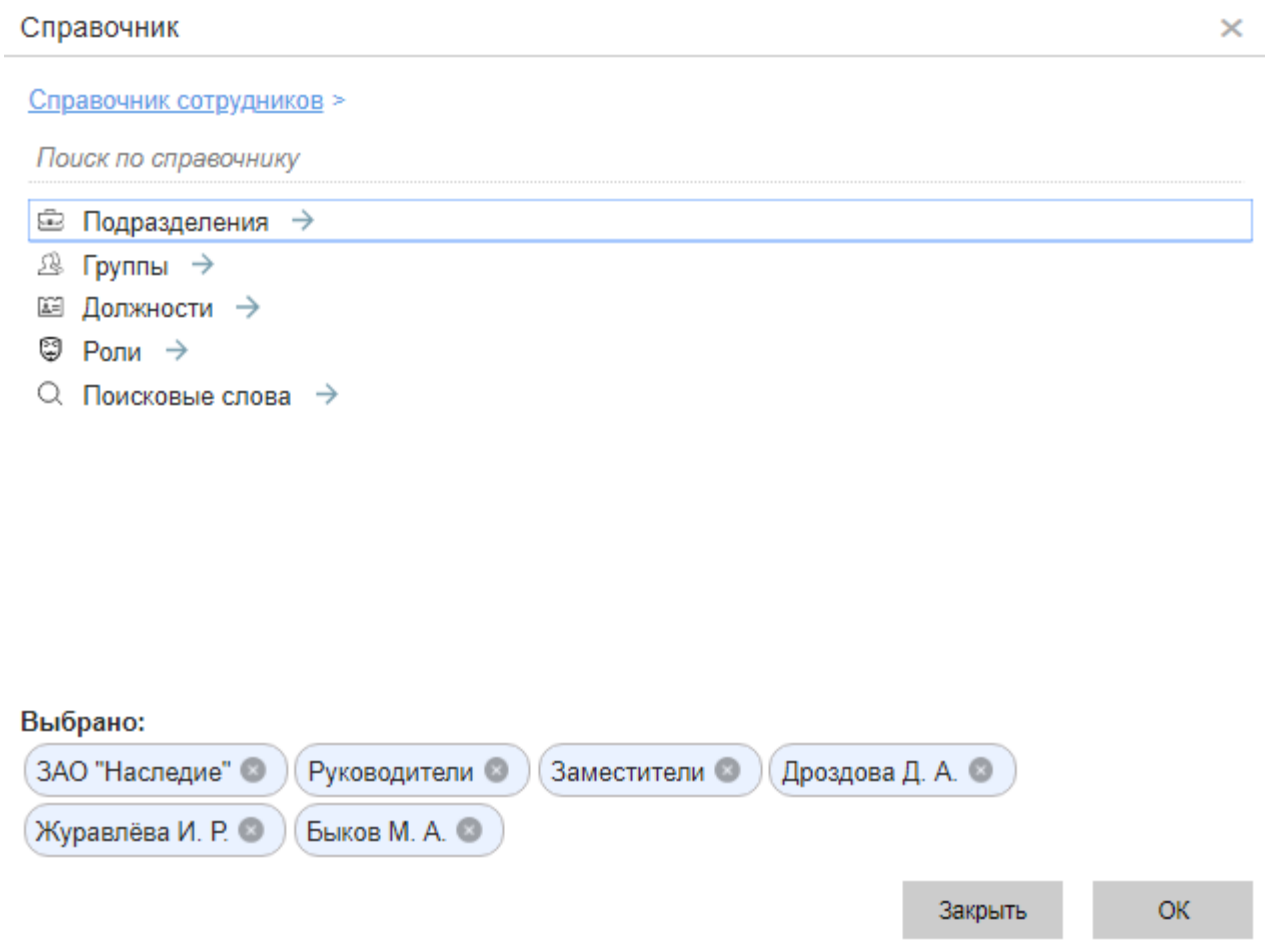


Рисунок 632. Окно выбора записей из Справочника сотрудников с выбранными значениями

- ***Без редактирования***. В данном режиме в ЭУ отображается список выбранных записей из связанной секции карточки.

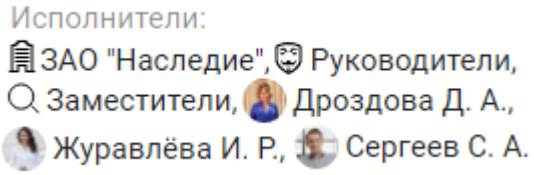


Рисунок 633. Элемент управления "Сотрудники" в режиме "Без редактирования"

- ***По месту***. При открытии карточки в ЭУ отображается список выбранных записей из связанной секции карточки. Изменение списка осуществляется в отдельном окне, открываемом при нажатии на текущее значение.

> Если ЭУ используется в разметке с режимом *Создание* или *Редактирование*, и разметка не была сохранена явным образом, при обновлении страницы выбранное в элементе значение будет сброшено, даже если ЭУ находится в режиме редактирования по месту.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, когда значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Формат отображения подсказки к сотрудникам

Формат отображения данных о сотруднике, отображаемых во всплывающей подсказке к записи Сотрудник. Возможные варианты:

- ***ФИО***.
- ***ФИО+Должность***.
- ***Не используется***. Во всплывающей подсказке отображается текст из поля Подсказка.

Формат отображения сотрудников

Формат отображения данных о сотруднике, отображаемых в записях Сотрудник. Возможные варианты:

- ***Фамилия И.О.***.
- ***Строка отображения*** (формат установленный в Справочнике сотрудников).

Данные

Алгоритм редактирования данных

Свойство предназначается для выбора алгоритма сохранения строк в ЭУ. Свойство позволяет выбрать следующие два варианта значений:

- Номера строк — режим по умолчанию. Если в разметке используется несколько ЭУ с одной табличной секцией, нужно использовать алгоритм основанный на номерах строк.
- Идентификаторы строк — если ЭУ только один, предпочтительнее использовать алгоритм основанный на идентификаторах строк.

Источник данных

Коллекционная или структурная секция карточки или связанной карточки, содержащая данные для элемента управления. Чтобы выбрать записи типов *Группа*, *Подразделение* и *Роль*, поле карточки, указанное в источнике данных, должно иметь тип `RefId` или `UniqueId`.

Чтобы выбрать источник данных, нажмите кнопку ... . Будет открыто окно:

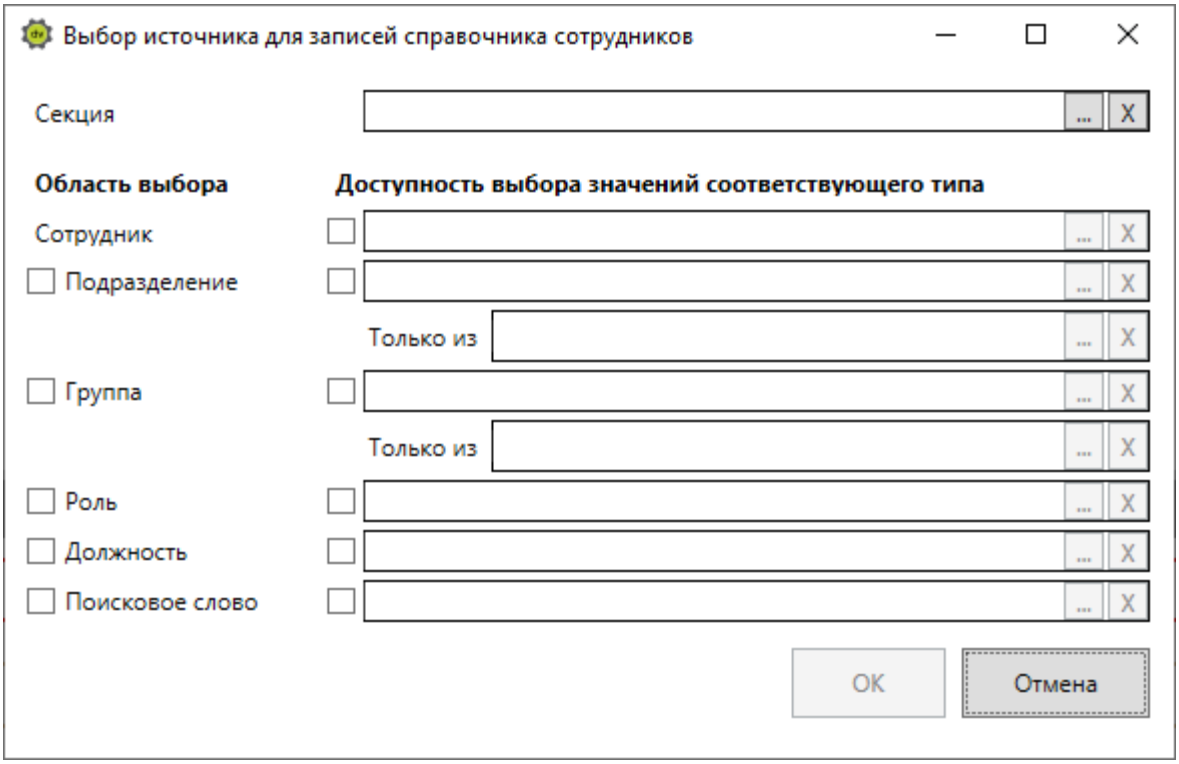


Рисунок 634. Выбор источника данных для элемента управления

Настройки источника содержат следующие элементы:

- Поле "Секция".
  Выбор секции с полями со значениями элемента управления. Если к разметке подключены дополнительные или расширенные источники данных, для выбора будут доступны секции карточки и подключенных источников.
- Столбец "Область выбора".
  При установленном флаге разрешает выбирать записи из соответствующей области (из подразделений, из группы и т.д.).
  - Строки "Сотрудник", "Подразделение", "Группа", "Роль", "Должность", "Поисковое слово".
    Флаги в начале строки позволяют выбрать поля карточки со значением записи соответствующего типа. Если поле не указано, возможность выбора значений данного типа будет недоступна.
    При использовании поисковых слов, в значение записывается само поисковое слово, а не соответствующий сотрудник. Конкретное значение определяется согласно поисковому слову автоматически при дальнейшей работе с карточкой.
- Столбец "Доступность выбор значений соответствующего типа".
  Если флаг установлен, разрешается выбирать записи соответствующего типа (сотрудников, подразделения и т.д.).

- Поле "Только из".
  Если значение задано, ограничивается возможность выбора записей: пользователи смогут выбирать записи только из указанного элемента и его дочерних элементов.

Пример 3. Образец настройки

Например, чтобы разрешить выбирать только сотрудников из подразделений или организаций, но не разрешать выбирать сами подразделения, должны быть установлены два флага:

1. Доступность выбора значений соответствующего типа в строке *Сотрудник*.



Также должно быть выбрано поле карточки, содержащее значение типа *Сотрудник*.

2. **Область выбора** в строке *Подразделение*.

Чтобы разрешить выбирать только подразделения (другие типы записей, в т.ч. сотрудники будут недоступны для выбора), должны быть установлены два флага:

1. **Область выбора** в строке "Подразделение".
2. **Доступность выбор значений соответствующего типа** в строке "Подразделение". Также должно быть выбрано поле карточки, содержащее значение типа "Подразделение".

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Задержка поиска (мс)

Интервал времени от ввода символа в строку поиска до выполнения быстрого поиска. По умолчанию 500 мс.

Использование последних

Определяет, требуется ли отображать последние выбранные записи. Последние выбранные записи отображаются при выборе элемента управления (только в режиме ***Редактирование***), а также при нажатии клавиш ↓ и Пробел. Отображается максимум десять записей.

При работе в карточках *Задание* и *Группа заданий* в списке отображаются последние исполнители, в других карточках — последние выбранные записи.

Список последних выбранных сотрудников не очищается при перезапуске службы Сервера Web-клиента или при нажатии использовании функции "Вернуть настройки по умолчанию" в профиле пользователя.

Множественный выбор

При установленном флаге разрешается выбирать несколько записей из Справочника сотрудников. Если флаг снят, пользователи смогут выбирать

только одно значение.



Свойство доступно только при работе с табличной секцией.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Показывать иконку

Свойство регулирует отображение иконки элемента управления. Когда флаг установлен (по умолчанию) иконка ЭУ отображается. Когда флаг снят, иконка не отображается.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется в отдельном окне, которое открывается при нажатии на элемент управления.

  Данный вариант подходит для разметки режима редактирования и разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

Данный вариант может быть выбран в разметке режима редактирования и просмотра.

Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Сотрудник по умолчанию

Определяет для элемента управления значение по умолчанию:

- ***Не задан***. Значение по умолчанию не установлено,
- ***Текущий пользователь***. По умолчанию будет выбран текущий сотрудник.

  Возможность выбора сотрудника по умолчанию доступна, если в свойстве Источник данных включена возможность выбора сотрудников.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед выбором значения

Вызывается перед выбором значения элемента управления.

Перед загрузкой результатов поиска

Вызывается перед загрузкой результатов поиска.

Перед удалением значения

Вызывается перед удалением значения элемента управления.

После выбора значения

Вызывается после выбора значения из справочника.

После загрузки результатов поиска

Вызывается после загрузки результатов поиска.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После закрытия окна справочника

Вызывается после закрытия окна выбора значения из справочника.

После изменения текущего фильтра

Вызывается после изменения фильтра отображаемых значений элемента управления.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна справочника

Вызывается после открытия окна выбора значения из справочника.

После удаления значения

Вызывается после удаления значения элемента управления.

При изменении текущего фильтра

Вызывается перед изменением фильтра отображаемых значений элемента управления.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Отображение организации сотрудника

Элемент управления Отображение организации сотрудника предназначен для отображения организации и подразделения сотрудника в формате строки.

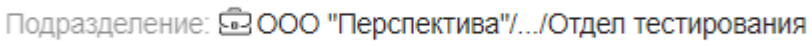


Рисунок 635. Элемент управления "Отображение организации сотрудника"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Вывод иерархии

Определяет вариант отображения иерархии подразделений:

- ***Выводить до первой организации***. Отображать все подразделения и только первую непосредственную организацию (без отображения вышестоящих организаций):

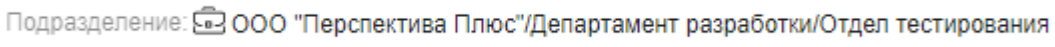
  

  Рисунок 636. Выводить до первой организации

- ***Выводить непосредственное подразделение***. Отображать только

подразделение сотрудника:

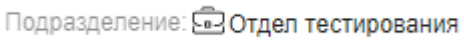


Рисунок 637. Выводить непосредственное подразделение

- ***Выводить всю иерархию***. Отображать все подразделения и все вышестоящие организации, кроме самой головной.



Рисунок 638. Выводить всю иерархию

Обрезать при отображении

При активном флаге все промежуточные организации и подразделения будут заменены на "...".

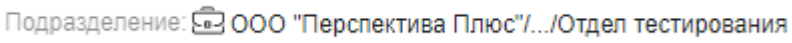


Рисунок 639. Обрезать при отображении

Отображать головную организацию

При включенной настройке в элементе управления будет также отображаться головная организация сотрудника, иначе только дочерние организации и подразделения (в соответствии со значением свойства `Вывод иерархии`).

Разделитель

Символ или строка, используемая для разделения дочерних элементов при отображении.

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый.

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, ссылающееся на сотрудника, для которого отображается подразделение. Тип поля: `RefId`.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства `Видимость`:

- Когда флаг `Видимость` установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг `Видимость` установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг `Видимость` НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Партнер

Элемент управления Партнер предназначен для выбора сотрудника из *Справочника контрагентов* и изменения данных справочника.

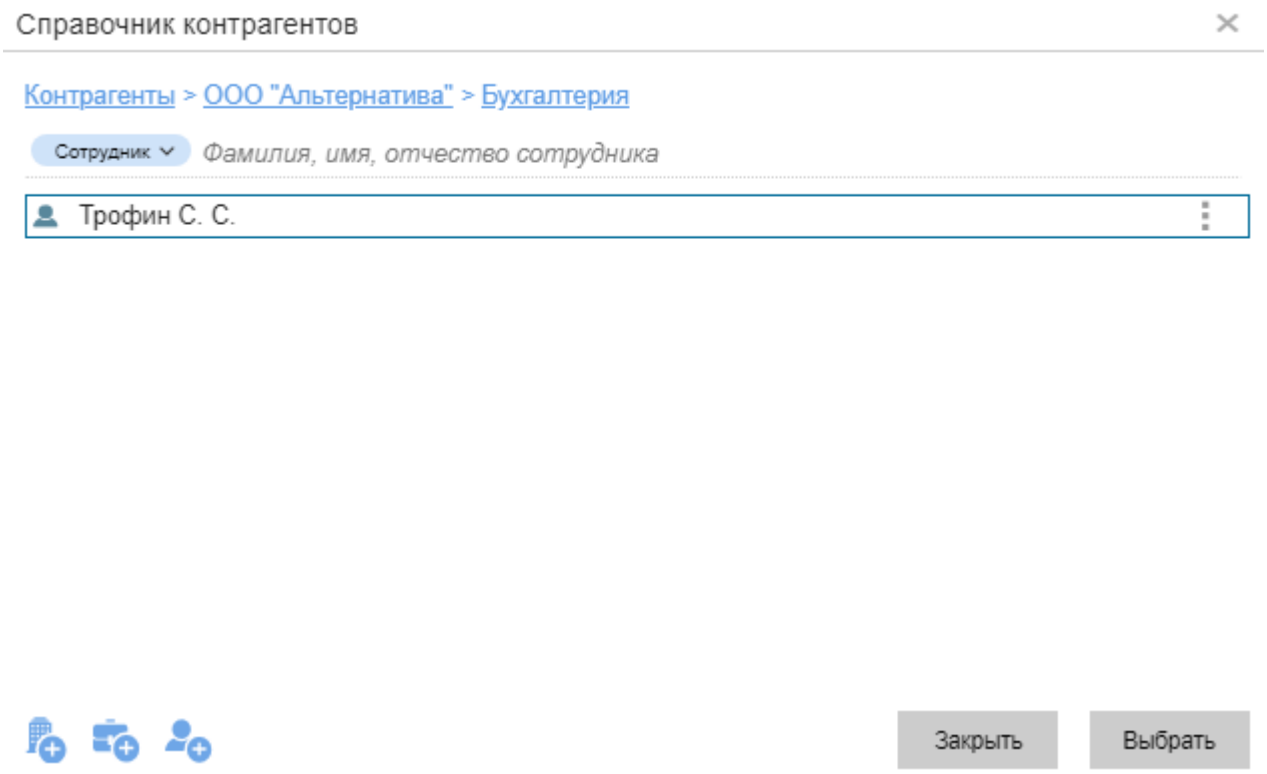


Рисунок 640. Справочник контрагентов

Вариант отображения ЭУ зависит от свойства **Режим редактирования**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

  В режиме Редактирование поддерживается быстрый поиск и выбор сотрудника из *Справочника контрагентов*. Быстрый поиск и поиск при выборе из справочника осуществляется по фамилии, имени и отчеству сотрудника.

  Рисунок 641. Элемент управления "Партнер" в режиме редактирования

  Быстрый поиск осуществляется по названию организации, подразделения контрагента и ФИО сотрудника контрагента, а в режиме выбора из

справочника также возможен поиск организации по ИНН.

- ***По месту***. Выбор и изменение выбранного сотрудника контрагента осуществляется при нажатии на ссылку в ЭУ. Поиск и выбор осуществляется аналогично поиску и выбору для режима ***Редактирование***.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Рисунок 642. Элемент управления "Партнер" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

Особенность удаления записей из Справочника контрагентов

Запись будет удалена от имени текущего пользователя, если пользователь после добавления записи в Справочник контрагентов откажется от сохранения изменений.

Удаление будет произведено, только если у текущего пользователя есть права на удаление.

Запись будет удалена от имени учётной записи Web-клиента, если у пользователя нет прав на удаление записей из Справочника контрагентов.

Если нужно запретить удаление записей для УЗ Web-клиента, запретите данную операцию для системной учётной записи в настройках дискреционной безопасности Справочника контрагентов.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например,

JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если сотрудник не выбран.

Локализуемый

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Подсказка к значению

Содержимое всплывающей подсказки, отображаемой если сотрудник выбран. Возможные варианты:

- ***ФИО***.
- ***ФИО+Должность***.
- ***Не используется***. Во всплывающей подсказке отображается текст из поля Подсказка.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Формат отображения

Определяет формат отображения данных сотрудника в ЭУ:

- ***Фамилия И.О.***.
- ***Строка отображения***. Используется формат отображения, настроенный в *Справочнике контрагентов*.

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее ссылку на сотрудника из *Справочника контрагентов*.

Выбирать из

Ограничивает возможность поиска/добавления сотрудников. Пользователь сможет искать и добавлять сотрудников только в выбранную организацию или в подчиненные организации и подразделения.

Если значение не установлено, область поиска/добавления не ограничивается.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Задержка поиска (мс)

Определяет время задержки (в мс) от ввода последнего символа в строку поиска до выполнения быстрого поиска по справочнику.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Показывать иконку

Свойство регулирует отображение иконки элемента управления. Когда флаг установлен (по умолчанию) иконка ЭУ отображается. Когда флаг снят, иконка не отображается.

Редактирование справочника

Активирует функцию редактирования данных Справочника контрагентов с помощью данного элемента управления. Когда флаг установлен, редактирование разрешено при наличии у пользователя прав. Когда флаг снят, функции редактирования справочника не предоставляются.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед загрузкой результатов поиска

Вызывается перед отправкой запроса на получение данных Справочника контрагентов с сервера.

Перед закрытием окна справочника

Вызывается перед закрытием окна выбора значения из справочника.

Перед открытием окна справочника

Вызывается перед открытием окна выбора значения из справочника.

После загрузки результатов поиска

Вызывается после получения данных Справочника контрагентов с сервера.

После закрытия окна справочника

Вызывается после закрытия окна выбора значения из справочника.

После изменения текущего фильтра

Вызывается после изменения пользователем фильтра данных Справочника контрагентов.

После открытия окна справочника

Вызывается после открытия окна выбора значения из справочника.

При изменении текущего фильтра

Вызывается при изменении пользователем фильтра данных Справочника контрагентов.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Справочник контрагентов

Справочник представляет собой иерархическую структуру узлов — организаций и подразделений, групп и предоставляет возможность работы с группами контрагентов и отдельными сотрудниками.

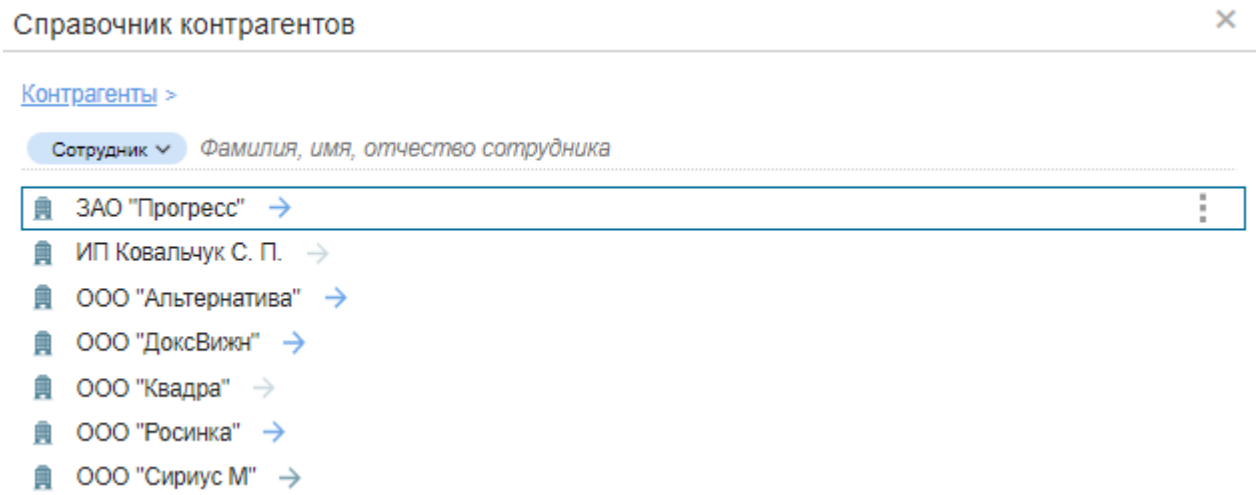


Рисунок 643. Представление справочника контрагентов

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

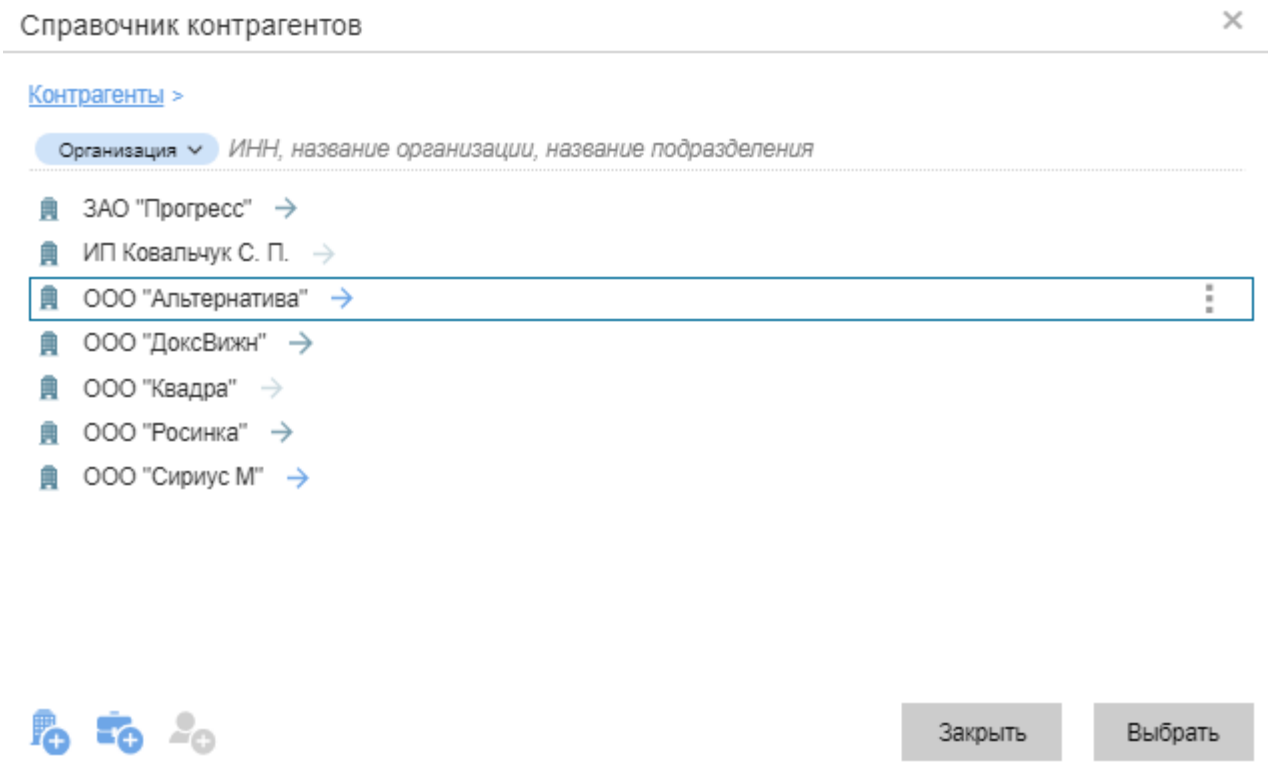
Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Подразделение контрагента

Элемент управления Подразделение контрагента предназначен для выбора организации или подразделения из *Справочника контрагентов* и изменения данных справочника.



Вариант отображения ЭУ зависит от свойства **Режим редактирования**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

  ЭУ в данном режиме поддерживает быстрый поиск и выбор из *Справочника контрагентов*. Быстрый поиск осуществляется по названию контрагента, а в режиме выбора из справочника — по названию, полному названию и ИНН.

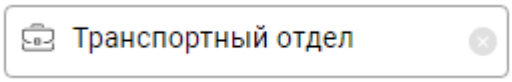


Рисунок 644. Элемент управления "Подразделение контрагента" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Выбор и изменение контрагента осуществляется при нажатии на ссылку в ЭУ. Поиск и выбор осуществляется аналогично поиску и выбору для режима ***Редактирование***.



Рисунок 645. Элемент управления "Подразделение контрагента" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

Особенность удаления записей из Справочника контрагентов

Запись будет удалена от имени текущего пользователя, если пользователь после добавления записи в Справочник контрагентов откажется от сохранения изменений.

Удаление будет произведено, только если у текущего пользователя есть права на удаление.

Запись будет удалена от имени учётной записи Web-клиента, если у пользователя нет прав на удаление записей из Справочника контрагентов.

Если нужно запретить удаление записей для УЗ Web-клиента, запретите данную операцию для системной учётной записи в настройках дискреционной безопасности Справочника контрагентов.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее ссылку на организацию/подразделение *Справочника контрагентов*.

Выбирать из

Ограничивает возможность поиска и добавления организаций. Пользователь сможет искать и добавлять организации или подразделения только в выбранную организацию. Если значение не установлено, область поиска или добавления не ограничивается.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Выбор организаций

Определяет возможность выбора организации из *Справочника сотрудников*. Когда флаг установлен выбирать организации разрешено. Когда флаг не установлен — не разрешено.

Выбор подразделений

Определяет возможность выбора подразделения из *Справочника сотрудников*. Когда флаг установлен выбирать подразделения разрешено. Когда флаг не установлен — не разрешено.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Показывать иконку

Свойство регулирует отображение иконки элемента управления. Когда флаг установлен (по умолчанию) иконка ЭУ отображается. Когда флаг снят, иконка не отображается.

Редактирование справочника

Активирует функцию редактирования данных Справочника контрагентов с помощью данного элемента управления. Когда флаг установлен, редактирование разрешено при наличии у пользователя прав. Когда флаг снят, функции редактирования справочника не предоставляются.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед загрузкой результатов поиска

Вызывается перед загрузкой результатов поиска.

Перед закрытием окна справочника

Вызывается перед закрытием окна выбора значения из справочника.

Перед открытием окна справочника

Вызывается перед открытием окна выбора значения из справочника.

После загрузки результатов поиска

Вызывается после загрузки результатов поиска.

После закрытия окна справочника

Вызывается после закрытия окна выбора значения из справочника.

После изменения текущего фильтра

Вызывается после изменения фильтра отображаемых значений элемента управления.

После открытия окна справочника

Вызывается после открытия окна выбора значения из справочника.

При изменении текущего фильтра

Вызывается перед изменением фильтра отображаемых значений элемента управления.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Руководитель подразделения

ЭУ представляет собой флаг, указывающий, является ли сотрудник руководителем подразделения, в котором он состоит. ЭУ может быть добавлен только на разметку сотрудника контрагента.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст для значения "да"

Текст, отображаемый в карточке, когда ЭУ в значении "да".

Текст для значения "нет"

Текст, отображаемый в карточке, когда ЭУ в значении "нет".

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Значение по умолчанию

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра,

необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

После смены данных

Вызывается после изменения содержимого элемента управления.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Подразделение

Элемент управления Подразделение предназначен для выбора организации или подразделения из *Справочника сотрудников*.

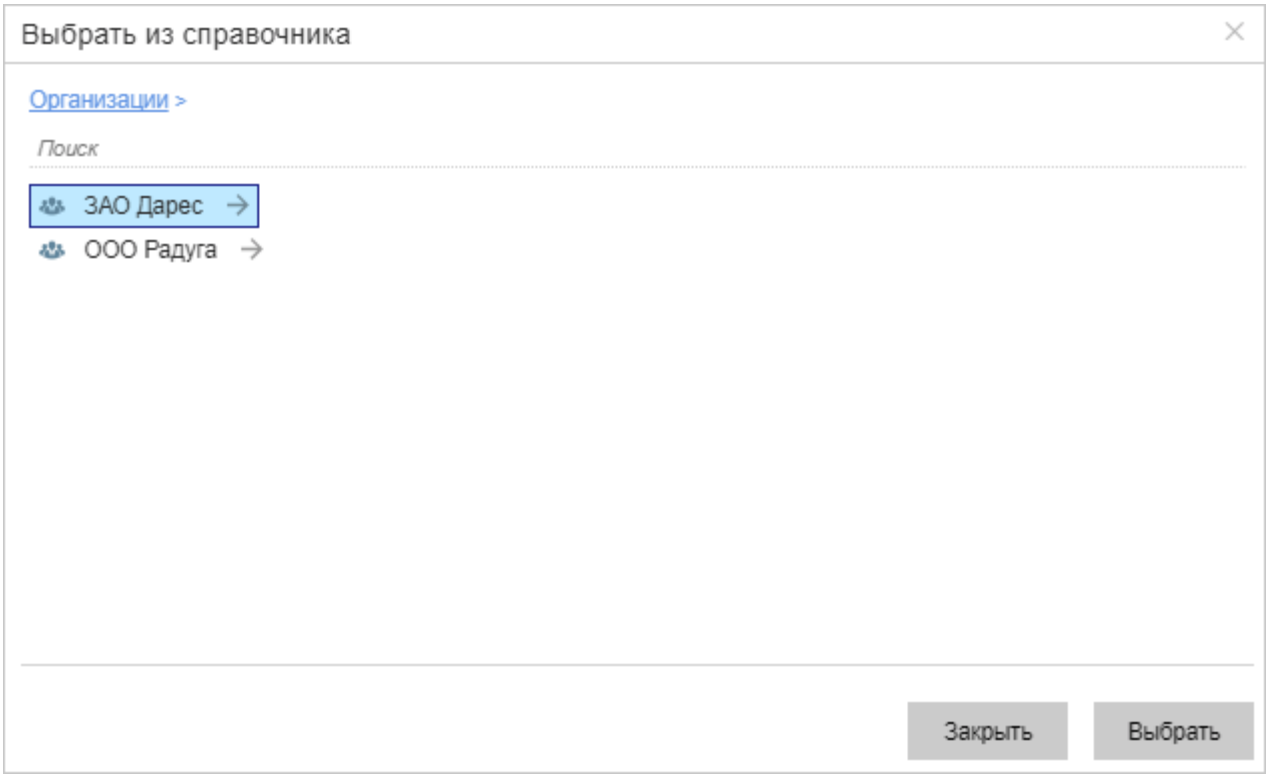


Рисунок 646. Справочник сотрудников

Вариант отображения ЭУ зависит от свойства `Режим редактирования`:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

  В режиме `Редактирование` поддерживается быстрый поиск и выбор подразделения из *Справочника сотрудников*.

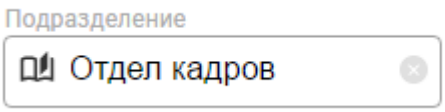
  

  Рисунок 647. Элемент управления "Подразделение" в режиме редактирования

  Быстрый поиск осуществляется по названию подразделения, а в режиме выбора из справочника — по названию, полному названию и ИНН.

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Выбор и изменение подразделения осуществляется при нажатии на ссылку в ЭУ. Поиск и выбор осуществляется аналогично поиску и выбору для режима ***Редактирование***.



Рисунок 648. Элемент управления "Подразделение" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

При выборе значения из справочника, представленного в виде списка, переход к подразделениям организации осуществляется при нажатии на значок →, а выбор значения — нажатием кнопки **Выбрать**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:

- Когда флаг установлен, текст метки отображается всегда.
- Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее ссылку на организацию *Справочника сотрудников*.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг

установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Выбор организаций

Определяет возможность выбора организации из *Справочника сотрудников*. Когда флаг установлен выбирать организации разрешено. Когда флаг не установлен — не разрешено.

Выбор подразделений

Определяет возможность выбора подразделения из *Справочника сотрудников*. Когда флаг установлен выбирать подразделения разрешено. Когда флаг не установлен — не разрешено.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Задержка поиска (мс)

Определяет время задержки (в мс) от ввода последнего символа в строку поиска до выполнения быстрого поиска по справочнику.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.

- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим диалога

Меняет режим отображения диалога для выбора подразделения. Доступны два варианта:

- ***Список***. Подразделения и организации будут отображаться в виде списка. Чтобы перейти на следующий уровень иерархии, нужно нажать кнопку →.
- ***Древовидный***. Подразделения и организации будут отображаться в виде древовидной иерархической структуры.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Сотрудник



Поддержка элемента управления прекращена, элемент устарел и недоступен для выбора в библиотеке. Рекомендуется использовать ЭУ "Записи справочника сотрудников".

Элемент управления `Сотрудник` предназначен для просмотра и выбора сотрудника из *Справочника сотрудников*.

Вариант отображения ЭУ зависит от свойства **`Режим редактирования`**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

  В режиме `Редактирование` поддерживается быстрый поиск и выбор сотрудника из *Справочника сотрудников*. Быстрый поиск и поиск при выборе из справочника осуществляется по фамилии, имени и отчеству сотрудника.



Рисунок 649. Элемент управления "Сотрудник" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Выбор и изменение выбранного сотрудника осуществляется при нажатии на ссылку в ЭУ. Поиск и выбор осуществляется аналогично поиску и выбору для режима Редактирование.

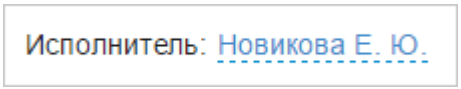


Рисунок 650. Элемент управления "Сотрудник" в режиме редактирования по месту

+ * ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, пока не выбран сотрудник.



Рисунок 651. Заполнитель

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Подсказка к значению

Содержимое всплывающей подсказки, отображаемой если сотрудник выбран. Возможные варианты:

- ***ФИО***.
- ***ФИО+Должность***.
- ***Не используется***.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные для элемента управления. Если выбрана коллекционная секция, то в качестве источника данных будет выбрана первая строка.

Поле данных

Поле карточки, содержащее ссылку на сотрудника *Справочника сотрудников*

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Значение по умолчанию

Значение, которое будет присвоено при создании новой карточки.

Возможны значения: Не задан и Текущий пользователь.

Использовать исполнителей

Определяет, требуется ли отображать последних выбранных исполнителей в начале списка сотрудников.

- Когда флаг установлен, при выборе/поиске десять последних выбранных исполнителей будут отображаться в начале списка (отделяются от других чертой). После выбора исполнитель сохраняется в списке последних.
- Когда флаг снят, последние исполнители не будут выделяться. Выбранный исполнитель в список последних добавляться не будет.



Данная настройка предназначена только для разметки редактирования карточек *Задание* и *Группа заданий*. Список последних выбранных исполнителей хранится в карточке пользователя и является общим для Web-клиента и Windows-клиента,для элементов управления Сотрудник и "Сотрудники".

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии

кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Последние выбранные

Определяет, требуется ли отображать последних выбранных сотрудников в начале списка сотрудников. Когда флаг установлен, десять последних выбранных сотрудников перемещаются в начало списка. Последние выбранные сотрудники отделяются от других чертой.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования ***По месту***.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования

По месту.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Сотрудники



Поддержка элемента управления прекращена, элемент устарел и недоступен для выбора в библиотеке. Рекомендуется использовать ЭУ "Записи справочника сотрудников".

Элемент управления `Сотрудники` предназначен для выбора нескольких сотрудников из *Справочника сотрудников*, а также отображения выбранных сотрудников.

Вариант отображения ЭУ зависит от свойства `Режим редактирования`:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

  В режиме Редактирование поддерживается быстрый поиск и выбор сотрудника из *Справочника сотрудников*. Быстрый поиск и поиск при выборе из справочника осуществляется по фамилии, имени и отчеству сотрудника.

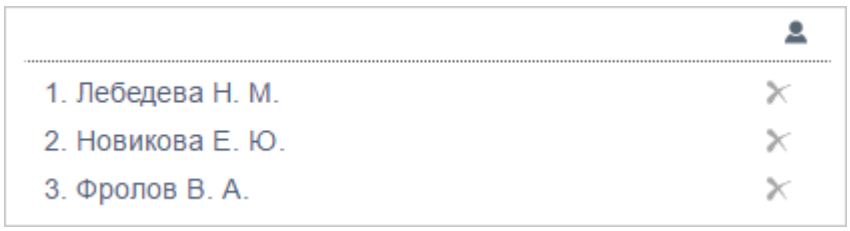


Рисунок 652. Элемент управления "Сотрудники" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Выбор и изменение выбранного сотрудника осуществляется при нажатии на ссылку в ЭУ. Поиск и выбор осуществляется аналогично поиску и выбору для режима ***Редактирование***.

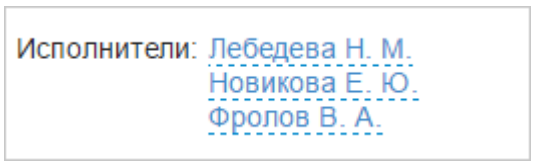


Рисунок 653. Элемент управления "Сотрудники" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, когда значение не выбрано.



Рисунок 654. Заполнитель

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Подсказка к значению

Содержимое всплывающей подсказки, отображаемой при наведении курсора на строку сотрудника. Возможные варианты:

- ***ФИО***.
- ***ФИО+Должность***.
- ***Не используется***. Во всплывающей подсказке отображается текст из поля Подсказка.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Коллекционная секция карточки, содержащая данные для элемента управления.

Поле данных

Поле карточки, содержащее ссылку на сотрудника *Справочника сотрудников*

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Вертикальная ориентация

Определяет вариант отображения выбранных сотрудников. Когда флаг установлен, сотрудники отображаются вертикально. Когда флаг не установлен, сотрудники отображаются горизонтально (в одну линию).

Свойство доступно, если в свойстве **Режим редактирования** выбран режим ***Без редактирования*** или ***По месту***.

Адаптивный

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются,

когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Значение по умолчанию

Значение, которое будет присвоено при создании новой карточки. Возможны значения: ***Не задан*** и ***Текущий пользователь***.

Использовать исполнителей

Определяет, требуется ли отображать последних выбранных исполнителей в начале списка сотрудников.

- Когда флаг установлен, при выборе/поиске десять последних выбранных исполнителей будут отображаться в начале списка (отделяются от других чертой). После выбора исполнитель сохраняется в списке последних.
- Когда флаг снят, последние исполнители не будут выделяться. Выбранный исполнитель в список последних добавляться не будет.



Данная настройка предназначена только для разметки редактирования карточек *Задание* и *Группа заданий*. Список последних выбранных исполнителей хранится в карточке пользователя и является общим для Web-клиента и Windows-клиента,для элементов управления Сотрудник и "Сотрудники".

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Последние выбранные

Определяет, требуется ли отображать последних выбранных сотрудников в начале списка сотрудников. Когда флаг установлен, десять последних выбранных сотрудников перемещаются в начало списка. Последние выбранные сотрудники отделяются от других чертой.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед добавлением сотрудника

Вызывается перед добавлением сотрудника. ==== Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования ***По месту***.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования ***По месту***.

Перед удалением сотрудника

Вызывается перед удалением сотрудника.

После добавления сотрудника

Вызывается после добавления сотрудника.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

После удаления сотрудника

Вызывается после удаления сотрудника.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Справочник сотрудников

Справочник представляет собой иерархическую структуру узлов — организаций и подразделений, групп и предоставляет возможность работы с группами сотрудников, отдельными сотрудниками и должностями.

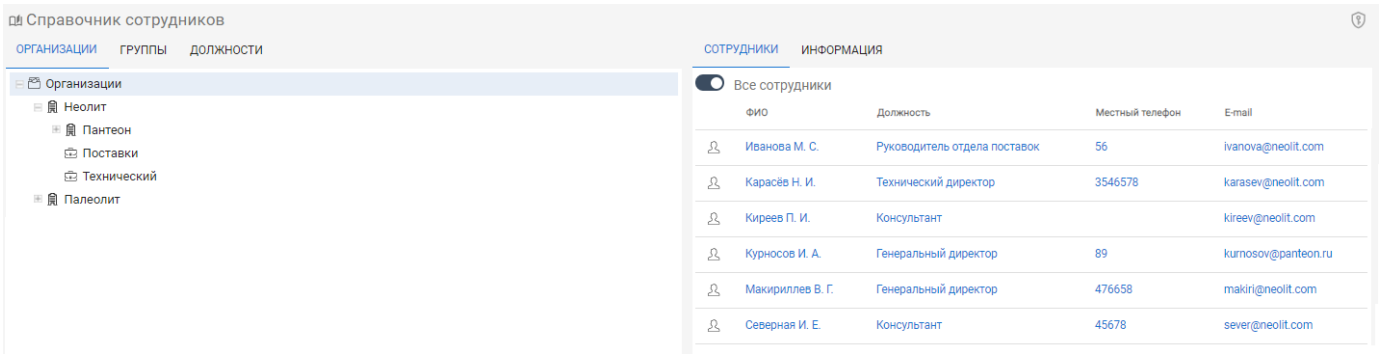


Рисунок 655. Представление справочника сотрудников

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Строка конструктора справочников

Элемент управления Строка конструктора справочников предназначен для отображения и выбора строки из *Конструктора справочников*.

Вариант отображения ЭУ зависит от свойства **Режим редактирования**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

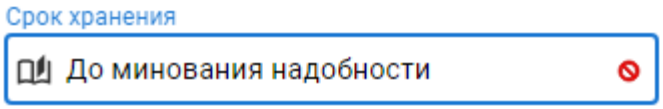


Рисунок 656. Элемент управления "Строка конструктора справочников" в режиме редактирования

Быстрый поиск осуществляется по названию записи, а в режиме выбора из справочника — по названию и описанию.

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

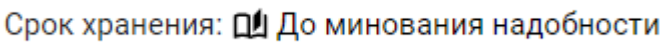


Рисунок 657. Элемент управления "Строка конструктора справочников" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

Для ЭУ можно настроить ограничение области выбора:

- Ограничить выбор записи определённым узлом Конструктора справочников.
- Ограничить быстрый поиск для выбора организации или подразделения.
- Ограничить область выбора: только узел или узел и дочерние узлы.



Если ЭУ используется в разметке с режимом *Создание* или *Редактирование*, и разметка не была сохранена явным образом, при обновлении страницы выбранное в элементе значение будет сброшено, даже если ЭУ находится в режиме редактирования по месту.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если строка справочника не выбрана.

Локализуемый

Использовать описание

Данное свойство влияет на отображение поля описание при открытии справочника.

Описания длиннее двух строк будут обрезаны. Полный текст описания отображается во всплывающей подсказке при наведении курсора. При наведении курсора название и описание выделяются цветом и становятся доступными для выбора. Когда значение в строке выбрано, описание в ЭУ не показывается.

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

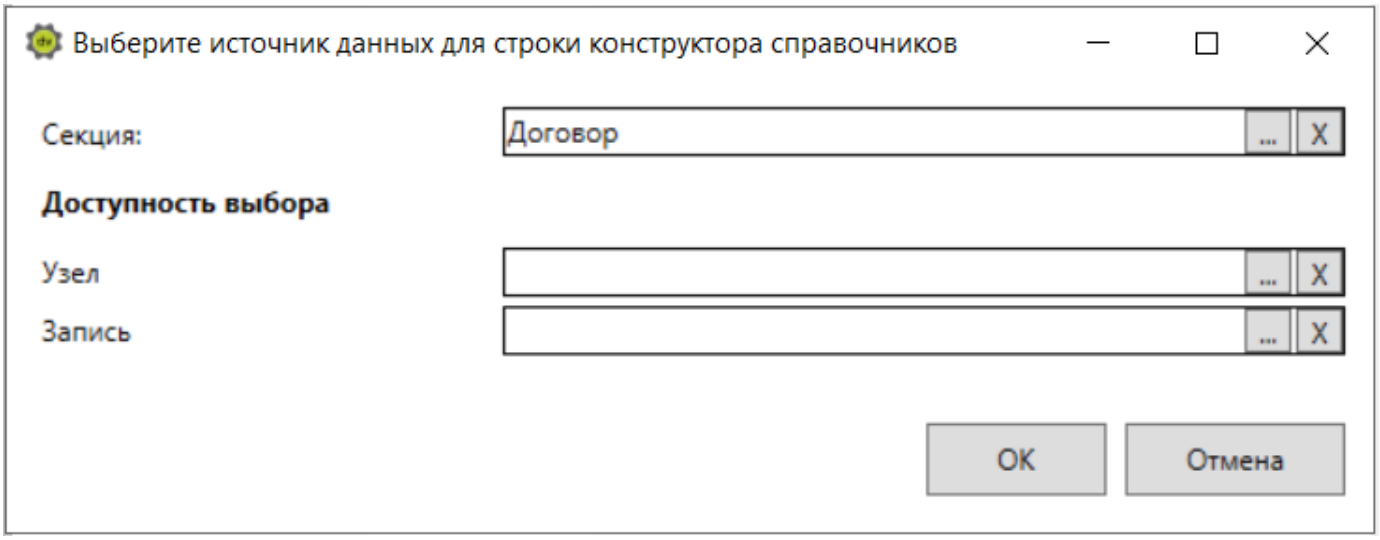
Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Коллекционная или структурная секция карточки или связанной карточки, содержащая данные для элемента управления.

Чтобы выбрать источник данных, нажмите кнопку ... . Будет открыто окно:

Выбор источника данных для элемента управления



Настройки источника содержат следующие элементы:

"Секция"

Выбор секции с полями со значениями элемента управления. Если к разметке подключены дополнительные источники данных, для выбора будут доступны секции карточки и три дополнительных источника.

Столбец "Доступность выбора"

Регулирует доступность выбора узлов и/или записей справочнике в элементе управления при помощи настроек в полях *Узел* и *Запись*.

Узел

Если указан источник данных для узла, в качестве значения элемента управления можно будет выбрать узел справочника.

Запись

Если указан источник данных для записи, в качестве значений ЭУ можно будет выбрать записи справочника.

Если указаны источники данных для обоих вариантов, в качестве значений можно будет выбирать как узлы, так и записи.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Доступные изменения

Свойство определяет, доступны ли пользователю изменения в конструкторе справочников.

Доступные варианты:

- Никакие — любые изменения запрещены.
- Узлы — разрешено изменять только узлы.

- Строки — разрешено изменять только строки.
- Узлы и строки — разрешено изменять и узлы и строки.

Редактирование и работа с ЭУ описаны в разделе "Строка конструктора справочников".

Задержка поиска (мс)

Определяет время задержки (в мс) от ввода последнего символа в строку поиска до выполнения быстрого поиска по справочнику.

Значение по умолчанию

Строка *Конструктора справочников*, выбранная по умолчанию.

Область выбора

Устанавливает ограничение возможности выбора пользователем строк из узла *Конструктора справочников*:



Использование режимов ***Выбранный узел и дочерние к нему*** и ***Только дочерние узлы к выбранному*** при большом количестве записей может сказаться на производительности системы.

- ***Только выбранный узел***. Пользователю будут доступны для выбора только строки выбранного узла.

  Требуемый элемент справочника может быть выбран только из выбранного узла. Быстрый поиск значения будет выполняться только среди строк "выбранного" узла.

- ***Только дочерние узлы к выбранному***. Пользователю будут доступны для выбора узлы и строки только из подчиненных узлов выбранного узла.
- ***Выбранный и дочерние к нему***. Пользователю будут доступны для выбора узлы и строки выбранного узла и его подчиненных узлов.



Данная настройка отображается, только когда в свойстве `Узел` выбран узел.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Показывать иконку

Свойство регулирует отображение иконки элемента управления. Когда флаг установлен (по умолчанию) иконка ЭУ отображается. Когда флаг снят, иконка не отображается.

Разрешить поиск узла

Флаг позволяет выполнять поиск узлов. Если флаг установлен (по умолчанию), поиск узлов доступен.

Режим редактирования

Определяет вариант отображения элемента управления и возможность

изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Узел

Определяет узел *Конструктора справочников*, из которого пользователь может выбирать строки. Если узел не выбран, то предоставляется возможность выбирать строку из всего справочника.

При работе, выбранный корневой узел не будет отображаться в дереве узлов справочника, отображаются только его подчинённые узлы и строки.

События

Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования ***По месту***.

Перед закрытием окна справочника

Вызывается перед закрытием окна выбора значения из справочника.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования

По месту.

Перед открытием окна справочника

Вызывается перед открытием окна выбора значения из справочника.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После закрытия окна справочника

Вызывается после закрытия окна выбора значения из справочника.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна справочника

Вызывается после открытия окна выбора значения из справочника.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Справочник полномочий для доверенностей

Справочник представляет собой двухуровневую иерархическую структуру:

- На первом уровне справочника представлен список справочников полномочий.
- В каждом справочнике представлены коды полномочий.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Полномочия

Элемент управления Полномочия предназначен для выбора полномочий из справочника *Полномочия для МЧД*.

Полномочия отображаются в ЭУ в формате: код + название полномочий.

Вариант отображения ЭУ зависит от свойства **Режим редактирования**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

  В режиме Редактирование поддерживается быстрый поиск и выбор из *Справочника*.



Рисунок 658. Элемент управления "Полномочия" в режиме редактирования

> Поиск по умолчанию осуществляется по всему справочнику. В результате поиска будет отображено, к какому справочнику относится найденный результат.

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Выбрать и изменить полномочия можно нажатием на ссылку в ЭУ. Поиск и выбор осуществляется аналогично поиску и выбору для режима ***Редактирование***.

> При быстром поиске и поиске из справочника происходит проверка прав. Пользователь увидит только те элементы, к которым у него есть доступ на чтение, изменение и использование полномочий.

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее ссылку на карточку доверенности.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Если в настройках ЭУ Полномочия раздел *Данные* не заполнен, ссылка на доверенность в карточку документа не добавляется.

Поведение

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Показывать иконку

Свойство регулирует отображение иконки элемента управления. Когда флаг установлен (по умолчанию) иконка ЭУ отображается. Когда флаг снят, иконка

не отображается.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства `Видимость`:

- Когда флаг `Видимость` установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг `Видимость` установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг `Видимость` НЕ установлен, ЭУ всегда скрыт.

Множественный выбор

При установленном флаге разрешается выбирать несколько полномочий из справочника *Полномочия для МЧД*. Если флаг снят, пользователи смогут выбирать только одно значение.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Задержка поиска (мс)

Определяет время задержки (в мс) от ввода последнего символа в строку поиска до выполнения быстрого поиска по справочнику. По умолчанию 500 мс.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Использование последних

Определяет, требуется ли отображать последние выбранные записи. Последние выбранные записи отображаются при выборе элемента управления (только в режиме ***Редактирование***), а также при нажатии клавиш `↓` и `Пробел`. Отображается максимум десять записей.

Список последних выбранных кодов полномочий очищается автоматически при перезапуске службы Сервера Web-клиента.

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

События

Перед выбором значения

Вызывается перед выбором значения элемента управления.

Перед загрузкой результатов поиска

Вызывается перед загрузкой результатов поиска.

Перед закрытием формы выбора дела

Вызывается перед закрытием формы выбора дела.

Перед открытием формы выбора дела

Вызывается перед открытием формы выбора дела.

Перед удалением значения

Вызывается перед удалением значения элемента управления.

После выбора значения

Вызывается после выбора значения из справочника.

После загрузки результатов поиска

Вызывается после загрузки результатов поиска.

После смены данных

Вызывается после изменения содержимого элемента управления.

После закрытия формы выбора дела

Вызывается после закрытия формы выбора дела.

После открытия формы выбора дела

Вызывается перед открытием формы выбора дела.

После удаления значения

Вызывается после удаления значения элемента управления.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Записи справочников

Элемент управления Записи справочников предназначен для выбора одного или нескольких значений из одного или нескольких справочников. В зависимости от настроек ЭУ пользователь может выбирать сотрудников, организации, подразделения и любые другие элементы из справочников

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Вертикальная ориентация

Когда свойство включено, в режиме *Без редактирования* и *По месту* значения отображаются вертикальным списком с иконками.

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Типы значений

Свойство, позволяющее выбрать значения, которые будут доступны для выбора в элементе управления. Чтобы настроить доступность, нажмите на кнопку . Будет открыто окно:

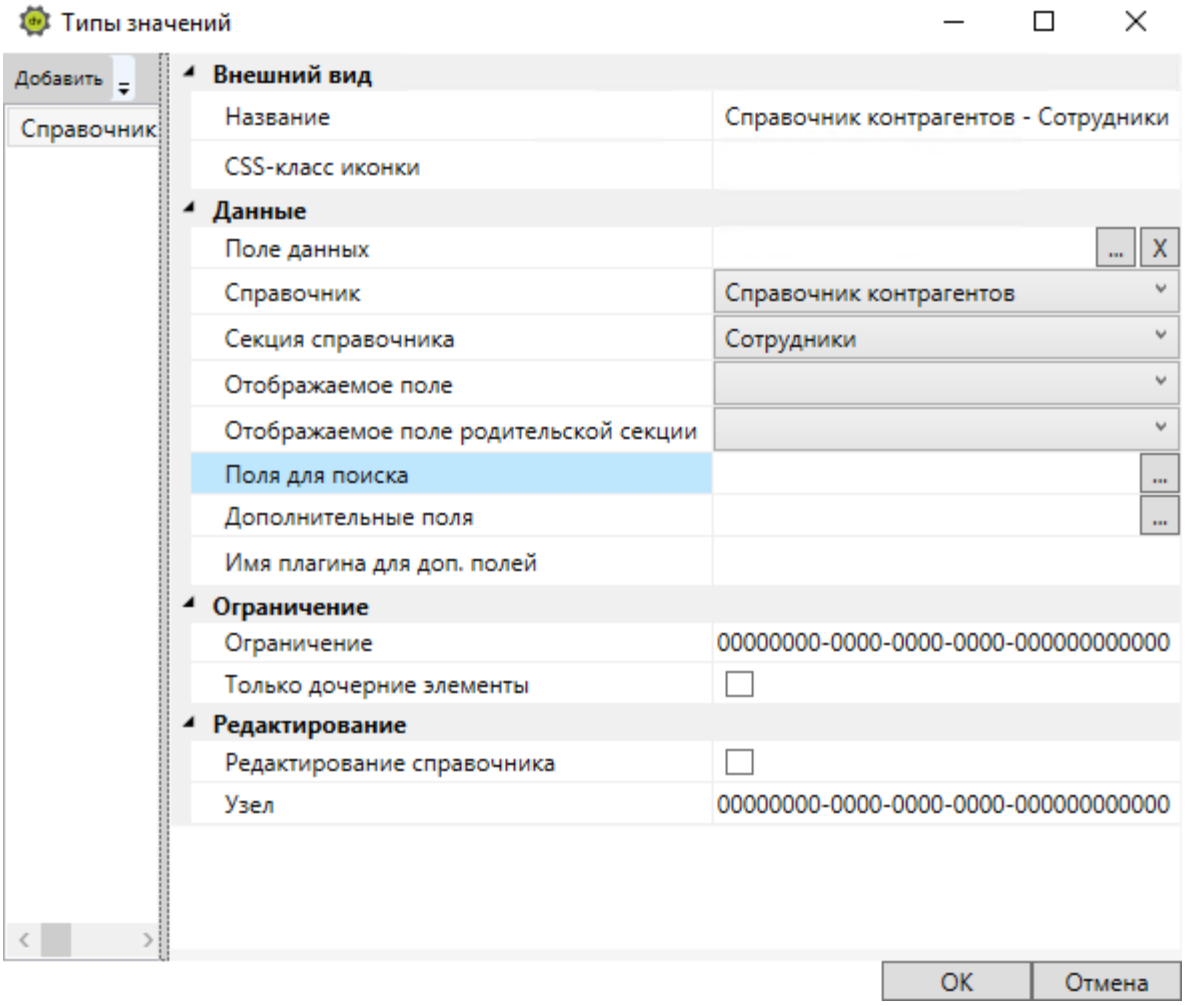


Рисунок 659. Окно "Типы значений"

- Внешний вид
  - *Название* — отображаемое название типа
  - *CSS-класс иконки* — класс иконки, которая будет отображаться слева от названия элемента данного типа, а также слева от названия типа элемента при выборе из нескольких доступных типов.
- Данные
  - *Поле данных* — поле типа `RefId` со ссылкой на строку карточки справочника, в которое будет сохранено значение данного типа.
  - *Справочник* — карточка справочника, из которой будет выбираться значение ЭУ данного типа. Например, если выбран один из справочников, поддерживающих редактирование, становятся доступными для редактирования свойства: *Редактирование справочника*, *Узел*.
  - *Секция справочника* — табличная или древовидная секция выбранной карточки справочника, из строк которой будет выбираться значение ЭУ

данного типа. Если в параметре *Справочник* выбран один из справочников, поддерживающих редактирование, и в текущем параметре выбрано ***Сотрудники***, становятся доступными для редактирования свойства *Редактирование справочника* и *Узел*.

- *Отображаемое поле* — строковое поле секции справочника, значение которого будет использовано в качестве строки отображения элемента данного типа.
- *Отображаемое поле родительской секции* — строковое поле, значение которого используется в качестве строки отображения для родительского элемента при построении иерархических структур (например, в справочниках с древовидной вложенностью).
  Поле становится доступным, если в свойстве "Секция справочника" выбрана табличная секция.
- *Поле для поиска* — строковые поля секции справочника, по которым будет осуществляться поиск строки секции (объединяются по ИЛИ) данного типа.

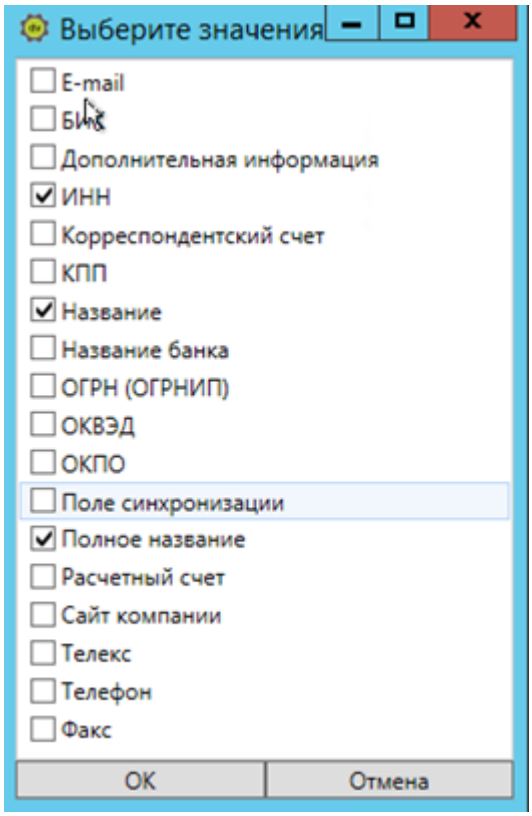


Рисунок 660. Выбор полей для поиска

- Дополнительные поля — список полей и их названий, которые будут отображены в выпадающем списке элементов и форме поиска, выбора

элементов и отображения значения. Появившееся окно позволяет добавить необходимое количество дополнительных полей. Для каждого добавленного поля можно задать отображаемое имя. Можно выбрать как простые поля, так и ссылочные.
Если в колонке "Отображать в выбранном значении" установлен флаг, значение данного поля отображается в скобках через запятую. Например, если в ЭУ выбраны подразделения, а в дополнительных полях указана почта с включенным флагом, то в итоге отображение будет таким: User (user@example.com).

- *Имя плагина для доп. полей* — можно указать плагин, который будет использован для получения дополнительных полей кодом. Это будет полезно для получения данных из, например, карточек сотрудников, подразделений, строк конструктора справочников, либо по какой-либо другой сложной логике. См. подробнее "Расширение возможностей ЭУ "Записи справочников"" в документации для разработчиков.

- Редактирование
  - Редактирование справочника — параметр, который в форме выбора из одного типа значений отображает кнопку добавления новых значений для справочников, поддерживающих редактирование.
  - Узел — ID узла, в котором должны создаваться записи по кнопке добавления новых значений. Если узел не указан, элементы будут добавляться в текущий выбранный узел (добавление доступно когда выбран только один тип значений).
- Ограничение
  - *Только дочерние элементы* — если флаг установлен и указано Ограничение, то поиск и выбор будут осуществляться только из дочерних элементов строки ограничения, а не всех потомков.
  - *Ограничение* — ID строки секции справочника, которая будет выступать ограничением при поиске строки секции (например, ID подразделения при выборе сотрудников).

    Если выбрана табличная секция, выбираемые элементы будут ограничены по полю `ParentRowID`. Если выбрана древовидная секция, то выбираемые элементы будут ограничены по полю `ParentTreeRowID`.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Выбор одного элемента

Позволяет настроить возможность выбора только одной строки справочника. Если выбор одного элемента запрещён, ЭУ должен быть привязан к табличной секции.

Не показывать недоступные значения

Даёт возможность выбирать строки справочника, помеченные как недоступные для выбора. Если поля `NotAvailable` нет в секции справочника, данная настройка игнорируется.

Поисковой индекс

Количество символов в текстовом поле, необходимое для начала поиска значений элементом управления.

Кол-во загружаемых за раз элементов

Количество результатов на "порцию" поиска. В зависимости от значения количество результатов поиска, отображаемых за один раз может изменяться.

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Показывать иконку

Свойство регулирует отображение иконки элемента управления. Когда флаг установлен (по умолчанию) иконка ЭУ отображается. Когда флаг снят, иконка не отображается.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

В зависимости от выбранного режима редактирования ЭУ отображается по-разному. См. подробнее "Записи справочников" в руководстве пользователя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

После закрытия окна справочника

Вызывается после закрытия окна выбора значения из справочника.

После открытия окна справочника

Вызывается после открытия окна выбора значения из справочника.

После загрузки результатов поиска

Вызывается после загрузки результатов поиска.

Перед закрытием окна справочника

Вызывается перед закрытием окна выбора значения из справочника.

Перед открытием окна справочника

Вызывается перед открытием окна выбора значения из справочника.

Перед загрузкой результатов поиска

Вызывается перед загрузкой результатов поиска.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования

По месту.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования ***По месту***.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

Стандартные

Web-клиент Конструктор Web-разметок

Стандартные

Описание элементов управления группы "Стандартные".

- Html тег
- Url
- Группа радиокнопок
- Другие элементы управления группы в навигационном меню слева…

Html тег

Элемент управления `Html тег` предназначен для добавления в разметку произвольного HTML-селектора с заданными содержимым. ЭУ не поддерживает запуск скриптов.

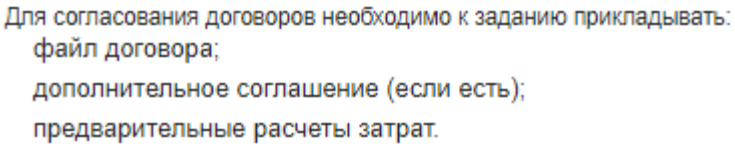


Рисунок 661. Пример отображения произвольного содержимого с помощью элемента управления "Html тег"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS-класс

Название CSS-класса, применяемого к ЭУ.

CSS-стиль

Название CSS-стиля, применяемого к ЭУ.

Содержимое

Содержимое, добавляемое внутрь HTML-тега.

Тэг

Название добавляемого HTML-селектора, например `div`.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Url

Элемент управления `Url` предназначен для отображения гиперссылки на страницу или почтовый адрес.

В настройках ЭУ может быть указан фиксированный адрес или поле карточки, из которого может быть получен адрес. Если указано поле карточки с адресом, то фиксированный адрес игнорируется.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст

Текст, замещающий адрес ссылки при отображении.

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки, содержащее адрес ссылки. Возможный тип поля: ***Text***, ***Unitext***, ***Unistring*** или ***String***.

Адрес

Фиксированный адрес ссылки. Если указано Поле данных, свойство Адрес игнорируется.

Протокол

Протокол адреса: `http://`, `https://`, `mailto:`. Свойство добавляет указанный префикс к адресу ссылки.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.

- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим открытия

Определяет способ открытия ссылки:

- ***Текущая вкладка***. Ссылка будет открыта в текущей вкладке.
- ***Новая вкладка***. Ссылка будет открыта в новой вкладке.
- ***Новое окно браузера***. Ссылка будет открыта в новом окне веб-браузера.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Группа радиокнопок

Элемент управления `Группа радиокнопок` предназначен для выбора и отображения значения из списка значений.

Элемент управления может быть связан с полем карточки следующих типов:

- ***int*** — предоставляется возможность самостоятельно создавать набор элементов, отображаемых в ЭУ.

- ***enum***, ***bool*** — набор значений (для enum) будет загружен из схемы карточки и изменению не подлежит. При типе поля ***enum*** в ЭУ загружается весь список значений, при типе поля ***bool*** — два значения: ***True*** и ***False***.

Независимо от типа связанного поля, предоставляется возможность локализовать значения элементов списка.

Вариант отображения ЭУ зависит от свойства `Режим редактирования`:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

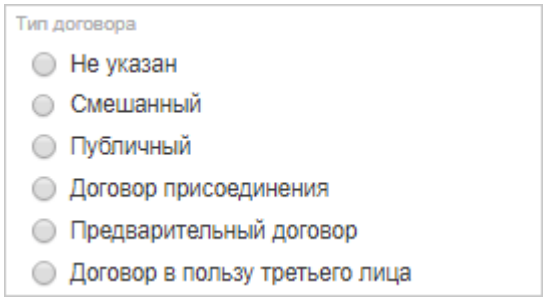


Рисунок 662. Элемента управления "Группа радиокнопок" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Изменение значения осуществляется при нажатии на ссылку в ЭУ.

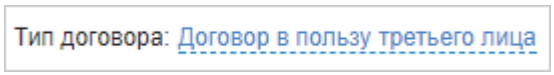


Рисунок 663. Элемент управления "Группа радиокнопок" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.



Чтобы настроенное при создании или редактировании карточки значение ЭУ корректно отображалось во время просмотра карточки, набор элементов в связанном списке должен быть идентичен в разметке режима редактирования/создания и в разметке режима просмотра. Создать ЭУ с идентичными настройками можно копированием исходного ЭУ.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Количество колонок

Количество колонок для отображения элементов списка. По умолчанию все элементы отображаются в одной колонке.

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки с данными элемента управления. Поле должно быть типа ***enum***, ***bool*** или ***int***.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Положение метки элемента

Положение текстовой части переключателя (относительно переключателя) в отображаемом элементе списка. Текстовая часть может отображаться справа или слева от переключателя.

Список элементов

Настройка списка элементов, отображаемых в элементе управления.

Список значений задаётся в форме, открываемой при нажатии кнопки ...:

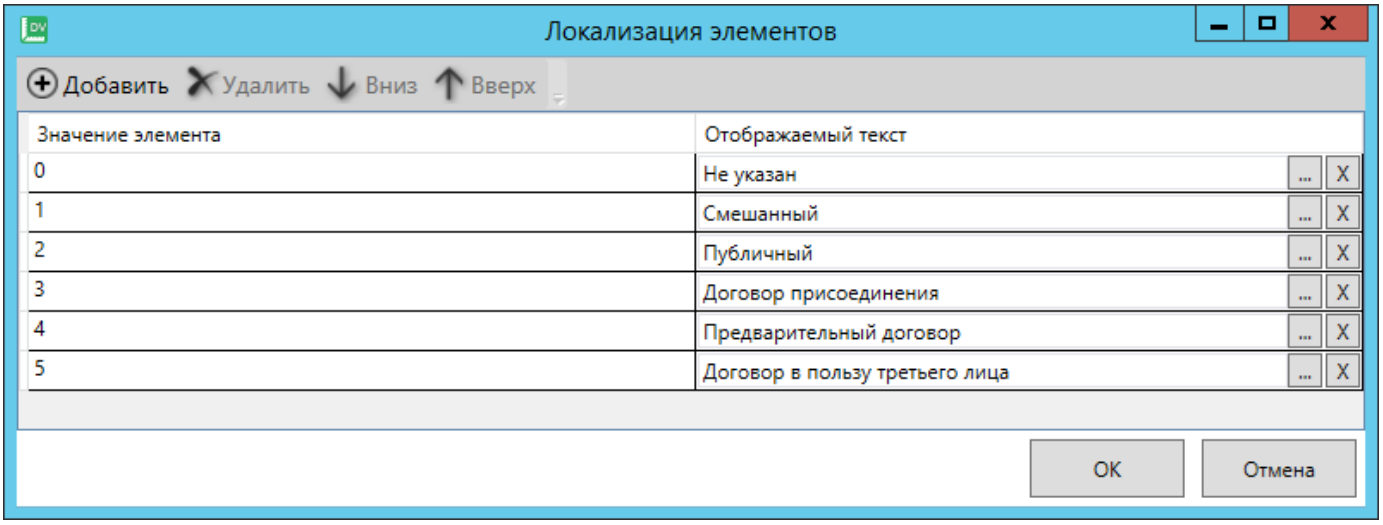


Рисунок 664. Настройка списка элементов элемента управления "Группа радиокнопок", связанного с полем типа "int"

Возможность создания собственного набора значений доступна только для ЭУ, связанного с полем типа ***int***.

Локализация элементов списка выполняется при нажатии кнопки ... в строке значения. Настройки локализации сохраняются после нажатия кнопки **ОК** в окне *Локализация элементов*.

Элемент по умолчанию

Элемент из набора значений ЭУ, который будет выбран по умолчанию в группе переключателей.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и

просмотра.

Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования ***По месту***.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования ***По месту***.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Дата/время

Элемент управления Дата/время предназначен для ввода и отображения даты или даты и времени из связанного поля карточки.

Интерфейс элемента управления зависит от свойства **Режим редактирования**:

- ***Редактирование***, В данном режиме ЭУ представляет собой поле для ввода значения, которое также можно выбрать из календаря (открывается кнопкой).
- В режиме ***По месту*** новое значение автоматически сохраняется в карточку.



Рисунок 665. Элемент управления "Дата/время" в режиме редактирования "По месту"

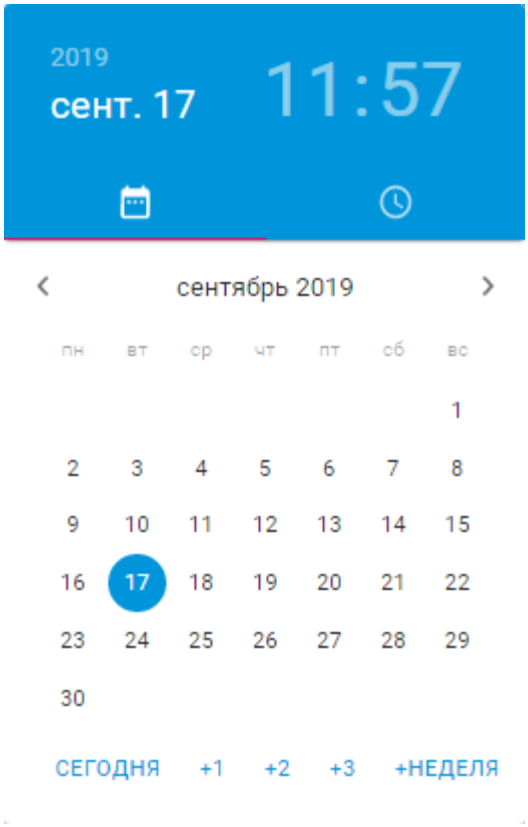


Рисунок 666. Пример календаря для выбора значения

- ***Без редактирования***. В данном режиме ЭУ представляет собой строку, в которой отображается значение даты и времени из поля карточки.



Рисунок 667. Элемент управления "Дата/время" в режиме без редактирования

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать диалог выбора времени

Флаг позволяет отключить или включить отображение диалога выбора времени для элемента управления. По умолчанию флаг активен, то есть диалог выбора времени отображается.

Когда флаг неактивен, пользователю доступен только календарь для выбора даты, время можно указать вручную с клавиатуры.



Диалог выбора времени доступен в разметках редактирования или редактирования по месту, только если в свойстве Тип выбран пункт **Дата и время**.

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсвечивать просроченность

Дата и время в ЭУ будут подсвечены красным цветом (значение по умолчанию), если установленный срок превышен.

- Флаг установлен. ЭУ будет подсвечен, если текущаие дата и/или время превышают значения ЭУ.

- Флаг не установлен. Сроки не проверяются, ЭУ не будет подсвечен.

ЭУ предоставляет возможность изменить цвет сигнализирования о просроченности. К примеру, чтобы установить зеленый цвет сигнализирования, создайте в папке `C:\Program Files\Docsvision\WebClient\Site\Content\Solutions\SampleSolution\Styles\` файл `sample.css` с содержимым:

```
.system-datetimepicker.highlight-overdue.overdue:not(.edit-mode) .labeled-text-helper
.text-cell .labeled-text { ①
font-weight: bold;
color: green;
}
```

① `not(.edit-mode)` — Данный стиль будет использован только при просмотре карточки.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Тип

Формат представления времени в элементе управления:

- ***Дата и время***. Отображаются и могут быть изменены дата и время.
- ***Дата***. Отображается и может быть изменена дата.

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле типа ***Дата/время*** с данными элемента управления.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение `Операции редактирования` наследуется от родительского `Блока`, название свойства меняется на `Операция редактирования (наследовано)`.

Значение по умолчанию



Свойства из категории "Значение по умолчанию" доступны только в режиме редактирования и недоступны в режимах просмотра и редактирования по месту.

Время по умолчанию

Свойство недоступно в режиме редактирования по месту, доступно только в режиме редактирования.

Время, которое будет установлено по умолчанию в ЭУ:

- ***Текущее время***. Будет установлено текущее время.
- ***Начало дня***. Будет установлено время `00:00`.
- ***Начало рабочего дня***. Будет установлено время начала рабочего дня из бизнес-календаря.
- ***Конец дня***. Будет установлено время `23:59`.
- ***Конец рабочего дня***. Будет установлено время конца рабочего дня из бизнес-календаря.

Если выбрано значение ***Начало рабочего дня*** или ***Конец рабочего дня***, то при расчете времени будет использован бизнес-календарь:

- Календарь по умолчанию, который выбран в справочнике *Системные настройки* (**Системные настройки** › **Делопроизводство** › **Задачи и задания** › **Расширенные настройки** › **Бизнес-календарь по умолчанию**).

- Если не выбран календарь по умолчанию, будет использован системный бизнес-календарь (рабочий день с 9 до 18 часов, суббота и воскресенье — выходной).

Значение по умолчанию

Свойство недоступно в режиме редактирования по месту, доступно только в режиме редактирования.

Дата и время, которые будут установлены по умолчанию в ЭУ. Флаг **Текущая дата** должен быть снят. Если в свойствах ЭУ установлено смещение значения по умолчанию, то значение ЭУ будет скорректировано с учётом этого параметра.

Если время не указано (время 00:00), то в значении ЭУ время будет получено из бизнес-календаря — время начала рабочего дня.[1].

Аналогичное поведение будет при работе с ЭУ в карточке: если стереть время, после выбора даты будет установлено время, полученное из бизнес-календаря.

Смещение значения по умолчанию (в часах)

Количество часов (целое число), которое будет прибавлено к предварительному значению ЭУ, вычисленому из значения свойств **Значение по умолчанию** или **Время по умолчанию**. Можно указывать отрицательное значение. Например, чтобы сдвинуть время по умолчанию на день вперед, в поле **Смещение значения по умолчанию** нужно ввести значение "24" (часа).

Текущие дата и время

- Когда флаг установлен, в ЭУ будет установлена по умолчанию текущая дата.
- Когда флаг не установлен, в ЭУ будет установлена по умолчанию дата из свойства ***Значение по умолчанию***.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Запретить выбор будущих дат

При установленном флаге пользователю будет запрещено устанавливать и выбирать из календаря даты, позже текущей. Если введена запрещённая дата, будет выдана ошибка.

Запретить выбор прошедшей даты

При установленном флаге пользователю будет запрещено устанавливать и выбирать из календаря даты, раньше текущей. Если введена запрещённая дата, будет выдана ошибка.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Показывать иконку

Свойство регулирует отображение иконки элемента управления. Когда флаг установлен (по умолчанию) иконка ЭУ отображается. Когда флаг снят, иконка не отображается.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Изображение

Элемент управления Изображение предназначен для показа изображений из данных карточки и загрузки изображений.



Рисунок 668. Пример элемента управления "Изображение"

Элемент управления поддерживает добавление только файлов формата JPEG (.jpeg, .jpg).

Для элемента управления можно разработать функцию, которая будет устанавливать изображение по умолчанию, если изображение не выбрано. См. пример в пункте Настройка разметки диалога профиля пользователя.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Высота

Определяет вертикальный размер элемента управления в пикселях.

В зависимости от размеров оригинального изображения при отображении оно будет увеличено или уменьшено с сохранением соотношения сторон. Часть изображения может быть обрезана по ширине.

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Заполнитель при перетаскивании

Текст, отображаемый в элементе управления при перетаскивании файла изображения с файловой системы в элемент управления.

Локализуемый

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Скругление

Устанавливает процент скругления углов изображения в процентах:

- `0` — скругление отсутствует,
- `50%` — максимальное скругление (значение по умолчанию).

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Может быть выбрана простая или коллекционная (табличная) секция. При выборе коллекционной секции элемент управления будет получать и изменять данные в первой строке секции.

Для полноценной работы с коллекционной секцией элемент управления *Изображение* нужно встроить в элемент управления "Таблица".

Поле данных

Поле карточки с данными элемента управления. Тип поля — ***image***.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Загрузка разрешена

Включает/отключает возможность загрузки изображений в карточку с помощью элемента управления.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.

- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед сбросом изображения

Вызывается при нажатии кнопки удаления изображения.

Получение изображения по умолчанию

Обработчик для получения изображения по умолчанию. Может быть указана собственная функция, возвращающая изображение по умолчанию для элемента управления.

После сброса изображения

Вызывается после нажатия кнопки удаления изображения.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка

Элемент управления Кнопка предназначен для вызова определённой пользовательской функции JavaScript.

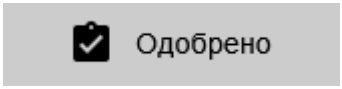


Рисунок 669. Пример отображения элемента управления "Кнопка"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на кнопке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
```

```
  width: 24px;
  height: 24px;
}
```

Основная кнопка

Включает/отключает использование для кнопки основного цвета карточки.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Растягивать на всю ширину

Когда флаг установлен, элемент управления растягивается на всю ширину родительского элемента. Когда флаг снят, ширина элемента управления соответствует содержимому.

Текст

Текст, отображаемый на кнопке.

Локализуемый

Данные

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопка-иконка

Элемент управления Кнопка-иконка предназначен для вызова определённой пользовательской функции JavaScript.



Данный ЭУ предназначен для использования в разметках локаций *Главная страница* и *Основное меню*.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

CSS класс иконки

Название CSS класса, предоставляющего иконку для отображения на кнопке. Если класс не указан, отображается только подпись без иконки.

Пример класса для предоставления иконки:

```
.button-icon {
  background-image: url('../../Solutions/SampleSolution/Images/ico-button.png');
  width: 24px;
```

```
  height: 24px;
}
```

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер локации

Элемент управления Контейнер локации предназначен для отображения данных карточки или другого элемента с разметкой в другой карточке.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Максимальная высота

Устанавливает ограничение по высоте для элемента управления. Значение в пикселях.

Данные

Источник локации

- ***Локация***. В элементе будет отображаться разметка из локации, выбранной в свойстве Локация.

- ***Связанная карточка/строка***. В элементе будет отображаться связанная карточка или данные строки связанной карточки. Ссылка на карточку содержится в поле, указанном в источнике данных. При отображении будет использована разметка с режимом из свойства Режим.



Свойство Локация доступно, если в Источник локации выбрано ***Локация***. Свойства Расширенный источник данных, Источник данных и Поле данных доступны, если в Источник локации выбрано ***Связанная карточка/строка***.

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле текущей или связанной карточки с идентификатором карточки или строки карточки, отображаемой в элементе управления.

Локация

Локация, разметка которой будет отображаться в элементе управления.

Параметры загрузки разметки

Дополнительные параметры, передаваемые при открытии разметки.

Поведение

Асинхронная загрузка

Определяет способ загрузки разметки локации.

- Разметка локации будет загружена после загрузки родительской разметки, когда флаг установлен.
- Разметка локации будет загружена вместе с родительской разметкой, когда флаг снят (потребуется больше времени для запуска родительской разметки).

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг

установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Пользовательская функция загрузки разметки

Название функции, переопределяющей механизм загрузки разметки локации. Указывается при необходимости.

Данная функция может быть использована, если нужно передавать в разметку собственные параметры, значение которых вычисляется во время исполнения.

Функция принимает единственный параметр с типом ***LocationContainerControl***. Описание типа см. в JSDocsApi.

Режим

Режим, в котором будет открываться разметка. Может принимать значение из списка: ***Любой***, ***Описание***, ***Справка***, ***Просмотр***, ***Редактирование***, ***Создание***, ***Внутри другой разметки***.

+ Свойство Режим используется при выборе подходящей разметки из списка разметок внедряемой карточки. Например, если установлен режим ***Редактирование***, то при внедрении будет использована разметка, в условии выбора которой указан режим **Редактирование**.

Использование специального режима ***Внутри другой разметки*** (он также доступен в условиях выбора разметки) позволяет отделить разметки, разработанные специально для отображения карточки внутри другой карточки, от разметок, используемых при обычной работе с карточкой (с режимами ***Просмотр***, ***Редактирование***, ***Создание***). При этом у внедряемой карточки должно быть условие выбора разметки с режимом ***Внутри другой разметки***.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Часть родительской разметки

Определяет режим интеграции разметки локации в родительскую разметку.

- Разметка локации будет загружена как часть родительской локации, когда флаг установлен (элементы управления разметки будут добавлены в родительскую разметку),
- Разметка локации не будет интегрирована в родительскую разметку, когда флаг снят.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Метка

Элемент управления `Метка` предназначен для отображения любой статической (не редактируемой и не связанной с данными карточки) строки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст

Отображаемый в ЭУ.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Раскрывающийся список

Элемент управления Раскрывающийся список предназначен для выбора и отображения значения из списка значений.

Элемент управления может быть связан с полем карточки следующих типов:

- ***int*** — предоставляется возможность самостоятельно создавать набор элементов, отображаемых в ЭУ.
- ***enum***, ***bool*** — набор значений (для enum) будет загружен из схемы карточки и изменению не подлежит. При типе поля ***enum*** в ЭУ загружается весь список значений, при типе поля ***bool*** — два значения: ***True*** и ***False***.

Независимо от типа связанного поля, предоставляется возможность локализовать значения элементов списка.

Вариант отображения ЭУ зависит от свойства Режим редактирования:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.



Рисунок 670. Элемент управления "Раскрывающийся список" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Изменение значения осуществляется при нажатии на ссылку в ЭУ.

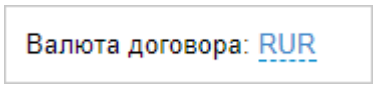


Рисунок 671. Элемент управления "Раскрывающийся список" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.



Чтобы настроенное при создании или редактировании карточки значение ЭУ корректно отображалось во время просмотра

> карточки, набор элементов в связанном списке должен быть идентичен в разметке режима редактирования/создания и в разметке режима просмотра. Создать ЭУ с идентичными настройками можно копированием исходного ЭУ.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Поле карточки с данными элемента управления. Поле должно быть типа ***enum***, ***bool*** или ***int***.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Список элементов

Настройка списка элементов, отображаемых в элементе управления.

Список значений задаётся в форме, открываемой при нажатии кнопки ...:

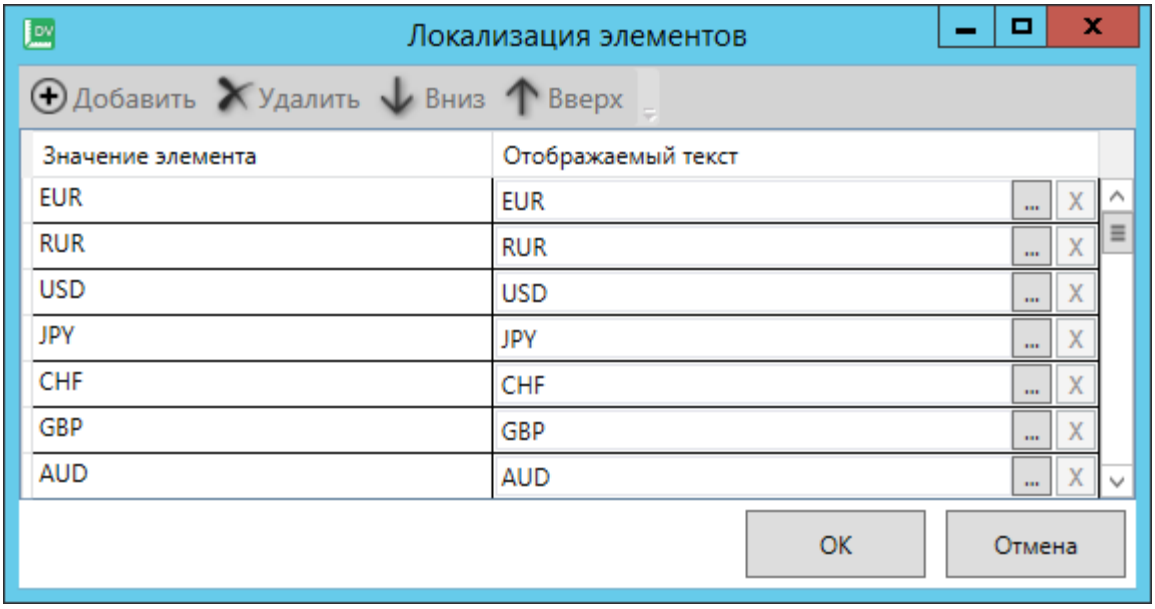


Рисунок 672. Настройка списка элементов элемента управления "Раскрывающийся список", связанного с полем типа "enum"

Возможность создания собственного набора значений доступна только для ЭУ, связанного с полем типа ***int***. При сохранении карточки в связанное поле будет записано соответствующее целочисленное значение.

Локализация элементов списка выполняется при нажатии кнопки ... в строке значения. Настройки локализации сохраняются после нажатия кнопки **OK** в окне *Локализация элементов*.

Элемент по умолчанию

Элемент из набора значений ЭУ, который будет выбран по умолчанию в группе переключателей.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования ***По месту***.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования ***По месту***.

Перед сворачиванием раскрывающегося списка

Вызывается перед закрытием списка элементов при выборе значения.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

После открытия раскрывающегося списка

Вызывается после открытия списка элементов при выборе значения.

После сворачивания раскрывающегося списка

Вызывается после закрытия списка элементов при выборе значения.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При открытии раскрывающегося списка

Вызывается перед открытием списка элементов при выборе значения.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Строка

Элемент управления `Строка` предназначен для ввода и отображения строковых данных карточки. В отличие от ЭУ "Текст" может содержать только одну строку.

Вариант отображения ЭУ зависит от свойства **`Режим редактирования`**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

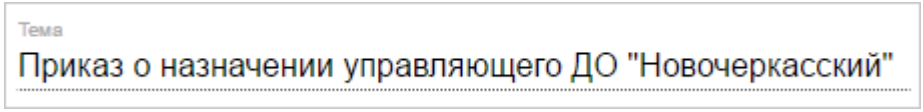


Рисунок 673. Элемент управления "Строка" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

Изменение текста осуществляется при нажатии на ссылку в ЭУ.

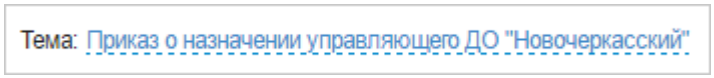


Рисунок 674. Элемент управления "Строка" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме

редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Текстовое поле карточки с данными элемента управления.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Значение по умолчанию

Значение элемента управления, присваиваемое при создании новой карточки.

Обрезать пробелы

Определяет необходимость удаления пробелов в начале и в конце строки.

- Пробелы удаляются при отображении и сохранении значения, если флаг установлен. Пробелы не удаляются, когда флаг снят.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Маска ввода

Свойство предназначено для написания регулярных выражений (regexp) для задания масок с использованием синтаксиса языка Perl. Свойство доступно только для режимов редактирования *Редактирование* и *По месту*. Значение можно задавать в редакторе, нажав кнопку ...

Структура окна редактора метки ввода:

Поле "Регулярное выражение"

В поле указывается значение маски ввода.

Поле "Маска"

В поле указывается маска ввода.

Область для проверки

Область предназначена для ввода проверочных значений.

В окне редактора пользователь может ввести только регулярное выражение, только маску или и то и другое сразу. Для масок существует два ограничения: "или" (например, +7 или 8 в номере телефона) и бесконечное количество символов.

Если значение соответствует регулярному выражению и маске, фон области подсвечивается зелёным, текст становится оранжевым. Если значение соответствует только маске, только текст становится оранжевым. Если значение соответствует только регулярному выражению, только фон подсвечивается зелёным. Если совпадения не обнаружены, цвет текста и фона не меняется, ошибка не выводится.

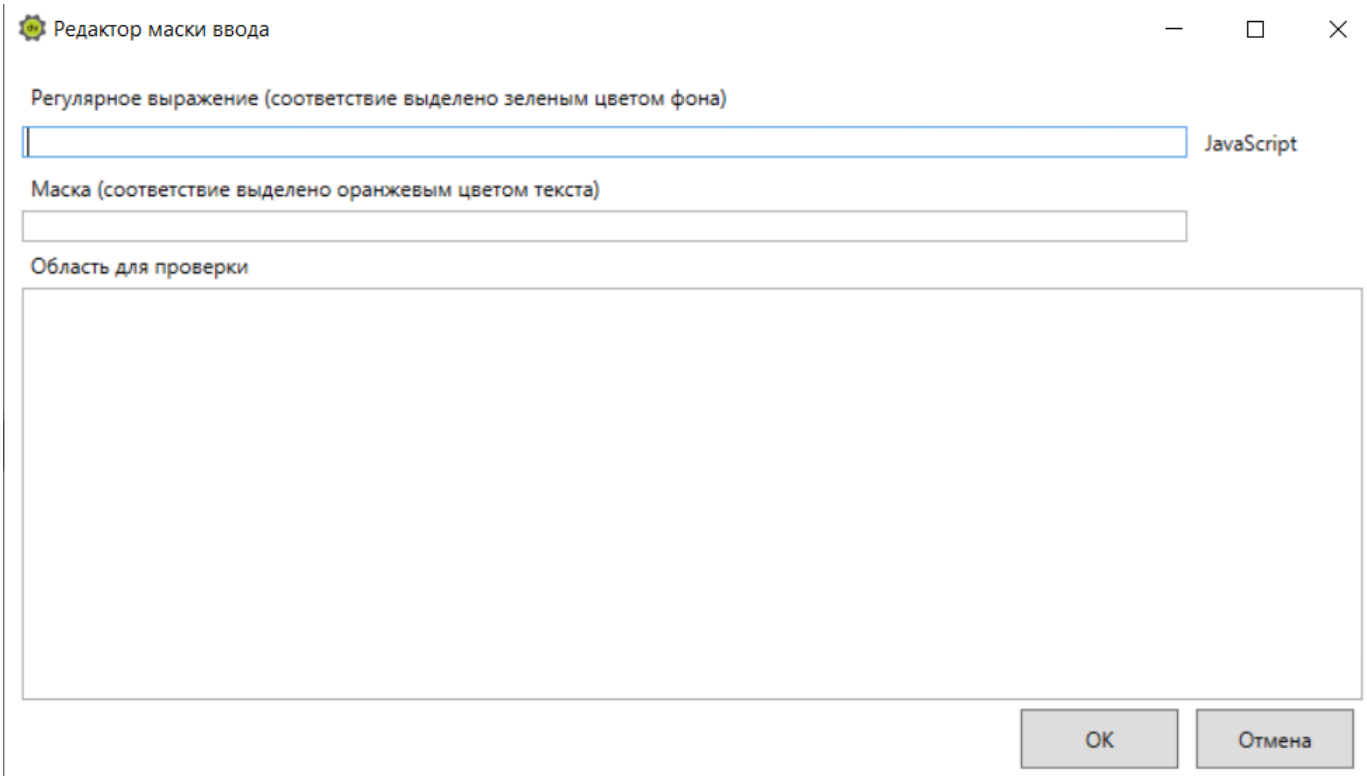


Рисунок 675. Пример указания и проверки маски ввода для номера телефона

Примеры указания регулярных выражений и масок ввода:

- Для телефона:
  - Регулярное выражение: `((8|+7)[- ]?)?((?\d{3})?[- ]?)?[\d- ]{7,9}`
  - Формат маски ввода: `\+7(000)000-00-00` или `8(000)000-00-00`, где `0` — цифра от 0 до 9
- Для СНИЛС:
  - Регулярное выражение: `^\d{3}-\d{3}-\d{3} \d{2}$`
  - Формат маски ввода: `000-000-000 00`, где `0` — цифра от 0 до 9
- Для адреса электронной почты:
  - Регулярное выражение: `([a-zA-Z0-9.-]+@[a-zA-Z0-9.-]+.[a-zA-Z0-9_-]+)`
  - Маску ввода задать нельзя, т.к. количество символов не ограничено.
- Для ИНН физического лица:
  - Регулярное выражение: : `(?!00)\d{12}$`. Не может начинаться с двух нулей, т.к. нет субъекта РФ с кодом `00`.
  - Формат маски ввода: `000000000000`, 12 циф.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования ***По месту***.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования ***По месту***.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Текст

Элемент управления `Текст` предназначен для ввода и отображения строковых и форматированных текстовых данных карточки.

Вариант отображения ЭУ зависит от свойств **Режим редактирования** и **Использовать HTML редактор**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

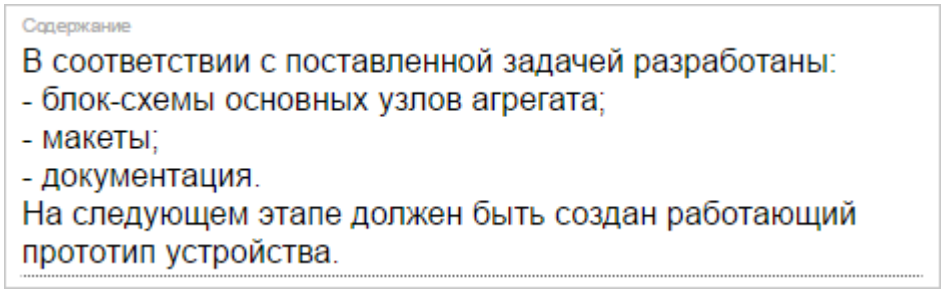


Рисунок 676. Элемент управления "Текст" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Изменение текста осуществляется при нажатии на ссылку в ЭУ.

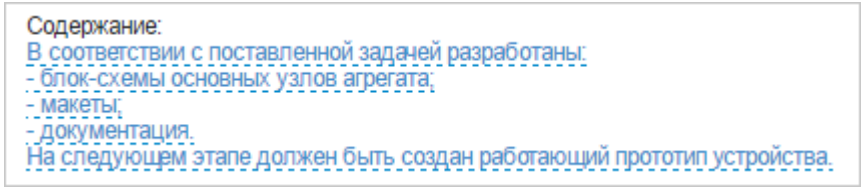


Рисунок 677. Элемент управления "Текст" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

При работе в режиме HTML редактора, ЭУ предоставляет дополнительные возможности форматирования:

- Настраивать стиль текста: Нормальный, Заголовок 1, Заголовок 2 и пр.; начертания: жирный, курсив; шрифта, размера шрифта и фоновой заливки текста.
- Использовать маркированные и нумерованные списков.
- Настраивать выравнивание текста.
- Добавлять таблицы, ссылки, рисунки и видео (видео с внешних ресурсов не поддерживается).
- Перетаскивать окно редактора в режиме редактирования ***По месту***.

Когда свойство `Использовать HTML редактор` активно, в режиме редактирования "По месту" для ЭУ отображается маленькое всплывающее окно.

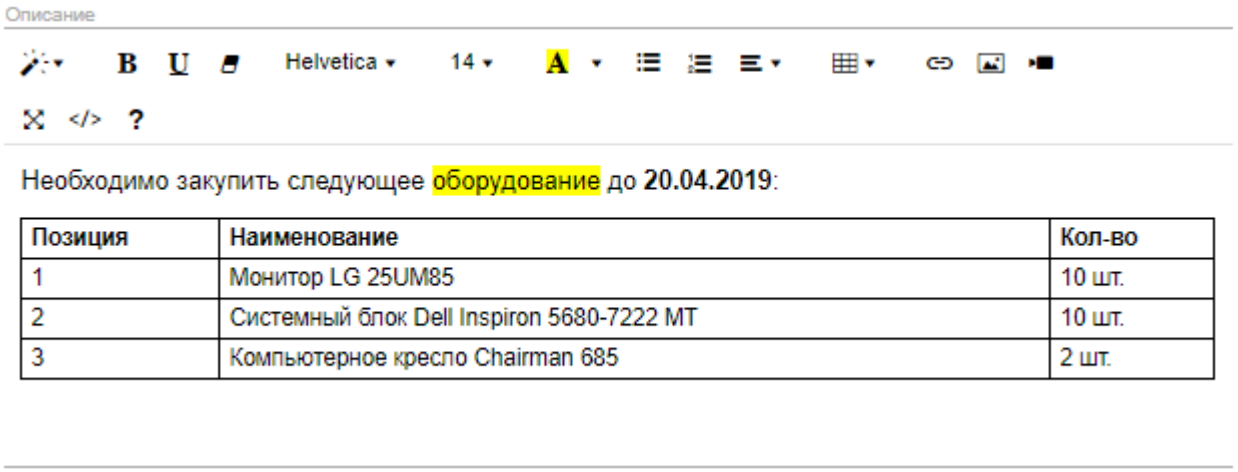


Рисунок 678. Элемент управления "Текст" в режиме HTML редактора

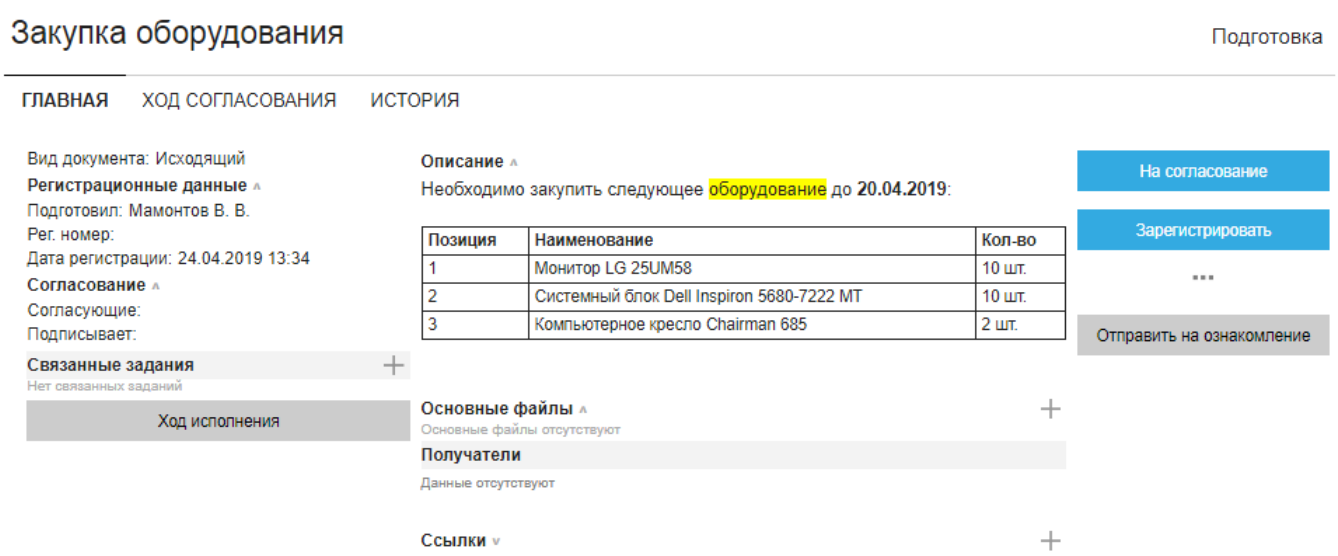


Рисунок 679. Пример отображения содержимого ЭУ "Текст"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.

- Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.

- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Текстовое поле карточки с данными элемента управления.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Высота HTML редактора

Высота элемента управления при использовании HTML редактора. Значение указывается в пикселях. Значение по умолчанию — `200px`.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Значение по умолчанию

Значение элемента управления, присваиваемое при создании новой карточки.

Использовать HTML редактор

Переключает элемент управления в режим с возможностью форматирования текста и добавления мультимедийных элементов.

В Windows-клиенте и PMA HTML всегда отображается в виде строки кода.

Лимит синхронной загрузки (Кб)

Определяет максимальный размер данных, загружаемых синхронно с элементом управления. Значение указывается в килобайтах.

Если размер данных больше указанного значения, используется асинхронная загрузка (для всего объема данных).

Максимальный размер изображения (Кб)

Максимальный размер для загружаемых изображений. Значение указывается в килобайтах.

Если размер загружаемого пользователем изображения больше указанного значения, будет выдана ошибка и изображение не будет загружено.

Обрезать пробелы

Определяет необходимость удаления пробелов в начале и в конце строки.

- Пробелы удаляются при отображении и сохранении значения, если флаг установлен. Пробелы не удаляются, когда флаг снят.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина HTML редактора

Относительная ширина окна HTML редактора (если включено свойство `Использовать HTML редактор`), устанавливаемое для режима редактирования "По месту".

События

Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования ***По месту***.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования ***По месту***.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По месту***.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Файл

Элемент управления Файл предназначен для работы с файлами карточек любых типов.

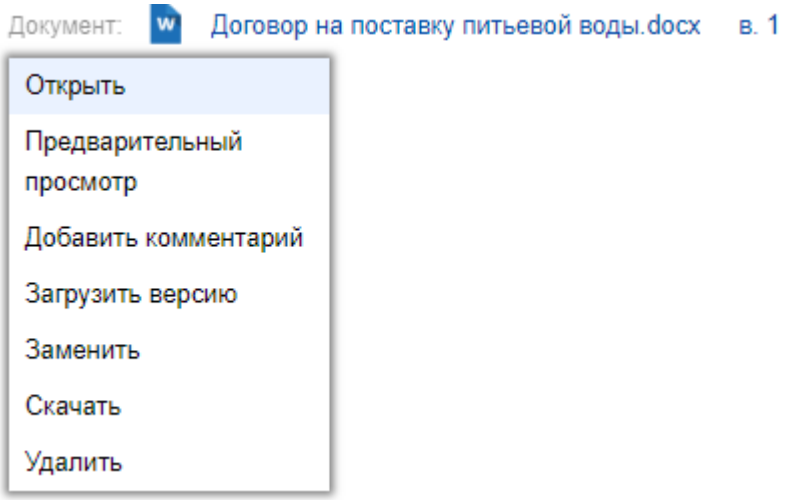


Рисунок 680. Пример элемента управления "Файл"

Данный элемент управления позволяет:

- Добавлять/удалять/скачивать файлы.
- Работать со списком версий.
- Комментировать версии.
- Просматривать содержимое файла в окне предварительного просмотра.
- Просматривать содержимое .zip-архивов: видеть виды файлов и их количество, просмотр файлов в архиве недоступен.

Работа с файлом осуществляется с помощью команд контекстного меню (открывается при щелчке по названию файла).

При щелчке по номеру версии (в. 1 на рисунке выше) будет открыт список версий файла с комментариями.

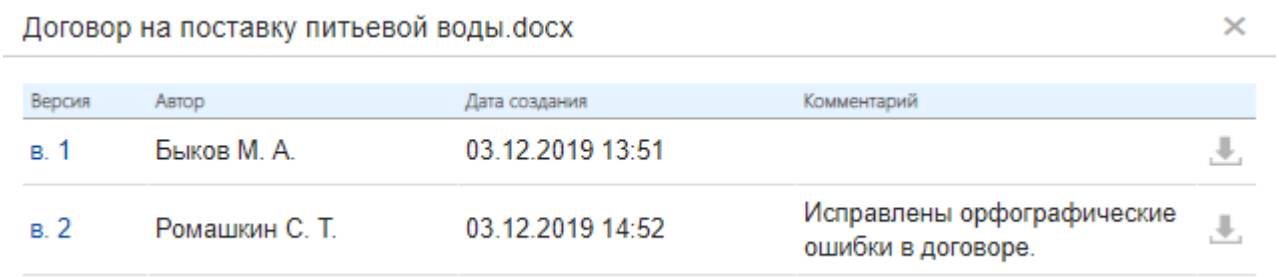


Рисунок 681. Версии файла с комментариями

Из данного окна можно скачать любую версию файла или открыть её в режиме предварительного просмотра.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Заполнитель при перетаскивании

Текст, отображаемый в элементе управления, когда файл перетаскивается с файловой системы в область ЭУ.

Локализуемый

Отображать метку без значения

Регулирует отображение метки, когда файл не загружен.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Сообщение при пустом значении

Сообщение, выводимое при попытке сохранить пустое значение в поле выбора файла.

Локализуемый

Текст метки

Содержимое метки, отображаемой около элемента управления. При размещении элемента управления в Таблице метка будет скрыта.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Файл может быть связан с плоской или табличной секцией (для добавления нескольких файлов). При связывании с табличной секцией Файл будет отображать данные из первой строки секции, если элемент управления Файл размещается не в ЭУ Таблица (см. Настройка элемента управления для работы с несколькими файлами).

Поле данных

Ссылочное поле, указывающее на карточку версионного файла.

Операция на скачивание

Операция редактирования, при доступности которой пользователю будет разрешено скачивать файл. Если операция не выбрана, возможность скачивания файла не ограничена.

Операция на чтение

Операция редактирования, при доступности которой пользователю будет разрешено открывать файл и использовать предварительный просмотр. Если операция не выбрана, возможность просмотра файла не ограничена.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Включить версионинг

Включает или отключает возможности версионирования, предоставляемые элементом управление: просмотр, добавление и комментирование версий.

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Доступные расширения

Список разрешённых для загрузки расширений.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Показывать контекстное меню

Регулирует отображение контекстного меню.

Режим редактирования

Определяет возможности настройки элемента управления и список операций в контекстном меню ЭУ. Возможные варианты:

- ***По месту***. Доступные операции: "Открыть", "Предварительный просмотр", "Добавить комментарий", "Загрузить версию", "Заменить", "Скачать".
- ***Редактирование***. Доступные операции: аналогично режиму ***По месту***.
- Без редактирования. Доступные операции: "Открыть", "Предварительный просмотр", "Комментировать", "Скачать".



Отдельные пункты меню могут быть недоступны, если соответствующая операция редактирования недоступна и/или отключен версионирования.



Особенность поведения: в режиме редактирования ***По месту*** элемент управления предоставляет возможность управлять версиями файла. В том числе при создании новой карточки.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед добавлением комментария к версии файла

Вызывается перед добавлением комментария к версии файла.

Перед закрытием диалога комментариев

Вызывается перед закрытием диалогового окна ввода комментария к версии файла.

Перед закрытием диалога просмотра версий

Вызывается перед закрытием диалогового окна со списком версий файла.

Перед открытием диалога комментариев

Вызывается перед открытием диалогового окна ввода комментария к версии

файла.

Перед открытием диалога просмотра версий

Вызывается перед открытием диалогового окна со списком версий файла.

Перед открытием меню файла

Вызывается перед открытием файлового меню.

Перед открытием предпросмотра

Вызывается перед открытием предварительного просмотра файла.

Перед открытием файла

Вызывается перед открытием файла.

Перед скачиванием файла

Вызывается перед скачиванием файла.

Перед удалением комментария к версии файла

Вызывается перед удалением комментария к версии файла.

После добавления комментария к версии файла

Вызывается после добавления комментария к версии файла.

После загрузки информации о версиях файла

Вызывается после загрузки информации о версиях файла в элемент управления (в окне списка версий).

После закрытия диалога комментариев

Вызывается после закрытия диалогового окна ввода комментария к версии файла.

После закрытия диалога просмотра версий

Вызывается после закрытия диалогового окна со списком версий файла.

После открытия диалога комментариев

Вызывается после открытия диалогового окна ввода комментария к версии файла.

После открытия диалога просмотра версий

Вызывается после открытия диалогового окна со списком версий файла.

После открытия предпросмотра

Вызывается после открытия предварительного просмотра файла.

При сохранении файла в WebDAV

Вызывается при сохранении файла с использованием WebDAV.

После открытия файла

Вызывается после открытия файла.

После скачивания файла

Вызывается после скачивания файла.

После смены данных

Вызывается после любого изменения (выбора, загрузки, удаления) данных элемента управления.

После удаления комментария к версии файла

Вызывается после удаления комментария к версии файла.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При сохранении файла в WebDAV

Вызывается при сохранении файла в WebDAV.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Файл выбран

Вызывается после выбора файла с помощью элемента управления или добавления файла перетаскиванием.

Файл загружен

Вызывается при успешном сохранении файла на сервере.

Файл удалён

Вызывается после очищения значения элемента управления.

См. также: **Настройка элемента управления для работы с несколькими файлами**.

Настройка элемента управления для работы с несколькими файлами

Элемент управления "Таблица" может быть использован для работы с табличной секций, содержащей ссылки на карточки Файла с версиями.

Чтобы настроить таблицу для работы с файлами:

1. Добавьте в разметку карточки элемент управления Таблица.
2. В свойстве **Источник данных** выберите табличную секцию карточки, содержащую ссылки на карточки файла с версиями. Например, для *Документа* можно указать секцию *Файлы*.



Поле в не типовой табличной секции должно обязательно иметь настройки: "Ссылка" — карточка файла с версиями, "Тип ссылки" — сильная.

3. Установите флаг **Скрыть кнопку добавления строк**.
4. В зависимости от назначения разметки установите **Режим редактирования**:
   - ***Редактирование***. Для разметки создания или редактирования карточки,



Допускается добавление элемента с режимом ***Редактирование*** в разметку, предназначенную для просмотра данных. В этом случае необходимо самостоятельно реализовать функцию сохранения значения элемента управления с использованием скриптов карточки.

   - ***По месту***. Для просмотра карточки.
5. В свойстве **Минимальная высота** ЭУ Таблица можно установить собственный размер для блока приёма файла. Блок будет отображаться в ЭУ Таблица при добавлении файлов перетаскиванием с устройства.

6. Если требуется, заполните другие свойства Таблицы (например, заголовок).
7. Добавьте в таблицу элементы управления:
   - "Таблица / Добавить файл". В свойствах можно указать текст, который будет отображаться при добавлении файла перетаскиванием, отключить режим добавления перетаскиванием, а также ограничить работу с функцией с помощью операции редактирования.
   - "Таблица / Скачать все". В настройках можно ограничить возможность использования функции с помощью операции редактирования.



- Если возможность добавления и/или скачивания всех файлов не требуется, указанные элементы управления можно не добавлять.
- ЭУ Файл в разметке просмотра должен быть в Режиме *Без редактирования*.

8. Добавьте в таблицу ЭУ "Столбец таблицы", в котором будут отображаться файлы.



При необходимости можно добавить столбцы для отображения других данных из секции, выбранной в настройках таблицы.

9. В ЭУ Столбец таблицы добавьте элемент управления "Файл", который будет использоваться для добавления и отображения добавленных файлов.
10. В свойстве **Поле данных** ЭУ Файл выберите поле, содержащее ссылку на карточку Файла с версиями. В свойстве **Источник данных** должна быть указана секция, которая была выбрана источником данных Таблицы.
11. В зависимости от назначения разметки установите **Режим редактирования**:
    - ***Редактирование***. Для разметки создания/редактирования или просмотра (сохранение значения реализуется с помощью скриптов) карточки.
    - ***По месту***. Для просмотра карточки.
12. Снимите флаг **Включить версионинг**, если не требуется работа с версиями файла (будут скрыты команды работы с версиями и номер версии файла).
13. Если требуется, заполните другие свойства элемента управления (например, **Обязательность**).

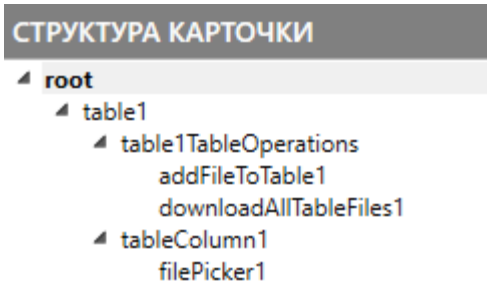


Рисунок 682. Пример разметки карточки с таблицей, настроенной для работы с файлами

14. Сохраните разметку.

Флажок

Элемент управления Флажок предназначен для просмотра и изменения булевых данных (да/нет) карточки.

Вариант отображения ЭУ зависит от свойства **Режим редактирования**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

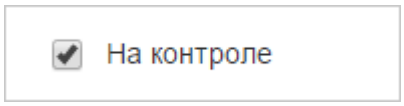


Рисунок 683. Элемент управления "Флажок" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  В карточке выглядит аналогично режиму редактирования.

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

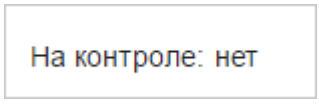
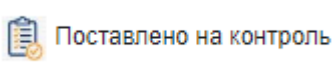


Рисунок 684. Элемент управления "Флажок" в режиме чтения

  В зависимости от настроек, ЭУ в режиме чтения также может содержать иконку, соответствующую значению, и собственный текст для значения.



Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Рисунок для значения "да"

Иконка, отображаемая в элементе управления при просмотре карточки, если флажок установлен. Иконка отображается слева от текста значения **да**.

Выбор иконки осуществляется при нажатии кнопки **...** в данном поле.

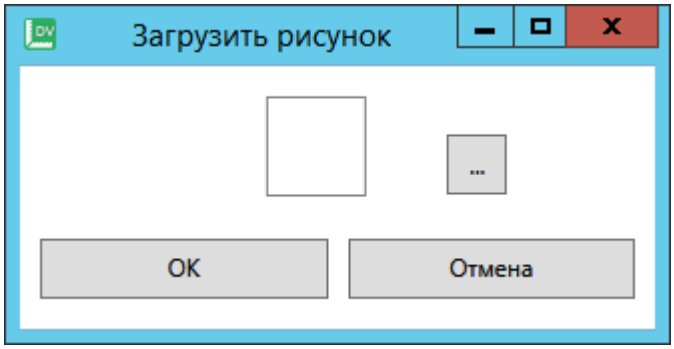


Рисунок 685. Окно выбора файла изображения для рисунка значения

Поддерживаются изображения следующих форматов: `.jpeg`, `.png`, `.bmp`, `.gif` (без анимации), `.tiff`, `.ico`, `.jxr`. Рекомендуемый размер изображения: 28 пикселей в высоту.

Рисунок для значения "нет"

Иконка, отображаемая в элементе управления при просмотре карточки, если флаг установлен. Иконка отображается слева от текста значения **нет**.

Выбор рисунка осуществляется аналогично выбору рисунка для значения **да**.

Скрыть поле для

Определяет значение элемента управления, при котором он должен быть скрыт:

- ***Значение не указано***. Если значение отсутствует.
- ***Да***. Если в значении ***true***.
- ***Нет***. Если в значении ***false***.
- ***Не скрывать***. Отображение элемента управления не зависит от значения).

Текст для значения "да"

Текст, отображаемый в режиме ***Без редактирования***, если Флажок установлен.

Локализуемый

Текст для значения "нет"

Текст, отображаемый в режиме редактирования ***Без редактирования***, если флаг не установлен.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Булево поле карточки, с которым связывается элемент управления.

Операция редактирования

Выбор операции для редактирования значения элемента управления. Если операция недоступна пользователю, то изменения списка ссылок или добавления комментария к ссылке будет недоступно. Если операция не выбрана, то возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента

управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Значение по умолчанию

Значение ЭУ, устанавливаемое по умолчанию при создании карточки.

- Когда флаг установлен в свойстве, он установлен также в ЭУ.
- Когда флаг не установлен в свойстве, в ЭУ он по умолчанию снят.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Число

Элемент управления `Число` предназначен для ввода и отображения числовых данных карточки. Позволяет работать с целыми и дробными числами.

Данный ЭУ поддерживает максимальную длину числа 28 символов: 14 цифр до точки и 14 после, т.е. самое большое число будет `99999999999999.99999999999999`.

Вариант отображения ЭУ зависит от свойства **`Режим редактирования`**:

- ***Редактирование***. Данный вариант может быть использован только в разметке режима редактирования карточки.

  Рисунок 686. Элемент управления "Число" в режиме редактирования

- ***По месту***. Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

  Изменение значения осуществляется при нажатии на ссылку в ЭУ.

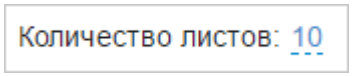
  

  Рисунок 687. Элемент управления "Число" в режиме редактирования по месту

- ***Без редактирования***. ЭУ в данном режиме отображает текстовое значение, соответствующее состоянию булевых данных в связанном поле карточки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заполнитель

Текст, отображаемый в элементе управления, если значение не выбрано.

Локализуемый

Отображать метку без значения

Определяет вариант отображения метки.

- Для режима просмотра:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если задано значение ЭУ.
- Для режима редактирования:
  - Когда флаг установлен, текст метки отображается всегда.
  - Когда флаг не установлен, текст метки отображается, только если поле содержит значение или редактируется. Для полей в режиме редактирования рекомендуется указывать Заполнитель.

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Секция карточки, содержащая данные элемента управления.

Поле данных

Числовое поле карточки с данными элемента управления. Для выбора доступны поля типа ***decimal***, ***float*** и ***integer***.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Значение по умолчанию

Число, которое будет указано в поле по умолчанию.

Использовать системные разделители

Флаг регулирует отображение разделителей числа. Свойство по умолчанию

активно, т.е. разделители отображаются.

Количество знаков после запятой

Определяет количество знаков после запятой.



Свойство доступно только для дробных чисел и не влияет на сохранение значения. Количество знаков после запятой учитывается только при отображении, само значение сохраняется полностью.

Обязательное

Определяет требование к заполнению значения ЭУ до сохранения карточки:

- Когда флаг установлен, значение ЭУ должно быть присвоено, иначе карточка не будет сохранена. При этом ЭУ помечается предупреждающим сообщением.
- Когда флаг не установлен, присваивать значение необязательно.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед закрытием окна редактирования

Вызывается перед закрытием окна редактирования в режиме редактирования ***По месту***.

Перед открытием окна редактирования

Вызывается перед открытием окна редактирования в режиме редактирования ***По месту***.

После закрытия окна редактирования

Вызывается после закрытия окна редактирования в режиме редактирования ***По месту***.

После открытия окна редактирования

Вызывается после открытия окна редактирования в режиме редактирования ***По***

месту.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

После смены данных

Вызывается после изменения содержимого элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Таблица

Web-клиент Конструктор Web-разметок

Таблица

Описание элементов управления группы "Таблица".

- Столбец таблицы
- Таблица
- Таблица / Добавить файл
- Другие элементы управления группы в навигационном меню слева…

Столбец таблицы

Элемент разметки `Столбец таблицы` предоставляет доступ к данным поля табличной секции карточки, являющейся источником данных для родительского

ЭР "Таблица".



ЭР **Столбец таблицы** может быть встроен только в ЭР **Таблица**.

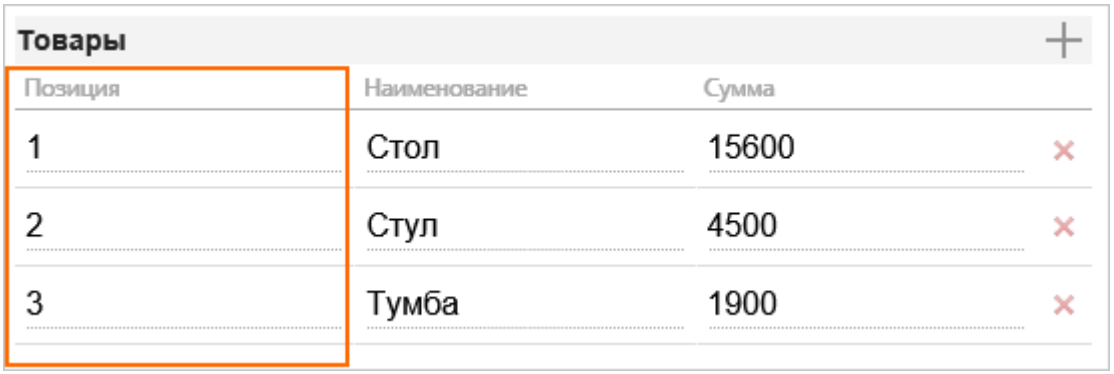


Рисунок 688. Пример элемента разметки "Столбец таблицы" в таблице

Столбец таблицы не связывается непосредственно с полем секции карточки, а выступает контейнером для размещения другого ЭУ, который и будет связан с данным полем. Тип встраиваемого ЭУ должен соответствовать типу отображаемых или редактируемых данных.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заголовок

Текст, отображаемый в заголовке данного столбца.

Локализуемый

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Ширина

Ширина столбца. Значение может быть задано в пикселях, процентах или масштабируемых единицах, зависящих от размера шрифта (например: 2em). Если значение не указано, то ширина столбца определяется автоматически.



Единица, в которой определяется ширина столбца, должна быть явно указана вместе со значением, например: `20%`, `30px` или `4em`.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства `Видимость`:

- Когда флаг `Видимость` установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг `Видимость` установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг `Видимость` НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Таблица

Элемент разметки Таблица предназначен для ввода и отображения данных табличной секции карточки.

Столбцы в ЭР Таблицы представляют собой отдельные элементы разметки — "Столбец таблицы". В один ЭР Таблица может быть добавлено необходимое количество столбцов.

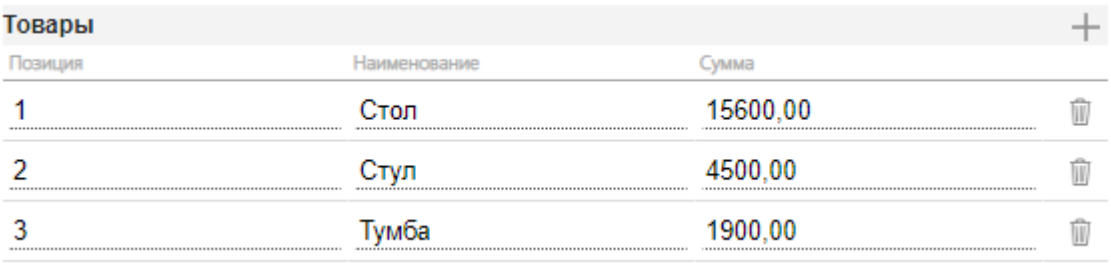


Рисунок 689. Пример элемента разметки "Таблица"

Таблица предоставляет возможность добавлять новые строки кнопкой + и удалять существующие кнопкой 🗑. Данные кнопки отображаются, если разрешено редактирование таблицы. Кнопка добавления строки может быть скрыта.

Также в таблицу могут быть добавлены новые команды.



Способ сохранения изменений таблицы отличается в зависимости от режима работы с ней:

- Если карточка открыта с разметкой режима просмотра (таблица также в режиме просмотра), изменение данных в ячейке сохраняется сразу.
- Если карточка открыта в режиме редактирования, то все изменения таблицы сохраняются при сохранении карточки.

Количество отображаемых в ЭР строк ограничивается настройками (по умолчанию — 15), при этом настоятельно не рекомендуется связывать ЭР с секцией, содержащей более 30 строк т.к. это может привести к увеличению времени открытия карточки.

По умолчанию строки в таблице не сортируются, а отображаются в порядке следования в связанной секции или в соответствии с реализованным разработчиком алгоритмом. Пользователю может быть предоставлена возможность вручную изменять порядок отображения строк в таблице (не сортировка) путём их перетаскивания.

Чтобы изменить порядок строк в таблице:

- В секции, связанной с таблицей, должно быть числовое поле (целое число), в котором будет сохраняться порядок строк.
- Данное поле должно быть выбрано в свойстве `Сохранять номер последовательности`.



Изменять порядок отображения строк возможно в режимах редактирования ***По месту*** или ***Редактирование***. Чтобы отображать строки с учётом заданной пользователем последовательности необходимо указать поле с порядком строк в свойстве `Сохранять номер последовательности` разметок редактирования и просмотра. В режиме редактирования порядок строк обновляется не моментально, а только после сохранения карточки.

Пример использования данного ЭУ в разметке рассмотрен в разделе Использование таблицы для работы с коллекционной секцией карточки.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заголовок

Текст, отображаемый в названии таблицы.

Локализуемый

Скрыть кнопку добавления строк

При включении скрывает из пользовательского интерфейса кнопку +.

Данные

Расширенный источник данных

Выбор типа источника данных элемента управления: текущая карточка или один из типов, настроенных в корневом элементе разметки.

Источник данных

Коллекционная секция карточки, содержащая данные для элемента управления.

Операция редактирования

Выбор операции редактирования значения элемента управления. Если операция недоступна пользователю, изменение значения элемента управления также недоступно. Если операция не выбрана, возможность редактирования значения элемента управления не проверяется.

Если значение Операции редактирования наследуется от родительского Блока, название свойства меняется на Операция редактирования (наследовано).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Максимальное количество строк для отображения

Определяет максимальное число отображаемых в *ЭР* строк (по умолчанию — 15). Если число строк в связанной секции больше, чем данное ограничение, в конце таблицы отображается сообщение: *There are more rows, that are not shown due to table settings*.

Минимальная высота

Минимальная высота элемента управления.

Адаптивный

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством Операция редактирования. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Отображать свернутым

Определяет начальное состояние Таблицы:

- Когда флаг установлен, содержимое Таблицы не отображается в карточке.
- Когда флаг не установлен, содержимое Таблицы отображается в карточке.

Проверяется только при установленном флаге "Сворачиваемый"

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Режим редактирования

Определяет вариант отображения элемента управления и возможность изменения его значения:

- ***По месту***. Значение изменяется нажатием на элемент управления.

  Данный вариант подходит как для разметки режима редактирования, так и для разметки режима просмотра карточки.

- ***Редактирование***. Значение изменяется непосредственно в элементе управления.

  Данный вариант может быть выбран в разметке режима редактирования и просмотра.

  Если элемент с режимом ***Редактирование*** добавлен в разметку просмотра, необходимо самостоятельно обеспечить сохранение его значения. Например, с использованием скриптов карточек.

- ***Без редактирования***. Значение изменить нельзя.

Сворачиваемый

Настройка возможности скрытия содержимого ЭУ:

- Когда флаг установлен, содержимое может быть скрыто кнопкой сворачивания.
- Когда флаг не установлен, содержимое не может быть скрыто.

Сохранять номер последовательности

Свойство, сохраняющее порядок вывода строк таблицы. Должно быть указано, если требуется иметь возможность изменять прядок строк перетаскиванием. Требуемый тип поля: ***целое***.

Способ редактирования строк

Определяет способ редактирования значения в строках таблицы:

- ***Встроенный***. Редактирование значения осуществляется в строке самой таблицы.
- ***Разметка***. Редактирование значения осуществляется в отдельном окне со своей разметкой. Разметка настраивается отдельно.



Данное свойство доступно только в разметках секций карточек для режима просмотра.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

Перед разворачиванием

Вызывается перед разворачиванием ЭУ кнопкой `v`.

Перед сворачиванием

Вызывается перед сворачиванием *Блока* кнопкой ^.

После добавления строки

Вызывается, когда строка таблицы была добавлена.

После разворачивания

Вызывается после разворачивания таблицы.

После сворачивания

Вызывается после сворачивания таблицы.

После удаления строки

Вызывается, когда строка таблицы была удалена.

При добавлении строки

Вызывается перед добавлением строки таблицы.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При удалении строки

Вызывается перед удалением строки таблицы.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Таблица / Добавить файл

Элемент управления Таблица / Добавить файл вызывает операцию добавления файла в список файлов, отображаемых в элементе управления "Таблица". Операцию предполагается использовать, если работа с файлами в Таблице осуществляется с помощью элемента управления "Файл".

ЭУ следует добавлять в список операций ***TableOperationsNode*** элемента управления Таблица.

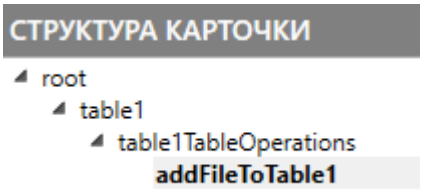


Рисунок 690. Операция "Добавить файл" в списке операций элемента управления "Таблица"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Перетаскивание файла

При включении разрешает добавлять файлы в таблицу перетаскиванием с файловой системы.

Текст при перетаскивании

Определяет текст, отображаемый в ЭУ в блоке приёма перетаскиваемого файла.

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Данные

Операция редактирования

Определяет операцию редактирования, которая должна быть доступна

пользователю для возможности использования данной команды.

Не ограничивает возможность добавления файлов другими средствами. Если операция не выбрана, команда будет доступна.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед добавлением файлов

Вызывается при добавлении файла в таблицу.

После добавления файлов

Вызывается после добавление файла в таблицу.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Таблица / Скачать все

Элемент управления Таблица / Скачать все вызывает операцию скачивания всех файлов, отображаемых в элементе управления "Таблица". Операцию предполагается использовать, если работа с файлами в Таблице осуществляется с помощью элемента управления "Файл".

ЭУ следует добавлять в список операций ***TableOperationsNode*** элемента управления Таблица.

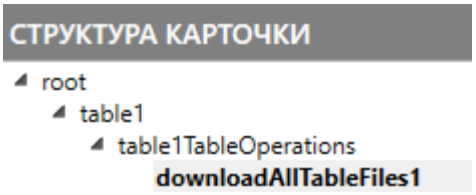


Рисунок 691. Операция "Скачать все" в списке операций элемента управления "Таблица"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Данные

Операция редактирования

Определяет операцию редактирования, которая должна быть доступна пользователю для возможности использования данной команды. Если операция не выбрана, команда будет доступна.

Если значение **Операции редактирования** наследуется от родительского Блока, название свойства меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются,

когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Отключен

Когда флаг установлен, отключается возможность изменить значения элемента управления. Работает совместно со свойством **Операция редактирования**. Редактирование будет запрещено, если одно из свойств запрещает редактирование.

Адаптивный

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед началом скачивания файлов

Вызывается перед скачиванием всех файлов.

Перед скачиванием файла

Вызывается перед скачиванием каждого файла из группы скачиваемых файлов.

После скачивания всех файлов

Вызывается после скачивания всех файлов.

После скачивания файла

Вызывается после скачивания каждого файла из группы скачиваемых файлов.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контрол табличных данных

Элемент управления отображает данные в виде таблицы на основании модели, полученной с сервера. Модель формируется программно при помощи плагина.

Таблицу можно использовать как в карточке, так и в любой разметке, например в разметке дашборд.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Данные

Название плагина

Обязательный параметр, на основании которого происходит поиск плагина для формирования таблицы. Например, плагин `EDIMessagesLog`, входящий в комплект поставки модуля "Модуль интеграции с операторами ЭДО", подробнее см. в разделе "Плагины для грида".



Создание плагина подробнее описано в руководстве разработчика.

Параметры формирования таблицы

Набор дополнительных атрибутов, которые можно указать в Конструкторе Web-разметок. Указанные атрибуты попадут в плагин.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

После загрузки таблицы

Вызывается после загрузки таблицы данных.

При загрузке таблицы

Вызывается при загрузке таблицы данных.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Фильтр папки

Web-клиент Конструктор Web-разметок

Фильтр папки

Описание элементов управления группы "Фильтр папки".

- Кнопка сброса фильтра папки
- Элементы фильтра папки

Кнопка сброса фильтра папки

Элемент управления `Кнопка сброса фильтра папки` предназначен для добавления кнопки, сбрасывающей настройки фильтра папки, в разметку локации *WebFrame / Панель поиска по столбцам представления*.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Элементы фильтра папки

Элемент управления Элементы фильтра папки предназначен для добавления списка настроек фильтра папки в разметку локации *WebFrame / Панель поиска по столбцам представления*.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Элементы разметки

Особая группа элементов управления — *Элементы разметки*.

Элементы данной группы являются контейнерами и не имеют привязки к источнику данных или собственных функций.

Блок

Элемент разметки Блок является контейнером для размещения других *ЭР* и ЭУ, а также других Блоков. *Блок* является основным инструментом реализации адаптивной разметки. За счет наличия предоставления Блок обладает рядом свойств, значение которых зависит от типа устройства пользователя.

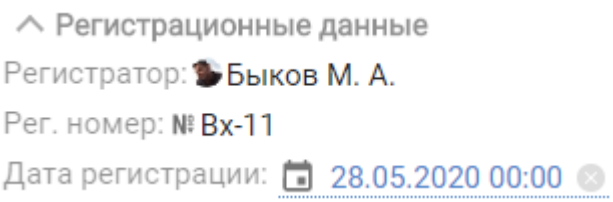


Рисунок 692. Пример элемента разметки "Блок"

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заголовок

Текст, который будет отображаться над содержимым *Блока*. Заголовок выделяется жирным шрифтом.

Локализуемый

Максимальная высота

Если блок начинает превышать максимальную высоту у него появляется полоса прокрутки из нового грида. Высота задаётся в пикселях.

Адаптивный

Отступы

Устанавливает дополнительные внешние отступы в *Блоке*:

- ***true*** — по границе *Блока* будут добавлены дополнительные отступы.
- ***false*** — *Блок* в карточке отображается со стандартными отступами от других ЭУ.

Данные

Операция редактирования

Выбор операции редактирования значения для вложенных элементов управления и Блоков.

- Если операция недоступна пользователю, изменение значений вложенных в Блок элементов управления также недоступно.
- Если операция не выбрана, возможность редактирования значения вложенных элементов управления не проверяется.



Данная настройка применяется только ко вложенным элементам управления, у которых не указано собственное значение настройки *Операция редактирования*, если такая настройка есть.

Если значение **Операции редактирования** Блока наследуется от родительского, название настройки меняется на **Операция редактирования (наследовано)**.

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Выравнивание слева

Выравнивание дочерних элементов внутри Блока:

- Когда флаг установлен, содержимое Блока выравнивается по левой стороне.
- Когда флаг не установлен, содержимое Блока выравнивается по правой стороне.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

В случае ЭУ Блок **Операция редактирования для видимости** также применяется для ЭУ внутри Блока.

Отображать свернутым

Определяет начальное состояние *Блока*:

- Когда флаг установлен, содержимое Блока не отображается в карточке;
- Когда флаг не установлен, содержимое *Блока* отображается в карточке.

Проверяется только при установленном флаге "Сворачиваемый"

Адаптивный

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Сворачиваемый

Настройка возможности скрытия содержимого ЭУ:

- Когда флаг установлен, содержимое может быть скрыто кнопкой сворачивания.
- Когда флаг не установлен, содержимое не может быть скрыто.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (целое число) от ширины родительского элемента управления.

Адаптивный

Ширина метки (проценты)

Устанавливает относительную ширину области для размещения меток всех элементов управления, вложенных в Блок.

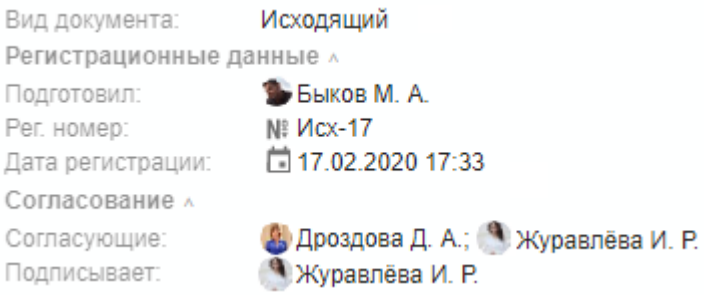


Рисунок 693. Блок с заданной шириной метки

Действие настройки не распространяется на элементы управления

вложенные во вложенный Блок.

Если значение настройки не задано, значение элемента управления отображается непосредственно после метки.

Если значение данного свойства задано, действие свойства **Выравнивание слева** распространяется только на метку, т.е. значение элемента управления будет выравниваться по левой стороне.

Если установленной ширины области недостаточно для размещения текста метки, оно переносится на следующую строку.

Действие настройки не распространяется на следующие элементы управления с меткой: Файл, Файл замечаний, Контейнер строка.

Адаптивный

События

Перед разворачиванием

Вызывается перед разворачиванием ЭУ кнопкой ∨.

Перед сворачиванием

Вызывается перед сворачиванием *Блока* кнопкой ∧.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При разворачивании

Вызывается после разворачивания ЭУ кнопкой ∨.

При сворачивании

Вызывается после сворачивания *Блока* кнопкой ∧.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Вкладки

Элемент разметки Вкладки предназначен для размещения нескольких переключаемых страниц. На каждой из страниц может быть размещён набор отображаемых или изменяемых данных карточки.

Каждая вкладка ЭР Вкладки представляет собой отдельный элемент разметки — "Страница вкладок". В ЭР Вкладки может быть добавлено необходимое количество ЭУ Страница вкладок со своими наборами подчиненных ЭУ. В ЭУ Вкладки могут быть добавлены только ЭУ Страница вкладок.

ЭР Вкладки является контейнером, который непосредственно не отображается в карточке.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Индекс вкладки по умолчанию

Порядковый номер вкладки (начиная с `0` — первая вкладка), которая будет открыта по умолчанию при открытии карточки и обновлении страницы.

Минимальная ширина

Минимально возможная ширина элемента управления в пикселях.

Адаптивный

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Основной таб в мобильной верстке

При работе на мобильном устройстве, если в разметке используется несколько ЭР Вкладки, один из ЭР Вкладки может быть назначен основным. Страницы основного ЭР Вкладки отображаются под панелью управления. Страницы прочих ЭР Вкладки — на месте, определённом разметкой данной карточки.

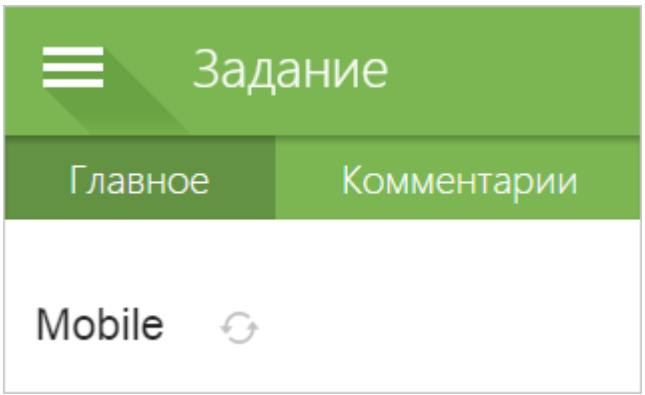


Рисунок 694. Элемент разметки "Вкладки" в карточке, открытой на мобильном устройстве

Данное свойство определяет, какой из нескольких ЭР Вкладки является основным:

- Когда флаг установлен, текущий ЭР Вкладки является основным.
- Когда флаг не установлен, текущий ЭР Вкладки будет размещен в карточке стандартным образом (согласно настройкам разметки). В одной разметке может быть только один основной ЭР Вкладки. Данная настройка не влияет на разметку карточки при работе на персональном компьютере.

Порядок

Определяет порядок отображения элемента управления в родительском Блоке. ЭУ с более низким порядком имеет более высокий приоритет в порядке отображения в разметке.

Адаптивный

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

Ширина в процентах

Ширина элемента управления в процентах (указывается целое число) от ширины родительского элемента управления.

Адаптивный

События

После загрузки всех ЭУ

Вызывается после загрузки всех элементов разметки. В зависимости от режима загрузки данных, на момент вызова события:

- Данные элементов управления будут загружены. Применимо для ЭУ с синхронной загрузкой данных и ЭУ с загрузкой данных из внешнего источника, включая источники с асинхронным режимом загрузки.
- Данные могут быть загружены не полностью. Применимо для ЭУ с асинхронным режимом загрузки данных, например, Задания, Ссылки и т.п.



Если в разметке размещен элемент управления Вкладки, то событие вызывается только после загрузки элементов первой активной вкладки.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При переключении активной вкладки

Вызывается при переключении на другую вкладку.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Кнопки сохранения

Элемент разметки Кнопки сохранения включает две кнопки **Сохранение** и **Отмена**, предназначенные для вызова стандартных функций сохранения или отмены изменений карточки.



Данный ЭР следует обязательно добавлять в разметку режима редактирования (и создания) карточки. В противном случае сохранение карточки стандартным образом будет невозможно. Если не реализованы функции сохранения с помощью других ЭУ.



Рисунок 695. Пример отображения элемента управления "Кнопки сохранения"



Данный ЭУ может быть добавлен только в разметку, предназначенную для редактирования карточек.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Текст кнопки ОК

Текст, отображаемый в кнопке сохранения изменений карточки.

Локализуемый

Текст кнопки Отмена

Текст, отображаемый в кнопке отмены изменений карточки.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг Видимость установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг Видимость НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

Перед щелчком

Вызывается перед обработкой щелчка мыши по любой области элемента управления.

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Контейнер строка

Элемент разметки Контейнер строка является контейнером для других элементов управления. Дочерние элементы управления в ЭР Контейнер строка размещаются последовательно слева направо.

Особенности элемента управления "Контейнер строка"

- Содержимое элемента управления при необходимости переносится на следующую строку (без ограничений по высоте).
- Вложенные элементы размещаются последовательно без пропусков с настраиваемым разделителем.
- Вложенные элементы управления должны быть в режиме редактирования ***Без редактирования***.
- Поддерживаются следующие типы дочерних элементов управления: Строка, Метка, Дата/время, Флаг, Раскрывающийся список, Число, Партнер, Подразделение, Подразделение контрагента, Строка конструктора справочников.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Разделитель

Символ или строка, используемая для разделения дочерних элементов при отображении.

Текст метки

Содержимое метки, отображаемой слева от элемента управления.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства Видимость:

- Когда флаг Видимость установлен и выбрана *операция редактирования для*

видимости, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.

- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Страница вкладок

Элемент разметки Страница вкладок представляет собой одну вкладку в ЭР "Вкладки", на которой размещаются требуемые элементы управления.

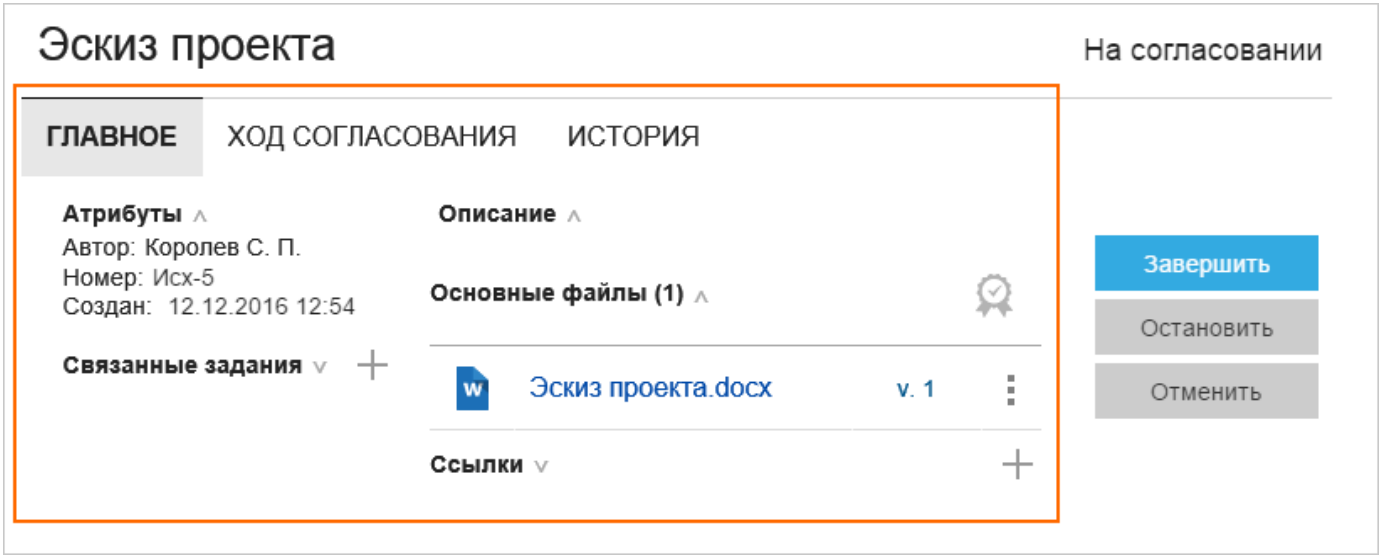


Рисунок 696. Пример элемента разметки "Страница вкладок"



ЭР Страница вкладок может быть встроен только в ЭР **Вкладки**.

Свойства

Общие

Название ЭУ

Название типа элемента управления.

Название

Уникальное название элемента управления.

Тег

Простое текстовое свойство, позволяющее добавить для ЭУ скрипт (например, JSON).

Внешний вид

Заголовок

Текст, отображаемый в названии вкладки. Максимальное количество символов — 30.

Локализуемый

Подсказка

Содержимое всплывающей подсказки.

Локализуемый

Поведение

Видимость

Настройка видимости. Элемент управления отображается в карточке, когда флаг установлен. Элемент управления и любое его содержимое не отображаются, когда флаг снят.

Адаптивный

Дополнительные css классы

Названия дополнительных классов CSS для изменения стиля элемента управления. Перечисляются через пробел.

Переходить по TAB

Флаг определяет последовательность перехода по ЭУ карточки при нажатии кнопки `TAB`. Если флаг установлен, переход по `TAB` разрешён.

Операция редактирования для видимости

Определяет операцию, которая должна быть доступна, чтобы ЭУ отображался для пользователя. Работа свойства зависит от значения свойства **Видимость**:

- Когда флаг **Видимость** установлен и выбрана *операция редактирования для видимости*, видимость элемента определяется исходя из доступности пользователю выбранной операции редактирования.
- Когда флаг **Видимость** установлен, и *операция редактирования для видимости* НЕ выбрана, ЭУ отображается всегда.
- Когда флаг **Видимость** НЕ установлен, ЭУ всегда скрыт.

Стандартный css класс

Название CSS класса, в котором определён стандартный стиль элемента управления.

События

При наведении курсора

Вызывается при входе курсора мыши в область элемента управления.

При отведении курсора

Вызывается, когда курсор мыши покидает область элемента управления.

При получении фокуса

Вызывается, когда элемент управления выбирается.

При потере фокуса

Вызывается, когда выбор переходит к другому элементу управления.

При щелчке

Вызывается при щелчке мыши по любой области элемента управления.

Приложение А. Правила выбора разметки для открытия карточки

При открытии карточки в Web-клиенте используемая разметка выбирается по следующим правилам:

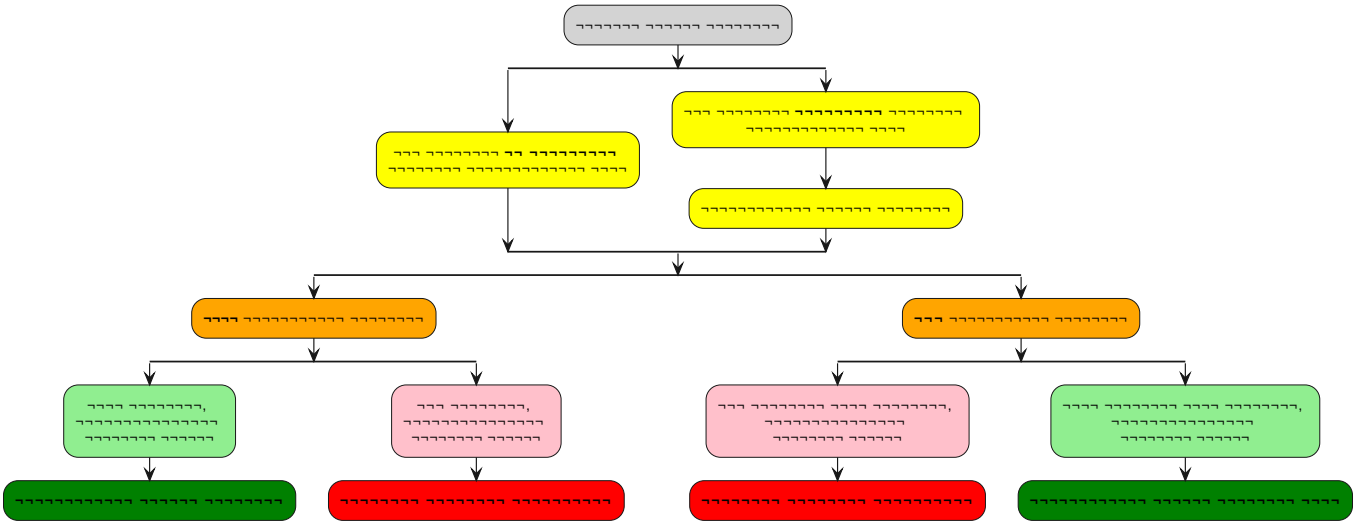


Рисунок 697. Правила выбора разметок при открытии карточки

[0] Операции над всеми выбранными карточками.

[1] Правила выбора календаря приведены в описании свойства Время по умолчанию

Руководство по разработке решений

Руководство по разработке

Данный раздел посвящён разработке программных компонентов для решений на базе Web-клиента.

В разделе приводится информация по разработке:

- Клиентских скриптов,
- Серверных и клиентских расширений,
- Расширений программы Конструктор Web-разметок,
- Элементов управления,
- Другую информацию, связанную с разработкой Решений,
- Дополнительных компонентов Web-клиента,
- Комментарии к исходному коду примеров дополнительных компонентов,
- Описания членов объектной модели Web-клиента, задействованных при разработке компонентов,
- Справочные сведения, касающиеся разработки на базе Web-клиента.

Руководство предназначено для специалистов, обладающих знаниями в области веб-разработки, знакомых с разработкой компонентов на ReactJS, обладающих опытом разработки на платформе Docsvision 6.



Обратите внимание. При обращении к API, методы следует вызывать последовательно, один за другим. Параллельный вызов методов API не гарантирует их работу.

Перевод решений на .NET 8.0

Начиная с версии 6.1 сборки 170 и выше, Web-клиент работает на .NET 8.0, поэтому при переходе с Web-клиент, использующего .NET предыдущих версий, необходимо перевести на .NET 8.0 собственные решения.

Изменение файла проекта

Конвертация существующего проекта может быть выполнена:

- При помощи автоматических инструментов (см. документацию Microsoft).
- При помощи создания нового проекта, нацеленного на .NET 8. При создании

проекта предпочтительно использовать шаблоны VisualStudio из примеров разработки расширений для версии 6.1.

В проектах нового типа нет необходимости добавлять файлы с исходным кодом в `.csproj`-файл проекта. По умолчанию автоматически все файлы в папке проекта считаются частью проекта. Это существенно упрощает процесс преобразования проекта.

Важной особенностью нового проекта является атрибут `Sdk` на корневом элементе `Project` файла `.csproj`, это xml-файл, который можно открыть в блокноте. Для web-расширений рекомендуется значение `Microsoft.NET.Sdk.Web`, для расширений конструктора разметок `Microsoft.NET.Sdk`.

Значение тега `TargetFramework` для web-расширений должно быть `net8.0`, для расширений конструктора разметок — `net8.0-windows`.

Для проектов расширений конструктора разметок также могут быть полезны следующие элементы:

```
<UseWindowsForms>true</UseWindowsForms>
<UseWPF>true</UseWPF>
<ImportWindowsDesktopTargets>true</ImportWindowsDesktopTargets>
```

Особенности сборки

Для сборки C# кода для .NET 8 требуется VisualStudio 2022 или старше.

При использовании сторонних зависимостей требуется обновить их до версий, поддерживающих .NET 8. В случае нарушений совместимости при сборке будет выдаваться предупреждение `CA1416`. Данные предупреждения следует все устранять.

При разработке расширений для программы Конструктор Web-разметок и другого кода, рассчитанного на работу в ОС Windows следует использовать атрибут `SupportedOSPlatform`. Его можно назначить на отдельный класс, либо на всю сборку:

```
[assembly: SupportedOSPlatform("windows")]
```

Без этого атрибута также может возникать предупреждение `CA1416`, при обращении к API, рассчитанному на работу в ОС Windows.

Логика привязки параметров в запросах

- Метод контроллера либо принимает несколько параметров простых типов через query-параметры, либо один параметр, значение которого читается из тела запроса.
- Рекомендуется явно указывать, откуда читается параметр с помощью атрибутов `[FromBody]` и `[FromQuery]`. Если ранее использовался атрибут `FromUri`, то нужно заменить на `FromQuery`.
- Рекомендуется явно помечать методы атрибутами `[HttpGet]`, `[HttpPost]`, для указания метода вызова

Например, метод может быть объявлен так:

```
public class MathController : ControllerBase
{
    [HttpGet]
    public double GetSquare([FromQuery] double value) {
        return
    }
}
```

Обращение к методу будет, соответственно, выглядеть так:

```
let result = await requestManager.get(`/Math/GetSquare?value=2`, null);
```

Метод может быть объявлен так:

```
class MinValueParams {
    public List<double> Values { get; set; };
}

public class MathController : ControllerBase
{
    [HttpPost]
    public double MinValue([FromBody] MinValueParams  arg) {
        return arg.Values.Min();
    }
}
```



Если не указать `[FromBody]` для сложного параметра, его значение не привяжется.

Обращение к такому методу будет выглядеть следующим образом:

```
let arg = { values: [ 1, 5, 3, 2] };
let result = await requestManager.post(`/Math/MinValue`, JSON.stringify(arg));
```



При передаче параметра в теле запроса нельзя указывать имя параметра, т.к. он может быть только один.

Следующий код не будет работать:

```
let data = { arg: { values: [ 1, 5, 3, 2] } }; ①
let result = await requestManager.post(`/Math/MinValue`, JSON.stringify(data)); ②
```

① Неверный код.

② Код не будет работать.

Внедрение зависимостей

В Web-клиенте 18 (6.1) вместо Autofac используется стандартный механизм внедрения зависимостей .NET. В связи с этим в классе серверного расширения нужно произвести следующие изменения:

Вместо метода `InitializeContainer` *определить метод:*

```
public override void InitializeServiceCollection(IServiceCollection services)
{
①
}
```

① Регистрация сервисов.

Регистрация сервисов изменяется следующим образом:

```
  containerBuilder.RegisterType<YourServiceClass>().As<YourServiceInterface
>().SingleInstance();
  ->
  services.AddSingleton<YourServiceInterface, YourServiceClass>();

  containerBuilder.RegisterOrderedType<YourBindingConverterType, IBindingConverter>();
  ->
  services.AddSingleton<IBindingConverter, YourBindingConverterType>();

  containerBuilder.RegisterOrderedType<YourBindingResolverType, IBindingResolver>();
  ->
  services.AddSingleton<IBindingResolver, YourBindingResolverType>();
```

```
  containerBuilder.RegisterOrderedType<YourControlResolverType, IControlResolver>();
  ->
  services.AddSingleton<IControlResolver, YourControlResolverType>();

  containerBuilder.RegisterOrderedType<YourPropertyResolverType, IPropertyResolver>();
  ->
  services.AddSingleton<IPropertyResolver, YourPropertyResolverType>();

  containerBuilder.RegisterType<YourCardLifeCycle>().Keyed<ICardLifeCycle>(CardTypeID
).SingleInstance();
  ->
  services.AddTransient<ICardLifeCycle, YourCardLifeCycle>();

  containerBuilder.RegisterType<YourRowLifeCycle>().Keyed<IRowLifeCycle>(SectionID
).SingleInstance();
  ->
  services.AddTransient<IRowLifeCycle, YourRowLifeCycle>();
```

Прочие изменения

- В конструкторе класса расширения при вызове базовой реализации теперь не нужно передавать `serviceProvider`:

  Было

  ```
  public ServerExtension(IServiceProvider serviceProvider)
    : base(serviceProvider)
  {
  }
  ```

  Стало

  ```
  public ServerExtension(IServiceProvider serviceProvider)
    : base()
  {
  }
  ```

- Классы контроллеров рекомендуется наследовать от `Microsoft.AspNetCore.Mvc.ControllerBase`. При переводе методы `requestManager get` и `post` в некоторых случаях могут изменить возвращаемое значение. В частности, ранее они могли возвращать строку, а после перевода — объект — результат `JSON.parse` этой строки. С примером можно ознакомиться ниже.

Было

```
const response = JSON.parse(await requestManager.get(url));
```

Стало

```
const response = await requestManager.get(url)
```

- Также в отдельных случаях может измениться регистр первой буквы в названии свойств ответа сервера. Например, следующим образом:

  Было

  ```
  const name = response.IssueName;
  ```

  Стало

  ```
  const name = response.issueName;
  ```

- Изменился способ загрузки и скачивания файлов. Пример реализации соответствующего метода на .NET6 можно найти в примере `ExternalWebService`, в файле `DocumentsController.cs`, методах `UploadFile` и `DownloadFile`.
- Чтение настроек из конфигурационного файла Web-сервиса изменилось. Прежние сервисы оставлены для обратной совместимости, однако рекомендуется использовать стандартный механизм работы с настройками .NET 8. В частности, использовать внедрение интерфейса `IOptions` через конструктор для получения настройки и `AddOptions` + `BindConfiguration` при регистрации сервисов для регистрации настройки. Пример работы с настройками можно найти в примере `KonturIntegration` (регистрация см. `KonturServerExtension.cs/InitializeServiceCollection`, чтение см. `KonturRequestService.cs`).
- Библиотека `iTextSharp`, которая использовалась в примере `Watermark`, поддерживает .NET6 только в новой версии, под названием `itext7`. В этой новой версии полностью изменился API.

С чего начать

Исходные коды примеров серверных и клиентских расширений, расширений программы Конструктор Web-разметок, а также элементов управления содержатся в репозитории WebClient-Samples на ресурсе GitHub.

Загрузите к себе репозиторий, чтобы начать работать с примерами.

Подготовка

Для сборки примеров в Visual Studio[0] (серверные расширения и расширения программы Конструктор Web-разметок) необходимы актуальные версии компонентов Web-клиента и системы Docsvision, которые можно скопировать в папку Assemblies из каталога установки Web-клиента и системы Docsvision.

Для проверки зависимостей и быстрой сборки сразу всех примеров можно воспользоваться установочным пакетом Docsvision WebClient SDK. При установке файлы примеров будут собраны и помещены в соответствующие каталоги. При необходимости (например, если Web-клиент уже установлен) вы можете изменить путь сборки через переменную окружения в node и проектах Visual Studio[1]. Если в примере требуется загрузка разметки или библиотеки карточек, это необходимо произвести вручную.

После установки SDK в каталоге `Docsvision\WebClient\SDK\USAGE.md` указаны инструкции по разворачиванию примеров.

1. Подготовьте среду разработки.

   На компьютере разработчика должны быть установлены:

   Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.
- TypeScript 3.8.



Если указанные компоненты не установлены, при установке SDK будет предложено их автоматически загрузить и установить.

1. Загрузите на компьютер репозиторий WebClient-Samples.
2. Установите шаблоны проектов из полученного репозитория:

- *TemplateDesignerExtension* — шаблон проекта для разработки расширения программы Конструктор Web-разметок.
- *TemplateServerExtension* — шаблон проекта для разработки серверного расширения.

Чтобы установить шаблоны:

1. Скопируйте папки `Docsvision Web-layout designer extension template` и `Docsvision Web-client server extension template` из каталога репозитория `ProjectTemplates\VSTemplates\` в каталог шаблонов проектов Visual Studio[2] (`C:\Users\\%username%\Documents\Visual Studio %version%\Templates\ProjectTemplates\`).
2. Запустите Visual Studio[3]. Будут добавлены новые типы проектов: *TemplateDesignerExtension* и *TemplateServerExtension*.

Что дальше

1. Определите цель разработки.

   Web-клиент имеет несколько точек расширения функциональных возможностей.

   Программист может разработать:

   - Серверное расширение, добавляющее методы WebAPI, для вызова с клиентского уровня.

     Серверные методы могут быть использованы для взаимодействия с сервером Docsvision: создание карточек, запуск БП и т. д.

   - Новый тип элементов управления для создания разметок.

     Данный тип расширения включает два компонента: *расширение программы Конструктор Web-разметок* (обеспечивает возможность настройки разметки с использованием ЭУ) и *клиентское расширение* (обеспечивает функционирование ЭУ на клиентской стороне).

   - Клиентский скрипт, содержащий обработчики событий разметки.

     Данные скрипты могут добавлять в Web-клиент необходимую бизнес-логику или обеспечивать другие функции клиентского уровня.

   - Клиентское расширение — общее название компонентов, обеспечивающих функциональность клиентского уровня.
2. Ознакомьтесь с инструкцией по разработке требуемого типа компонента Web-клиента в данном руководстве.
3. Ознакомьтесь с предоставляемыми примерами.

Примеры

Папка репозитория `WebClient-Samples` *содержит каталоги:*

- `Assemblies` — папка с подключаемыми сборками Web-клиента, платформы Docsvision и др..
- `ClientScripts` — папка с исходными кодами клиентских расширений, предоставляющих обработчики событий. Исходные коды приведены на TypeScript (подкаталог TypeScript).
- `ControlProperties` — папка с исходным кодом расширения программы Конструктор Web-разметок, добавляющего новый тип свойств элемента управления.
- `Controls` — папка с исходными кодами элементов управления.
- `Others` — папка с прочими примерами.
- `Obsolete` — устаревшие примеры, работа которых не поддерживается в текущей версии Web-клиента.
- `ProjectTemplates` — папка с шаблонами проектов серверного расширения и расширения программы Конструктор Web-разметок.
- `ServerExtensions` — папка с исходными кодами серверных расширений.
- `Install` и другие папки — служебные директории.

Примеры сопровождаются файлами `README.md`, в которых приведено краткое описание примера и инструкция по его сборке и проверке.

Часть исходных кодов представляет собой проекты Visual Studio[4] (решение `Samples`), а часть — отдельные файлы клиентских расширений (TypeScript, CSS).

Список примеров

Примеры клиентских расширений

Все клиентские скрипты также включают в себя проекта расширения Web-клиента.

Примеры клиентских расширений с обработчиками событий (скриптов):

- `ClientScripts\TypeScript\ChangeLogoAndBackground` — демонстрирует пример реализации скрипта, изменяющего заголовок страницы и её фоновое изображение для папок и дашборда.
- `ClientScripts\TypeScript\ControlsRelation` — пример демонстрирует способ

организации взаимодействия элементов разметки, через обработчики событий (в примере видимость одного элемента управления зависит от значения другого).

- `ClientScripts\TypeScript\CopyValueToControl` — пример метода, в котором выполняется копирование значения одного элемента управления в другой.
- `ClientScripts\TypeScript\ExtendServerQueryData` — индикатор высокого приоритета для просроченных заданий.
- `ClientScripts\TypeScript\FileListHighlightRow` — добавляет красную обводку при скачивании файла.
- `ClientScripts\TypeScript\GridColumnBackgroundRows` — новая колонка таблицы грида — *Фон строки*.
- `ClientScripts\TypeScript\GridContextMenuItem` — новый пункт контекстного меню Открыть карточку в модальном окне.
- `ClientScripts\TypeScript\GridHoverPanelGrouping` — группировка по столбцу.
- `ClientScripts\TypeScript\GridToolbarButton` — кнопка в гриде отображает общее количество карточек, а также количество непрочитанных.
- `ClientScripts\TypeScript\LoginPage` — настройка страницы авторизации.
- `ClientScripts\TypeScript\ShowRequiredFields` — отображение незаполненных обязательных полей.
- `ClientScripts\TypeScript\ValidateOnStateChanging` — пример реализации обработчика изменения состояния карточки, осуществляющего проверку значения текстового поля, с отменой изменения состояния, если поле не заполнено или количество знаков в строке больше `100`.
- `ClientScripts\AdvancedTypeScript` — примеры клиентских расширений, в которых задействован React API:
  - `ClientScripts\AdvancedTypeScript\AdaptiveMenuBar` — пример реализации панели меню с собственным набором элементов.
  - `ClientScripts\AdvancedTypeScript\MessageBox` — пример вызова стандартных диалоговых окон Web-клиента.
  - `ClientScripts\AdvancedTypeScript\ModalDialog` — пример реализации диалогового окна с собственным содержимым.
  - `ClientScripts\AdvancedTypeScript\ModalSidebar` — пример вызова стандартной боковой панели Web-клиента.

Примеры серверных расширений

- `ServerExtensions\CSPSignatureVisualization` — пример изменения визуализации штампа электронной подписи: вставки собственных изображений штампов электронной подписи в файл, приложенный к карточке Docsvision.
- `ServerExtensions\CreateCard` — пример реализации функции создания исходящих документов, представленной в виде клиентской и серверной частей:
  - `ServerExtensions\CreateCard\CreateCardServerExtension` — серверное расширение, добавляющее метод WebAPI, создающий карточку исходящего документа.
  - `ServerExtensions\CreateCard\CreateCardWebExtension` — клиентское расширение, добавляющее клиентский сервис создания документа (вызывает метод расширения `CreateCardServerExtension`) и пример обработчика событий, использующего клиентский сервис.
- `ServerExtensions\CreateCardDialog` — серверное расширение, добавляющее в Web-клиент возможность создания карточек типа *Этап согласования*:
  - `ServerExtensions\CreateCardDialog\CreateCardDialogServerExtension` — серверное расширение, добавляющее фабрику карточек, позволяющую создавать карточки Этап согласования.
  - `ServerExtensions\CreateCardDialog\CreateCardDialogWebExtension` — клиентское расширение, регистрирующее новый тип карточек на стороне клиента.
- `ServerExtensions\CustomConditionTypes` — отображение разметки в соответствии с заданными условиями.
  - `ServerExtensions\CustomConditionTypes\CustomConditionTypesDesignerExtension` — проект-расширение для программы Конструктор Web-разметок. Содержит класс `CustomConditionTypesProvider`, в котором реализованы методы для отображения новых условий выбора разметки по группе, пользователю, роли
  - `ServerExtensions\CustomConditionTypes\CustomConditionTypesLayoutExtension` — проект-расширение для Web-клиента, которое позволяет определить — выполняется условие выбора разметки или нет для текущего контекста.
  - `ServerExtensions\CustomConditionTypes\CustomConditionTypesInterfaces` — проект-расширение для программы Конструктор Web-разметок и серверного расширения, содержащий все константные значения для этих проектов.
- `ServerExtensions\CustomLibrary` — пример демонстрирует способ добавление нового типа карточек. Пример включает несколько компонентов:

    - ServerExtensions\CustomLibrary\CardDefs — схема новой библиотеки карточек с карточкой справочника.
    - ServerExtensions\CustomLibrary\CustomLibrary.ObjectModel — классы объектной модели библиотеки карточек.
    - ServerExtensions\CustomLibrary\CustomLibraryServerExtension — серверное расширение, добавляющее в Web-клиент поддержку нового типа карточек и WebApi функцию получения данных карточек нового типа.
    - ServerExtensions\CustomLibrary\CustomLibraryWebExtension — клиентское расширение, добавляющее клиентский сервис для вызова функции серверного расширения CustomLibraryServerExtension и пример обработчика событий, использующего клиентский сервис.
- ServerExtensions\DataGridControlExtension — плагин загружает информацию о файлах для документа.
    - ServerExtensions\DataGridControlExtension\DataGridControlServerExtension — проект-расширение для Web-клиента. Содержит плагин Files (FilesDataGridControlPlugin), реализующий загрузку информации о файлах документа.
- ServerExtensions\ExcelExport — экспорт в Excel.
    - ServerExtensions\ExcelExport\ExcelExportServerExtension — проект-расширение для Web-клиента, позволяющего вмешать в процесс экспорта в Excel.
- ServerExtensions\ExtendedCardInfo — пример демонстрирует возможность получения данных карточки, которые по умолчанию не загружаются и не передаются клиенту. Пример представлен в виде клиентской и серверной частей:
    - ServerExtensions\ExtendedCardInfo\ExtendedCardInfoServerExtension — серверное расширение, добавляющее WebApi функцию загрузки из Docsvision данных карточки.
    - ServerExtensions\ExtendedCardInfo\ExtendedCardInfoWebExtension — клиентское расширение, добавляющее клиентский сервис для вызова функции серверного расширения ExtendedCardInfoServerExtension и пример обработчика событий, использующего клиентский сервис.
- ServerExtensions\LicenseCheck — демонстрация способа проверки лицензии Docsvision на наличие дополнительной опции. Пример представлен в виде клиентской и серверной частей:

    - ServerExtensions\LicenseCheck\LicenseCheckServerExtension — серверное расширение, реализующее функцию проверки лицензии.
    - ServerExtensions\LicenseCheck\LicenseCheckWebExtension — клиентское расширение, добавляющее клиентский сервис для вызова функции проверки лицензии из серверного расширения LicenseCheckServerExtension и пример обработчика событий, использующего клиентский сервис.
- ServerExtensions\ShiftTasksEndDate — пример реализации функции изменения данных связанных карточек. Пример представлен в виде клиентской и серверной частей:
    - ServerExtensions\ShiftTasksEndDate\ShiftTasksEndDateServerExtension — серверное расширение, в котором реализована функция изменения времени исполнения в заданиях, связанных с документом.
    - ServerExtensions\ShiftTasksEndDate\ShiftTasksEndDateWebExtension — клиентское расширение, добавляющее клиентский сервис для вызова функции изменения связанных карточек и пример обработчика событий, использующего клиентский сервис.
- ServerExtensions\TableControl — пример демонстрирует способ получения данных из справочника контрагентов и отображения их в таблице. Пример представлен в виде клиентской и серверной частей:
    - ServerExtensions\TableControl\TableControlServerExtension — серверное расширение с функцией WebApi, предоставляющей данные контрагентов;
    - ServerExtensions\TableControl\TableControlWebExtension — клиентское расширение, реализующее несколько функций: добавляет клиентский сервис для вызова функции серверного расширения TableControlServerExtension. Предоставляет обработчик открытия карточки, использующий клиентский сервис для получения данных контрагентов. Обеспечивает загрузку данных контрагентов в таблицу.

Примеры элементов управления

- Controls\AcquaintancePanel — элемент управления Панель отправки на ознакомление, предназначен для запуска БП отправки документа на ознакомление. Пример включает несколько компонентов:
    - AcquaintancePanel\AcquaintancePanelDesignerExtension — текстовый описатель элемента управления, а также расширение программы Конструктор Web-разметок с новым типом свойств и локализованными ресурсами.

    - AcquaintancePanel\AcquaintancePanelServerExtension — серверное расширение, предоставляющее функцию запуска бизнес-процесса отправки документа на ознакомление.
    - AcquaintancePanel\AcquaintancePanelWebExtension — клиентское расширение с клиентским компонентом элемента управления и сервисом вызова функции запуска БП из расширения AcquaintancePanelServerExtension.
- Controls\CheckBox — элемент управления "Флаг", предназначен для установки и отображения значения булева типа. Пример включает два компонента:
    - CheckBox\CheckBoxDesignerExtension — расширение программы Конструктор Web-разметок с бинарным описателем элемента управления, новым свойством DefaultValue и редактором BooleanMetadataEditor значения свойства DataField, который ограничивает список доступных для выбора полей карточки.
    - CheckBox\CheckBoxWebExtension — клиентское расширение с клиентским компонентом элемента управления.
- Controls\DocumentSignBatchOperation — пример реализации элемента управления Групповая операция / Подписание.
    - Controls\DocumentSignBatchOperation\DocumentSignBatchOperationWebExtension — проект-расширение серверной части Web-клиента. Содержит ЭУ групповой операции Подписание документа.
    - Controls\DocumentSignBatchOperation\DocumentSignBatchOperationDesignerExtension — содержит решение для программы Конструктор Web-разметок с локализованными ресурсами.
- Controls\DownloadFilesGroupOperation — пример реализации элемента управления группой операции — загрузки файлов выбранных карточек Документ на компьютер. Пример включает несколько компонентов:
    - DownloadFilesGroupOperation\DownloadFilesGroupOperationDesignerExtension — расширение программы Конструктор Web-разметок с бинарным описателем элемента управления, новыми свойствами downloadDocumentFileMode и BatchOperationRestrictionFoldersPropertyDescription, и источником данных DownloadDocumentFileModeSource для свойства downloadDocumentFileMode.
    - DownloadFilesGroupOperation\DownloadFilesGroupOperationServerExtension — серверное расширение с функцией, возвращающей файлы карточки.
    - DownloadFilesGroupOperation\DownloadFilesGroupOperationWebExtension — клиентское расширение с клиентским компонентом элемента управления и

сервисом вызова функции получения файлов карточки из расширения `DownloadFilesGroupOperationServerExtension`.

- `Controls\ExchangeRates` — элемент управления `ExchangeRates`, предназначен для отображения курса валют, получаемого с внешнего ресурса. Пример включает два компонента:
  - `ExchangeRates\ExchangeRatesDesignerExtension` — текстовый описатель элемента управления, а также расширение программы Конструктор Web-разметок с локализованными ресурсами.
  - `ExchangeRates\ExchangeRatesWebExtension` — клиентское расширение с клиентским компонентом элемента управления.
- `Controls\HyperComments` — элемент управления `HyperComments`, предназначен для отображения виджета комментариев внешнего веб-приложения `HyperComments`. Пример включает два компонента:

  Обратите внимание, пример устарел. Для его использования требуется лицензия.

  - `HyperComments\HyperCommentsDesignerExtension` — расширение программы Конструктор Web-разметок с бинарным описателем элемента управления, новым типом свойств и локализованными ресурсами.
  - `HyperComments\HyperCommentsWebExtension` — клиентское расширение с клиентским компонентом элемента управления.
- `Controls\Image` — элемент управления `Image`, предназначен для отображения галереи изображений. В примере демонстрируется возможность работы со значением элемента управления сложного типа и локализацией. Пример включает несколько компонентов:
  - `Image\ImageDesignerExtension` — расширение программы Конструктор Web-разметок с бинарным описателем элемента управления, новыми типами свойств и новым редактором `SliderEditor`.
  - `Image\ImageServerExtension` — серверное расширение с конвертером значения `SliderConverter`, который применяется при загрузке значения элемента управления в клиент.
  - `Image\ImageWebExtension` — клиентское расширение с клиентским компонентом элемента управления.
- `Controls\Link` — элемент управления `Link`, предназначен для отображения

ссылки. Пример включает два компонента:

  - `Link\LinkDesignerExtension` — текстовый описатель элемента управления, а также расширение программы Конструктор Web-разметок с локализованными ресурсами.
  - `Link\LinkWebExtension` — клиентское расширение с клиентским компонентом элемента управления.

- `Controls\RefCases` — элемент управления `RefCases`, предназначен для выбора и отображения дела из Справочника номенклатуры дел.

  Обратите внимание, пример устарел. Модуль *Делопроизводство 5* устарел и не поддерживается. Функциональность

  - `RefCases\RefCasesDesignerExtension` — текстовым описателем элемента управления, а также расширение программы Конструктор Web-разметок с дополнительными редакторами;
  - `RefCases\RefCasesServerExtension` — серверное расширение с конвертером значения RefCasesConverter, который применяется при загрузке значения элемента управления в клиент, и сервисами для получения данных из *Справочника номенклатуры дел*.
  - `RefCases\RefCasesWebExtension` — клиентское расширение с реализацией элемента управления.

- `Controls\TextBox` — элемент управления `TextBox`, предназначен для текстового поля с меткой. Пример включает два компонента:

  - `TextBox\TextBoxDesignerExtension` — расширение программы Конструктор Web-разметок с бинарным описателем элемента управления, новым типом свойств и локализованными ресурсами.
  - `TextBox\TextBoxWebExtension` — клиентское расширение с клиентским компонентом элемента управления.

- `ControlProperties\Url` — расширение программы Конструктор Web-разметок с описателем нового свойства **`Url`**.

Прочие примеры

- `Others\CustomStageService` — демонстрационный вариант специального сервиса логики этапа согласования.
- `Others\ExternalWebService` — пример отдельного веб-сервиса, предоставляющего

методы для взаимодействия с платформой Docsvision через WebApi:

- Получение информации о карточке документа по её идентификатору.
- Создание карточки документа по переданной модели.
- Обновление данных карточки документа.
- Удаление карточки документа по её идентификатору.
- Изменение состояния карточки документа.
- Прикрепление файла к карточке документа.
- Получение файла по его идентификатору.
- Получение результата выполнения расширенного отчёта.

- `Others\KonturIntegration` — пример реализации сервиса интеграции с системой Контур.Фокус, который открывает возможность:

  a. При создании Контрагента заполнить реквизиты, полученные из Контур.Фокус на основе указанного ИНН.

  b. В карточке Документа получить из Контур.Фокус отчёт о Контрагенте и приложить его в карточку Документа.

  Содержит следующие компоненты:

  - `KonturServerExtension` — папка с серверным расширением Web-клиента, в котором реализовано проксирование запросов к Контур.Фокус (для обхода ограничений CORS в браузере).
  - `KonturWebExtension` — папка с клиентским расширением, в котором реализовано открытие отчета о контрагенте и загрузка информации.

- `Others\SignalForUsers` — пример использования сервиса отправки оповещений пользователям Web-клиента:
  - `Others\SignalForUsers\SignalServerExtension` — серверное расширение, предоставляющее функцию рассылки оповещений.
  - `Others\SignalForUsers\SignalWebExtension` — клиентское расширение с клиентским сервисом и обработчиком, вызывающим функцию рассылки оповещений.

- `Others\Watermark` — пример расширения программы _DVWebTool_s, добавляющего водяной знак в PDF файлы документа:
  - `Others\Watermark\WatermarkServerExtension` — серверное расширение, предоставляющее методы для получения и добавления файлов в документ.

- Others\Watermark\WatermarkWebExtension — клиентское расширение, вызывающее функцию добавления водяного знака.
- Others\Watermark\WatermarkWebToolExtension — расширение программы _DVWebTool_s, реализующее функцию добавления водяного знака в PDF-файлы на компьютере пользователя.

Шаблоны проектов серверных расширений

Папка ProjectTemplates содержит шаблоны проектов для создания серверных расширений Web-клиента, программы Конструктор Web-разметок, а также клиентских расширений.

- ProjectTemplates\TemplateDesignerExtension — шаблон проекта расширения программы Конструктор Web-разметок.
- ProjectTemplates\TemplateServerExtension — шаблон проекта серверного расширения Web-клиента.
- ProjectTemplates\TemplateWebExtension — шаблон проекта web-расширения.
- ProjectTemplates\TemplateXmlDesignerExtension — XML-шаблон нового ЭУ.

Репозиторий "WebClient-Samples"

Репозиторий "WebClient-Samples" — репозиторий компании "ДоксВижн" на ресурсе GitHub.

Адрес репозитория: https://github.com/Docsvision/WebClient-Samples/.

Репозиторий "WebClient-Samples" содержит исходные коды примеров и шаблоны проектов для Visual Studio[5] для текущей и предыдущих версий Web-клиента.

Как работать со справочником по JS API

Справочник по JavaScript API модуля Web-клиент — ресурс, содержащий описание классов, функций и других членов предоставляемого Web-клиентом клиентского API, используемого при разработке клиентских расширений и скриптов.

Справочник по JS API размещен по адресу: https://docsvision.com/docs/webclient/версия-Web-клиента/JsDocApi/, например, для Web-клиента 6.1 адрес будет следующим: https://docsvision.com/docs/webclient/6.1/JsDocApi/.

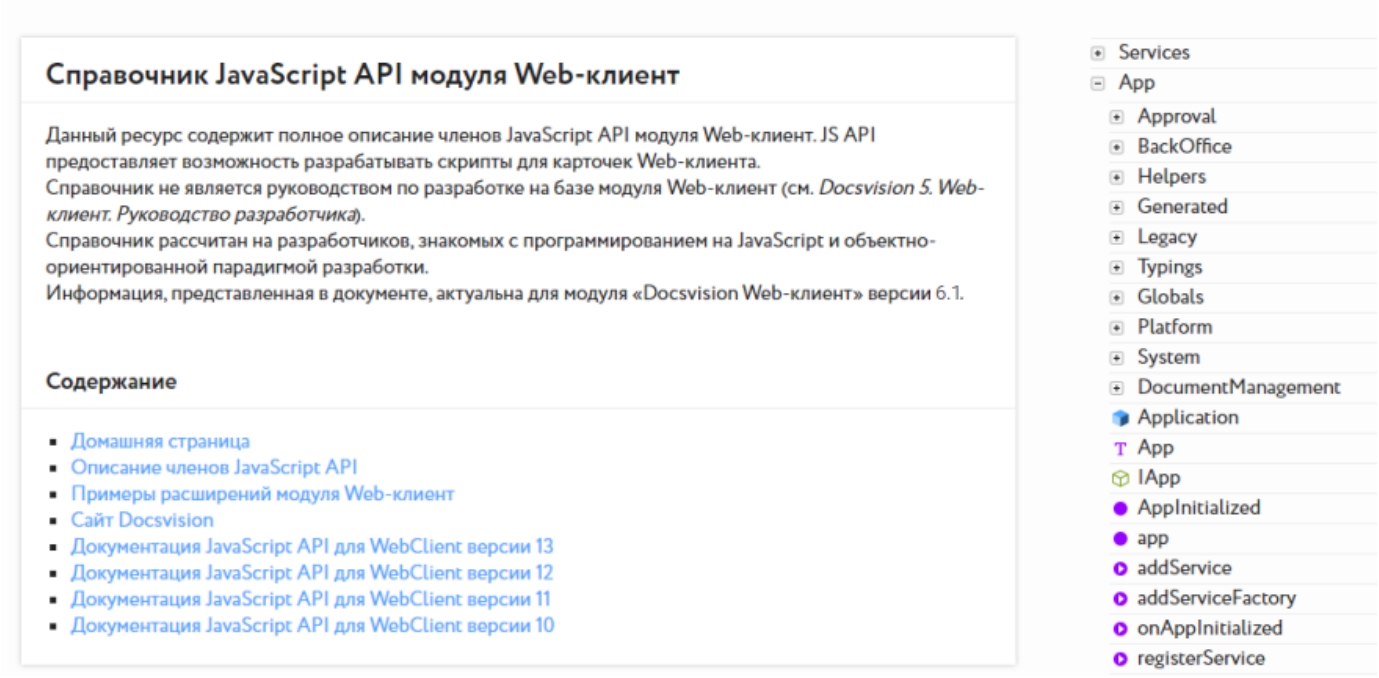


Рисунок 698. Главная страница справочника по JS API

Окно справочника содержит:

- Строку поиска,
- Флаги `Показывать protected-члены` (включает показ защищенных членов JS API) и `Показывать унаследованные члены` (включает показ унаследованных членов JS API),
- Меню для перехода к описанию элементов JS API,
- Легенду с описанием условных обозначений.

Как работать с описаниями сущностей

Чтобы получить описание для определённого класса, метода и т.п., нужно воспользоваться поиском или перейти к описанию из правого меню. К примеру, чтобы получить описание для класса элемента управления `Сотрудник`, нужно выполнить поиск по слову `Employee` (название класса элемента управления `Сотрудник`), или открыть из правого блока элемент **App › BackOffice › Controls › Employee › Employee ›** .



Значок слева от названия ссылки указывает, что данный раздел посвящен описанию класса. Условные обозначения всех типов сущностей приведены в легенде.

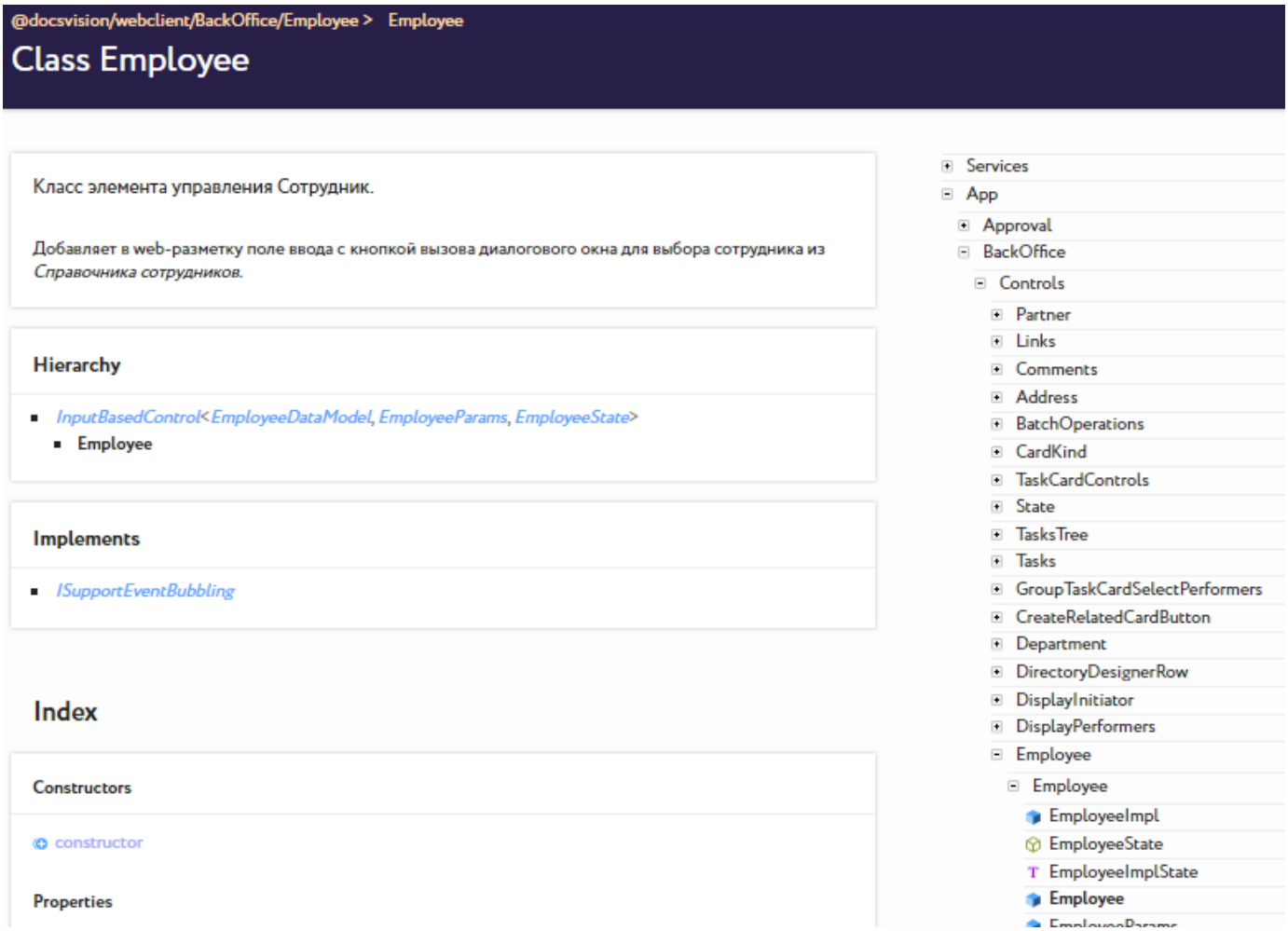


Рисунок 699. Страница с описанием класса Employee

На странице с описанием класса приведены:

- Общая информация о классе,
- Родительский класс,
- Конструкторы,
- Методы,
- Свойства и др.



В зависимости от типа описываемого члена JS API список может изменяться.

Некоторые члены JS API могут быть отмечены предупреждением:



Это внутренний нестабильный API, который может измениться в следующих версиях Web-клиент.

Использовать такой класс, интерфейс или метод следует с осторожностью, т.к. его сигнатура не стабильна и может измениться в следующих версиях Web-клиента.

Над названием сущности указан модуль Web-клиента (на изображении — `@docsvision/webclient/BackOffice/Employee`), содержащий данную сущность.

Чтобы перейти к описанию метода, полей и пр., кликните по ссылке с названием данного члена API.

Описание поля класса содержит:

- Название поля,
- Тип поля,
- Примечание.

Описание метода содержит:

- Название метода,
- Сигнатуру, примечание,
- Список параметров метода с типами,
- Тип возвращаемого значения.

Получение описания класса элемента управления

1. Определите название класса элемента управления.



Название класса ЭУ обычно совпадает с названием его типа, которое указано в свойстве `Название ЭУ` в программе Конструктор Web-разметок.

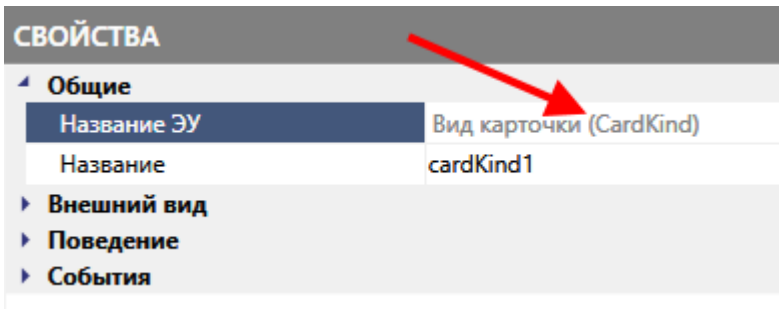


Рисунок 700. Тип элемента управления в программе Конструктор Web-разметок

2. Перейдите к описанию класса в справочнике:
   - Воспользуйтесь прямой ссылкой из описания элемента управления,

которое приведено в данной документации.

- Воспользуйтесь поиском по справочнику или правым меню.

Публичные свойства и события элемента управления программно связаны с его классом и содержатся в свойстве **params** класса.

Чтобы получить описание для класса публичных свойств ЭУ:

a. Найдите на странице секцию Properties.

b. Найдите описание свойства params.

c. Щелкните по ссылке с типом свойства. Будет открыто описание типа свойства params.

   Описание типа свойства params аналогично описанию других типов.

Чтобы получить описание для класса значения ЭУ:

a. Перейдите к описанию класса публичных свойств ЭУ.

b. Найдите на странице секцию **Properties**.

c. Найдите описание свойства **value**.

d. Щелкните по ссылке с типом свойства. Будет открыто описание типа свойства **value**.

   Описание типа свойства **value** аналогично описанию других типов.

Чтобы получить описание для класса события ЭУ:

a. Перейдите к описанию класса публичных свойств ЭУ.

b. Найдите на странице секцию **Properties**.

c. Найдите описание требуемого события — на уровне объектной модели ЭУ его событие является свойством ЭУ.

d. Щелкните по ссылке с типом события. Будет открыто описание класса.

К примеру, тип события **click** элемента управления `Кнопка`, которое возникает при нажатии, — `BasicApiEvent`. Тип аргумента, передаваемого с событием, указан в угловых скобках.

Событие типа `BasicApiEvent` (реализует интерфейс **IBasicEvent**) является непрерываемым, т. е. остановить обработку данного события другими обработчиками нельзя. Если тип события `CancelableApiEvent` (реализует интерфейс `ICancelableEvent`), то данное событие является

прерываемым (подробнее в разделе *Прерывание выполнения операции*).

Как работать с описанием веб-API

Справочник с описанием веб-методов, предоставляемых сервером Docsvision и его расширениями, доступен по адресу http://WebC-server.domain.com:5004/swagger/). Справочник также включает REST Web-API, доступный на стороне клиента Web-клиента, который может вызывать функции Docsvision API (работа с дискреционной безопасностью, карточками, файлами, метаданными и т.д.). API предназначается для работы из клиентских скриптов.



Страница Swagger доступна только пользователям из группы **DocsVision Administrators**. Выполнение запросов недоступно.

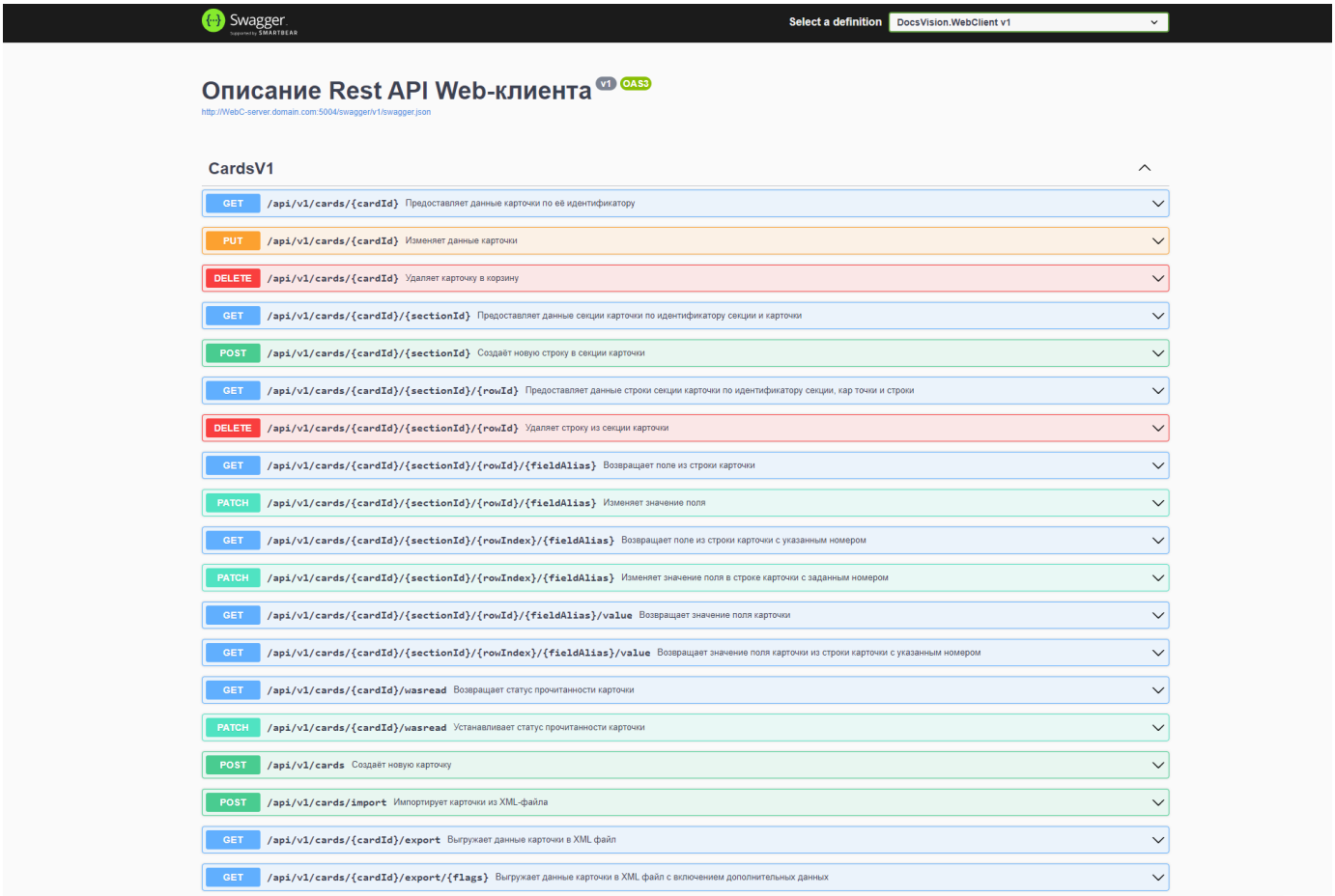


Рисунок 701. Страница описания веб-API

Описание методов сгруппировано по контролёрам, предоставляющим методы.

Чтобы перейти к описанию метода:

1. Нажмите на название контролёра, затем по названию метода. Будет раскрыт блок описания.

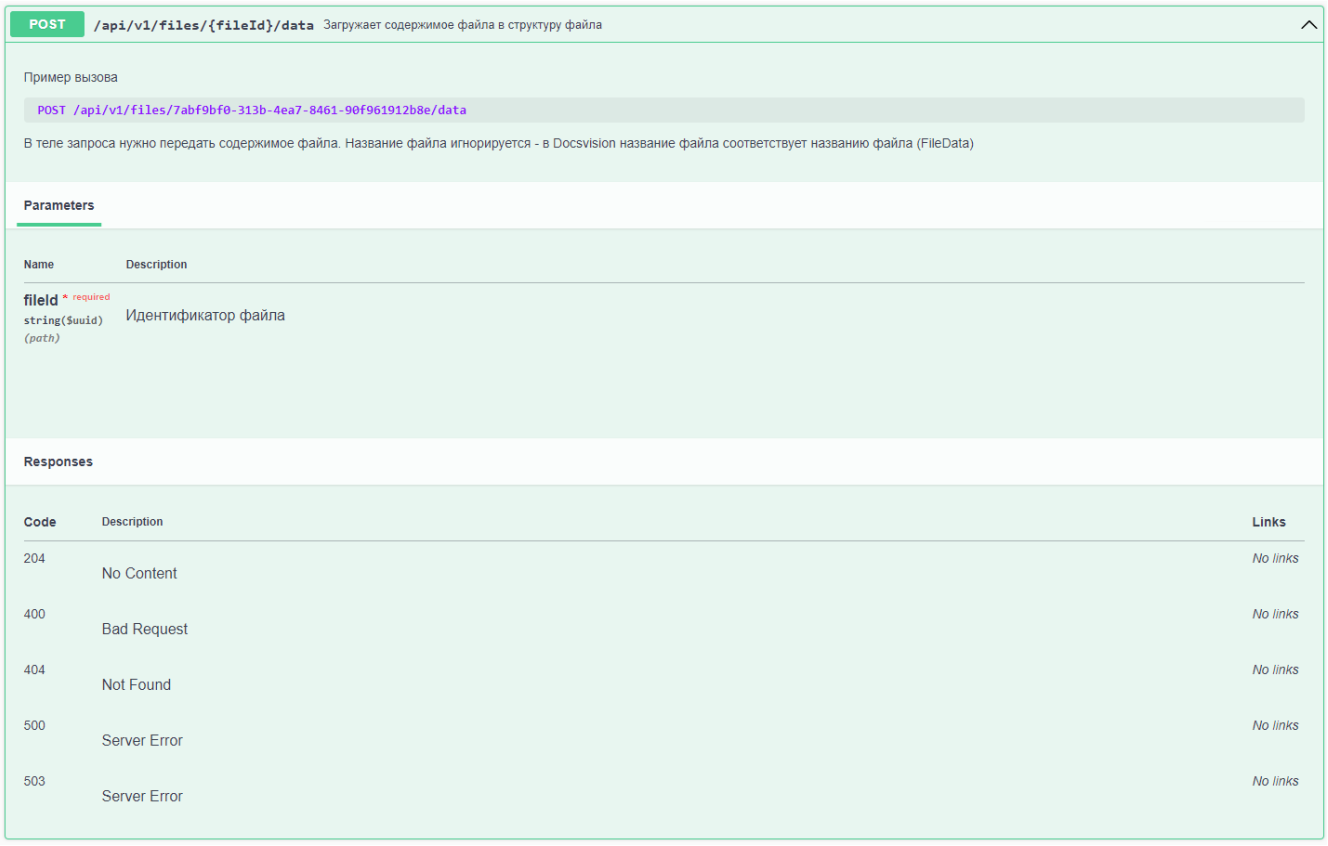


Рисунок 702. Описание метода "BarcodeSearchButton/{id}"

Описание метода включает описание модели возвращаемых данных и перечень параметров метода с примером из модели.

Как работать с описанием REST Web-API

В данной версии модуля с описанием REST API можно ознакомиться в справочнике веб-API, см. подробнее "[WebClient:how-to-use-swagger]".

Использование "общих справочников" в разработке

Функциональная возможность "общие справочники" включает режим использования данных из справочников только в режиме чтения и призвана сократить потребление памяти и повысить производительность любого приложения, реализованного с применением клиентского API Docsvision. Функциональность доступна для Web-клиента, начиная с версии 17, данная статья описывает взаимодействие с режимом при разработке собственных решений.

Если при разработке кода решения для работы со справочниками используется

сборка DocsVision.Platform.ObjectModel.dll (или DocsVision.BackOffice.ObjectModel.dll) решение будет автоматически работать и с общими справочниками. Стандартная логика Docsvision реализует автоматическое переключение режима работы со справочниками в зависимости от того читаются данные или изменяются.



Функциональность доступна только при использовании DocsVision.Platform.ObjectManager.Rest.dll в качестве API для работы с системой.

Если код решения взаимодействует со справочниками через DocsVision.Platform.ObjectManager.dll разработчику доступны два варианта:

- При работе на стороне Web-клиента для доступа к общим справочникам необходимо использовать методы класса CardManager — GetDictionaryAsReadonly и GetDictionaryDataAsReadonly. Эти методы передают доступные только для чтения CardData запрошенных справочников.
- Если код работает вне Web-клиента, необходимо до использования методов GetDictionaryAsReadonly и GetDictionaryDataAsReadonly создать сессию, через которую будет осуществляться чтение данных справочников и установить эту сессию как общую путём вызова SessionManager.SetCommonSession. Для создания сессии следует использовать УЗ, обладающую правами чтения всех данных справочников, например, состоящую в группе **DocsVision Administrators**.

Основные особенности использования функциональности общих справочников

1. При использовании функциональности общих справочников нельзя использовать операции, которые их модифицируют. Данные справочников в режиме чтения нельзя передавать как аргументы в методы, где они могут быть изменены, даже если изменения требуются крайне редко. Несоблюдение этого правила приведёт к ошибкам, источник которых сложно определить.
2. Вызов RowData.Refresh у строчек справочников в режиме чтения приведёт к ошибке для данных, полученных таким образом.
3. Для сохранения совместимости не рекомендуется возвращать справочники в режиме чтения в старых публичных методах, где возвращались справочники не в режиме чтения.
4. Рекомендуется использовать интерфейс IReadOnly<T> при работе со справочниками в режиме чтения через классы ObjectManager, см. ниже.

5. При работе со справочниками через API `ObjectModel` дополнительных действий не требуется.
6. Справочники в режиме чтения не повышают производительность в однопользовательских приложениях как, например, Конструктор Web-разметок или Панель управления Web-клиентом.

Использование интерфейса IReadOnly<T>

При вызове методов с параметрами, которые имеют типы справочников или строчек справочников из `ObjectManager` (`CardData`, `RowData`, `FolderCard`, `Folder` и т.д.) во время компиляции отсутствует проверка того, что в качестве аргументов были переданы справочники или строки в режиме чтения. В результате разработчик может случайно изменить вызываемый метод, добавив в него код модификации для передаваемых в качестве аргументов справочников, не зная, что в этот метод могут быть переданы справочники в режиме чтения. Если аргументы попадут в метод по цепочке вызовов, отследить такие операции, определить, в каком режиме они используются, будет сложно.

Рекомендуется использовать в объявлении аргументов интерфейс-обертку `IReadOnly<T>`, чтобы не забывать, что метод работает со справочниками в режиме чтения.

Интерфейс служит напоминанием о том, что обертываемый объект (справочник или строка) может быть в режиме чтения. Обычный справочник или строка справочника могут быть приведены к `IReadOnly<T>` с помощью метода расширения `AsReadOnly()`, что является безопасным, т.к. с обычным справочником можно делать то же, что со справочником в режиме чтения плюс модификация и обновление.

Интерфейс `IReadOnly<T>` содержит метод `Apply`, который используется, например, следующим образом:

```
IReadOnly<FolderCard> folderCard = ⋯
IReadOnly<Folder> folder = folderCard.Apply((FolderCard f)=>f.GetFolder(folderId));
```

Если типы не проставлены явно:

```
Var folder = folderCard.Apply( f => f.GetFolder(folderId));
```

Метод-расширение `Unwrap()` возвращает обертываемое значение. Его

использование является небезопасным, но необходимым. Использовать `Unwrap` желательно как можно позже, в противном случае использование маркерного интерфейса теряет своё значение. Метод реализован как расширение, чтобы не вызывать ошибку при вызове, когда объект типа `IReadOnly<T> null`.

Расширение возможностей клиентской части Web-клиента

Клиентское расширение — исполняемая в веб-браузере пользователя программа, реализующая дополнительную функциональность клиентского уровня.

Клиентское расширение может добавлять:

- Локализованные ресурсы, которые можно использовать при формировании локализованного пользовательского интерфейса.
- Элементы управления (клиентскую часть).
- Обработчики событий, которые могут быть вызваны из разметки,
- Сервисы (публичные методы).

Создание и публикация клиентского расширения

Клиентское расширение разрабатывается на языке TypeScript с последующей компиляцией в JavaScript.

Чтобы создать клиентское расширение:

1. Скопируйте к себе в рабочую папку каталог `ProjectTemplates\TemplateWebExtension\` из репозитория WebClient-Samples.
2. Настройте проект:
3. Переименуйте папку `TemplateWebExtension` — укажите название расширения (например, "MyWebExtension").
4. В файле `MyWebExtension/copy-path.js` измените значения:
   - `SITE_ROOT` — укажите относительный путь к папке, в которую будут выгружены файлы, полученные при сборке.
   - `EXTENSION_NAME` — укажите название расширения (в данном примере — "MyWebExtension").
5. Откройте папку `MyWebExtension` в командной строке и выполните команду: `npm install`.
6. Создайте в папке `MyWebExtension/src` файл `ts/tsx` с кодом расширения.



Примеры клиентских расширений приведены в следующих

пунктах раздела, а также в репозитории "ДоксВижн".

7. Измените содержимое файла `src/Index.ts`. В файле должна выполняться регистрация расширения с учётом реализованных в нем функций:

```
extensionManager.registerExtension({
    name: "My web Extension",
    version: "1.0",

    getLocalizations: ... , ①
    controls: ... , ②
    globalEventHandlers: ... , ③
    globalServices: ... , ④
    layoutServices: ··· ⑤

})
```

① Расширение добавляет локализованные ресурсы.

② Расширение добавляет элементы управления.

③ Расширение добавляет обработчики событий.

④ Расширение добавляет публичные сервисы.

⑤ Расширение добавляет публичные сервисы, имеющие особое поведение в рамках каждой конкретной разметки.

Описание изменений, которые необходимо выполнить в файле `Index.ts`, приведено в описании примеров расширений, добавляющих обработчики событий, локализованные ресурсы и клиентские компоненты элементов управления.

1. Откройте папку `MyWebExtension` в командной строке и выполните команду: `npm run build`.

   Данная команда выполнит компиляцию проекта. В папку, указанную в параметре `SITE_ROOT`, будут выгружены файлы: `extension.js`, `extension.js.map`, `extension.css` и `extension.css.map`. Файлы `.map` используются для отладки кода в браузере.

   Npm может отслеживать изменения в исходных файлах, и при этом самостоятельно выполнять сборку. Выполните команду, чтобы включить режим отслеживания: `npm run`

```
            watch.
```

2. Скопируйте папку `%SITE_ROOT%\Content/Modules/MyWebExtension` в `/lib/docsvision/webclient/Content/Modules`.
3. После завершения отладки клиентского расширения может быть собрана "релизная версия". Для этого, находясь в командной строке в папке MyWebExtension, выполните команду: `npm run build:prod`.

Действие данной команды аналогично действию команды `npm run build`, за исключением того, что к полученным файлам будет применена процедура минификации (минимизации). Полученные файлы также необходимо скопировать в папку `/lib/docsvision/webclient/Content/Modules`.



Информация о загруженных клиентских расширениях отображается в Web-клиенте в окне "О программе".

Разработка расширения с обработчиками событий

Данный тип расширений предназначен для создания функций обработки событий разметки. Такие функции включают, например, функцию открытия карточки, функцию нажатия кнопки, функцию выбора значения из списка.

Одно клиентское расширение может содержать несколько функций-обработчиков.

1. Создайте проект клиентского расширения **без публикации**.
2. Добавьте код функции-обработчика в существующий файл `src/EventHandlers.tsx` или в новый файл TypeScript (`.ts`, `.tsx`) в папке `src`.

Следующий код демонстрирует пример функции-обработчика.

```
import { Employee } from "@docsvision/webclient/BackOffice/Employee";
import { IEventArgs } from "@docsvision/webclient/System/IEventArgs";
import { MessageBox } from "@docsvision/webclient/Helpers/MessageBox/MessageBox";

export function someHandler(sender: Employee, e: IEventArgs) { ①

 MessageBox.ShowInfo(sender.params.value.displayName); ②
```

```
}
```

① Пример обработчика.

② Отображает сообщение с именем сотрудника.

В первых строчках кода выполняется импорт внешних сущностей, типы которых используются в коде, из JS-модулей Web-клиента. При использовании Visual Studio Code[7] редактор автоматически подключит необходимые модули. При использовании другого редактора — обратитесь к справочнику по JS API, в котором указаны названия JS-модулей.

Функция `someHandler` — реализация обработчика

- В определении функции должно быть указано ключевое слово `export`.
- Тип аргумента `sender` должен соответствовать типу элемента управления, событие которого обрабатывается. В примере, `sender` с типом `Employee` обрабатывается событие элемента управления `Сотрудник`.
- Тип аргумента `e` зависит от обрабатываемого события:
  - `IEventArgs` — если событие вызывается после выполнения действия, т.е. не может его прервать.
  - `ICancelableEventArgs` — если событие вызывается при выполнении действия, т.е. может его прервать. Работа с прерываемыми действиями рассмотрена в пункте Прерывание выполнения операции.

3. Если на шаге 2 был создан новый tsx файл, добавьте в файле `src/Index.ts` строку импорта его сущностей.

```
import * as Handlers from "./FileWithHandlers"; ①
import { extensionManager } from "@docsvision/webclient/System/ExtensionManager";

extensionManager.registerExtension({
 name: "My web Extension",
 version: "1.0",
 globalEventHandlers: [Handlers] ②
})
```

① Импорт сущностей из файла FileWithHandlers (расширение файла не указывается).

② В массив globalEventHandlers нужно передать ссылку на импорт сущностей из файла FileWithHandlers.

4. Соберите проект клиентского расширения командой `npm run build`.
5. Скопируйте полученный файл `%BuildDir%/extension.js` на сервер Web-клиента в папку `/lib/docsvision/webclient/Content/Modules/%Каталог Решения%`.



Обратитесь к разделу Создание и публикация клиентского расширения, чтобы получить больше информации, связанной со сборкой проекта.

6. Настройте обработчик в программе Конструктор Web-разметок.
7. Откройте в программе Конструктор Web-разметок разметку, событие которой обрабатывается.
8. Выберите элемент управления, для которого настраивается обработчик события.
9. Укажите название функции-обработчика, реализованной в клиентском расширении, в значении события.

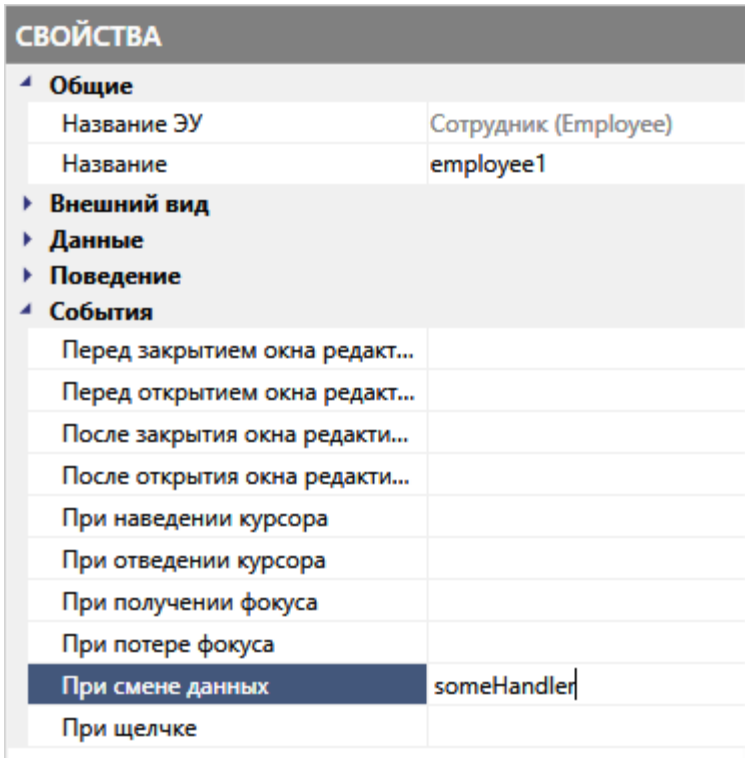


Рисунок 703. Настройка обработчика для события "При смене данных" в программе Конструктор Web-разметок

10. Сохраните настройки разметки.



Инструкция по работе в программе Конструктор Web-разметок приведена в документации по работе с программной Конструктор Web-разметок.

11. Откройте локацию с настроенной разметкой (например, карточку) и выполните действие, вызывающее реализованный обработчик.
12. После завершения отладки скрипта, соберите и опубликуйте на сервере Web-клиента "релизную версию" клиентского расширения. "Релизная версия" собирается командой `npm run build:prod`.

Получить все обработчики событий

При необходимости, например для поиска и исправления искать ошибок, можно получить имена всех скриптов, назначенных на разметку. Для этого выполните в консоли браузера следующий скрипт:

```
(function getHandlers(sender) {
    let controlsVisitor = new WebClient.ControlsRecursiveVisitor();
    controlsVisitor.visitDeep(sender.layout, (control) => {
        const paramsKey = Object.keys(control.params);
```

```
        const eventsKey = paramsKey.filter(key => control.params[key] instanceof
WebClient.BasicEvent);
        const handlers = eventsKey.reduce((handlers, key) =>
            control.props[key] && !handlers.includes(control.props[key])
                ? `${control.props[key]}${handlers.length ? ', ' : ''}${handlers}`
                : handlers
            , "");
        if (handlers !== "") {
            console.info(`${control.props.name}: ${handlers}`);
        }
    });
})(layoutManager.cardLayout);
```

Получение доступа к разметке

Прежде чем работать с элементами управления разметки, необходимо в скрипте получить доступ к данной разметке. Это можно сделать несколькими способами.

Получение разметки, из которой вызван обработчик события

```
export function someHandler(sender: Employee, e: IEventArgs) {
    let currentLayout = sender.layout;
  }
```

Получение разметки по имени локации

```
let someLayout = layoutManager.getLayout("DV.MainMenu");
```

В метод getLayout нужно передать название локации с разметкой:

- *DV.Dashboard* — главная странице (страница "дашборд").
- *DV.MainMenu* — основное меню (с деревом папок).
- *DV.GridFilter* — панель фильтра папки.
- *DV.SearchParameters* — панель поисковых параметров.
- *DV.UserProfile* — диалоговое окно профиля пользователя.

Локация, разметка которой получается, должна быть на текущей странице Web-клиента.

Получение разметки открытой карточки:

Для быстрого доступа к разметке из консоли браузера можно использовать

`layoutManger.cardLayout`. Однако использование в расширениях этого свойства не рекомендуется, следует использовать `sender.layout`.

```
let cardLayout = layoutManager.cardLayout;
```

В примерах кода используются функции объекта `layoutManager` (менеджер разметок). Для импорта объекта добавьте в скрипт следующий код:

```
import { layoutManager } from "@docsvision/webclient/System/LayoutManager";
```

Работа с элементами управления

Элементы управления являются частью разметки. Ссылка на коллекцию элементов управления разметки содержится в её поле `controls`:

```
let layoutControls = someLayout.controls; ①
```

① someLayout — разметка.

Доступ к элементу управления осуществляется по названию, которое ему присвоено в программе Конструктор Web-разметок:

```
let registrar = layoutManager.cardLayout.controls.get<Employee>("registrar"); ①
```

① Получаем элемент управления Сотрудник (Employee) с названием "registrar" из разметки карточки (layoutManager.cardLayout).

Доступ к свойствам и значению элемента управления осуществляется через его поле `params`:

```
let registrarName = registrar.params.value.displayName; ①

registrar.params.visibility = false; ②
```

① Получаем значение элемента управления registrar (params.value), из которого получаем отображаемое имя сотрудника (displayName).

② Изменяем значения свойства Видимость на "false" (элемент управления будет скрыт).

Изменение значения элемента управления осуществляется через свойство **params.value**.

Способ изменения зависит от типа значения:

Для простых типов (string, bool, int и пр.) значение изменяется напрямую:

```
let onControl = layoutManager.cardLayout.controls.get<CheckBox>("onControl"); ①

onControl.params.value = true; ②
```

① Получаем из разметки карточки элемент управления Флаг (CheckBox) с названием "onControl".

② Изменяем значения элемента управления на true (флаг будет установлен).

Для элемента управления Дата/время значение должно быть предварительно сформировано:

```
let endDate = layoutManager.cardLayout.controls.get<DateTimePicker>("endDate"); ①

endDate.value = new Date("2019-03-01T12:00:00"); ②
```

① Получаем из разметки карточки элемент управления Дата/время (DateTimePicker) с названием "endDate".

② Изменяем значения элемента управления на дату 01.03.2019 и время 12:00:00.

Для элементов управления, работающих с моделями объектов, значение должно быть предварительно сформировано:

```
export async function someHandler(sender: CustomButton, e: IEventArgs) { ①

    let departmentControllerService = sender.layout.getService($DepartmentController); ②

    let departmentModel = await departmentControllerService.getStaffDepartment("67E331F0-
EF31-4E2C-A1F6-D527606EC8CA"); ③

    let department = sender.layout.controls.get<Department>("department"); ④

    department.params.value = departmentModel; ⑤
}
```

① Функция является асинхронной.

② Получаем сервис, загружающий модель объекта для элемента управления Подразделение и Подразделение контрагента.

③ Получаем модель объекта для подразделения с идентификатором "67E331F0-EF31-4E2C-A1F6-D527606EC8CA".

④ Получаем из разметки карточки элемент управления Подразделение (Department) с названием "department".

⑤ Сохраняем в значение элемента управления полученную ранее модель. В элементе управления будет выбрано подразделение с идентификатором "67E331F0-EF31-4E2C-A1F6-D527606EC8CA".



Список сервисов генерации моделей объектов приведён в пункте Сервисы генерации моделей объектов.

Управление блокировкой элемента управления от изменений осуществляется с помощью параметра disabled:

```
let control = layoutManager.cardLayout.controls.superControl; ①

control.params.disabled = true ②
```

① Получение элемента управления superControl.

② Блокировка элемента управления.
Параметр *disabled* работает совместно с параметром *editOperation*:

- если *disabled* равен `false` и *editOperation* разрешён — редактирование ЭУ разрешено;
- иначе — редактирование ЭУ запрещено.

Для отрисовки заблокированного элемента управления вызывается метод `BaseControlImpl.renderDisabledControl`, который может быть переопределён, если требуется собственная реализация отрисовки заблокированного элемента управления.

Получение номера в нумераторе

Следующий код демонстрирует пример получения номера в нумераторе перед изменением состояния карточки. Предполагается, что функция будет указана в качестве обработчика для события изменения состояния карточки.

```
export async function onStateChanging(sender: Layout, e: CancelableEventArgs
<ICardStateChangingEventArgs>) {
①
    let regNumber = sender.controls.get<Numerator>("regNumber");

    if (!regNumber.hasValue()) {
        e.wait(); ②
```

```
        try {
            await regNumber.generateNewNumber() ③
            console.log("Присвоен регистрационный номер");
            e.accept(); ④
        } catch (err) {
            MessageBox.ShowError("Ошибка при получении регистрационного номера: " + err);
            e.cancel();
        }
    }
}
```

① Получаем элемент Нумератор из разметки.

② Приостанавливаем изменения состояние карточки.

③ Используем метод generateNewNumber для получения нового номера. При этом будет выполнено сохранение карточки. Метод generateNewNumber нужно вызывать асинхронно.

④ Продолжаем обработку изменения состояния.

Получение информации из ЭУ "Записи справочника сотрудников"

Для выбора записей из *Справочника сотрудников* предназначен элемент управления "Записи справочника сотрудников". В зависимости от настроек пользователь может выбирать сотрудников, организации, подразделения, группы, роли и должности, а также поисковые слова.

Каждый из этих типов имеет свою модель данных:

- Сотрудники: `EmployeeDataModel`.
- Организации и подразделения: `DepartmentModel`.
- Группы, роли, должности, поисковые слова: `StaffDataModel`

Поля, которые содержит каждая модель, можно посмотреть в справочнике по JavaScript API.

Чтобы корректно в коде обрабатывать полученные значения из ЭУ `Записи справочника сотрудников` нужно явно указывать тип. Если в области выбора элемента указан только один из типов, в коде достаточно прописать этот тип:

```
sender.layout.controls.get<StaffDirectoryItems<GenModels.EmployeeDataModel>>(название
ЭУ);
```

Пример для ЭУ с множественным выбором:

```
sender.layout.controls.get<StaffDirectoryItems<GenModels.EmployeeDataModel[]>>(название
ЭУ);
```

Если в области выбора указано несколько типов, необходимо написать логику, определяющую тип элемента. Пример такой функции приведён ниже:

```
function showName(sender: Layout) {
    const control = sender.layout.controls.get<StaffDirectoryItemsSingle>(Название ЭУ);

    if(control.value.dataType === GenModels.DirectoryDataType.Employee) {

        const value = control.value as GenModels.EmployeeDataModel;

        console.log(value.firstName + " " + value.lastName); ①

    } else if (control.value.dataType === GenModels.DirectoryDataType.Department

        || control.value.dataType === GenModels.DirectoryDataType.Organization) {

            const value = control.value as GenModels.DepartmentModel;

            console.log(value.fullName); ②
    } else {

        const value = control.value as GenModels.StaffDataModel;

        console.log(value.name); ③
    }
}
```

① Логика работы с `value` типа `EmployeeDataModel`.

② Логика работы с `value` типа `DepartmentModel`.

③ Логика работы с `value` типа `StaffDataModel`.

Если установлен множественный выбор, код аналогичный. При получении элемента управления необходимо указать тип значения `StaffDirectoryItemsMultiple` вместо `StaffDirectoryItemsSingle`, далее циклом пройти все значения и в зависимости от типа совершить определенные действия с каждым из значений.

Проверка данных

Данный раздел содержит описание примера проверки действительности данных.

Выполнить проверку, что заполнено определенное текстовое поле и количество символов не превышает 100.

Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт с функцией `validateTextBoxControl`, реализующей валидацию данных ЭУ.

Ссылка на пример на GitHub: ControlValidation.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке `ControlValidationWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
2. Скопируйте каталог `SamplesOutput\Content\Modules\ControlValidationWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Импортируйте решение из файла `ControlValidationLayout.xml`.
3. Сделайте решение *ControlValidationLayout* активным для карточки типа Документ вида *ДокументУД/Исходящий*.
4. На разметке расположен ЭУ `Текст` и кнопки сохранения/отмены.
5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
6. Откройте документ ДокументУД\Исходящий и нажать кнопку редактировать. При этом должна открыться разметка с ЭУ `Текст` и кнопками сохранения.
7. Если в ЭУ `Текст` нет текста или длина текста превышает 100 символов, то при нажатии кнопки **Сохранить** возникает сообщение об ошибке и изменения не сохраняются.



Данный пример демонстрирует только доступ к элементу разметки. При реализации расширений рекомендуется использовать экспорт разметок в виде Решения.

Копирование значения ЭУ

Данный раздел содержит описание примера взаимодействия ЭУ: копирование значения. При завершении ввода значения в один ЭУ, это же значение копируется в другой. Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт с функцией `copyValueToControl`, реализующей копирование данных одного ЭУ в другой.

Ссылка на пример на GitHub: CopyValueToControl.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке CopyValueToControlWebExtension и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`
2. Скопируйте каталог SamplesOutput\Content\Modules\CopyValueToControlWebExtension в каталог "Путь к сайту Web-клиента\Content\Modules"
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Импортируйте решение из файла `CopyValueToControlLayout.xml`.

   Либо создайте собственное решение:

   a. Создайте для расширения разметку.

   b. Добавьте в разметку `Блок`, в него два ЭУ `Текст`.

   c. Задайте для первого ЭУ `Текст` режим редактирования по месту, на событие **`После смены данных`** добавьте вызов функции `copyValueToControl:CopyValueToControlExt`.
3. Сделайте решение `CopyValueToControlLayout` или собственное активным для карточки типа *Документ* вида *ДокументУД/Исходящий*.
4. На разметке расположено два ЭУ `Текст`.
5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

6. Откройте документ *ДокументУД\Исходящий* и нажать кнопку **Редактировать**. При этом должна открыться разметка с двумя ЭУ `Текст`.
7. Если в первый из них добавить текст, текст появится во втором.



Данный пример демонстрирует только доступ к элементу разметки. При реализации расширений рекомендуется использовать экспорт разметок в виде Решения.

Получение информации о карточке, разметке

JS API предоставляет возможность получать в скрипте информацию об открытой карточке и разметке, с использованием которой она открыта.

Следующий код демонстрирует пример получения информации о карточке и разметке:

```
export function showInformation(sender: Layout, e: IEventArgs) {
    let cardInfo = sender.getService($CardInfo); ①
    let cardId = sender.getService($CardId);
    console.log("Идентификатор карточки: " + cardId);
    console.log("Тип карточки: " + cardInfo.typeId);
    console.log("Вид карточки: " + cardInfo.kindId);

    let layoutInfo = sender.getService($LayoutInfo); ②
    console.log("Название разметки: " + layoutInfo.name);
    console.log("Тип разметки: " + layoutInfo.type);
    console.log("Тип пользовательского устройства: " + layoutInfo.deviceType);
}
```

① Информация о карточке из сервиса `$CardInfo`.

② Информация о разметке из сервиса `$LayoutInfo`.

Получение данных карточки на стороне клиента

Web-клиент предоставляет REST API для получения любых данных, см. подробнее "[WebClient:how-to-use-swagger]".

Отображение сообщений, запросов, предупреждений

Сервис отображения окна сообщений

Показать на экране простое информационное сообщение можно с помощью сервиса `$MessageWindow`.

Основные методы этого сервиса: `showInfo`, `showWarning`, `showError`. Они различаются только цветом оформления и отображают диалог следующего вида:

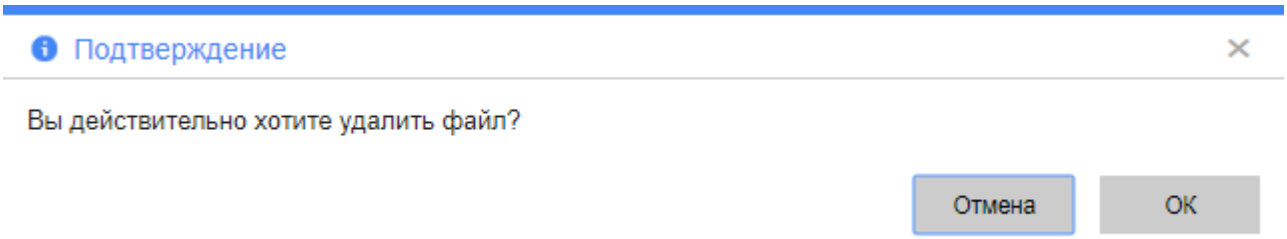


Рисунок 704. Диалоговое окно подтверждения

В качестве параметра для этих методов можно передать как простую строку, так и React-разметку для отображения произвольного содержимого.

```
export async function onButtonClick(sender: LayoutControl, args: IEventArgs) {
     const messageWindow = sender.getService($MessageWindow);
     await messageWindow.showConfirmation("Вы действительно хотите удалить файл?");
     deleteFile();
}
```

Работа с боковой панелью Web-клиента

Боковая панель Web-клиента — отображаемая по запросу правая панель, на которую могут быть добавлены требуемые элементы управления. Данный компонент реализован в классе `ModalSidebar` JS-модуля `@docsvision/webclient/Helpers/ModalSidebar/ModalSidebar`.

Следующий код демонстрирует пример использования боковой панели для отображения текущего времени.

```
export function showSidebar(sender: CustomButton, e: IEventArgs) { ①
    let host: ModalHost = new ModalHost("time-dialog", () => {
        return (
            <ModalBackdrop visible={true} onClick={() => host.unmount()} > ②
                {/* Элемент Боковая панель */}
                <ModalSidebar isOpen={true}>

                  {/* Кнопка */} ③
                  <ModalSidebarCloseButton onClick={() => host.unmount()} />

                  {/* Шапка */} ④
                  <ModalSidebarHeader>Заголовок панели</ModalSidebarHeader>

                  {/* Содержимое */} ⑤
```

```
                    <div>Текущее время: {(new Date()).toTimeString()}</div>
                </ModalSidebar>
            </ModalBackdrop>
        );
    });

    host.mount(); ⑥

    setInterval(() => host.forceUpdate(), 1000); ⑦
}
```

① Создаем контейнер для размещения модальных окон, которым является боковая панель.

Первый параметр конструктора `ModalHost` — класс, который будет задан элементу-контейнеру в DOM. Второй параметр — функция, выполняющая отрисовку модального окна.

② Элемент `ModalBackdrop` создаёт темный фон при отображении боковой панели.

При клике по элементу `ModalBackdrop` боковая панель будет закрыта.

③ Кнопка для закрытия боковой панели.

④ Заголовок боковой панели.

⑤ Содержимое боковой панели. В данном случае на боковой панели отображается текущее время.

⑥ Вызов метода, добавляющего проинициализированный контейнер с боковой панелью в DOM.

⑦ Периодически вызываемый метод перерисовки контейнера с боковой панелью. Метод обновляет отображаемое значение текущего времени.

Обработка события изменения состояния карточки

При изменении состояния карточки из пользовательского интерфейса Web-клиента вызывается событие **Перед изменением состояния карточки** разметки карточки (корневой элемент разметки), которое может быть обработано в клиентском скрипте.

Ссылка на пример на GitHub: ChangeStateByScript.

Данное событие является прерываемым — может быть использовано для отмены изменения состояния карточки.

Ниже приведён код обработчика события изменения состояния карточки (Документ, Задание):

```
async function changeState(sender: Layout, e: CancelableEventArgs
<ICardStateChangingEventArgs>) {
    e.wait();

    if (await myCheckPossibility()){ ①
        e.accept(); ②
    } else {
        sender.getService($MessageWindow).showWarning("Операция " + e.data.operationId  +
" запрещена"); ③
        e.cancel(); ④
    }
}
```

① Функция `myCheckPossibility` — пользовательский код, содержащий собственную логику.

② Подтверждаем выполнение операции. Обязательно, если вызвали `e.wait`.

③ Показываем сообщение о запрете операции с указанием идентификатора операции.

④ Для отмены операции, при необходимости, вызываем `e.cancel`.

Для карточки *Задание* после изменения состояния карточки будет вызвано событие **После изменения состояния карточки** элемента управления Кнопки операций задания.

Изменение внешнего вида и заголовка

Данный раздел содержит описание примера изменения внешнего вида и заголовка Web-клиента.

Ссылка на пример на GitHub: ChangeLogoAndBackground.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке `ChangeLogoAndBackgroundWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.

2. Скопируйте каталог `SamplesOutput\Content\Modules\ChangeLogoAndBackgroundWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Откройте стартовую страницу или любую папку.



Данный пример демонстрирует только изменение внешнего вида и заголовка системы с помощью скрипта.

Сохранение изменений карточки

Код сохраняет все изменения карточки:

```
export async function saveCard(sender: CustomButton, e: IEventArgs) {
    await MessageBox.ShowConfirmation("Сохранить карточку?");

    layoutManager.cardLayout.saveCard(); ①
}
```

① Получаем разметку карточки (cardLayout) и вызываем её метод saveCard для сохранения.

Чтобы сохранить изменения определённого элемента управления, используйте его метод save:

```
export async function saveDescription(sender: CustomButton, e: IEventArgs) {
    let cardDescription = sender.layout.controls.get<TextArea>("Description"); ①

    await cardDescription.save(); ②
}
```

① Получаем элемент управления Description, данные которого сохраняются.

② Вызываем метод save для сохранения изменений.



Если за сохранением следует другая операция, изменяющая карточку, сохранение карточки может привести к ошибке `Содержимое карточки было изменено другим пользователем. Пожалуйста, обновите карточку и выполните операцию снова`.

Например, данная ошибка возникнет при сохранении карточки, если сохранение вызывается в обработчике изменения состояния

карточки.

Чтобы обойти данную ошибку, приостановите обработку основного события до сохранения карточки, а сохранение сделайте асинхронным. Следующий код демонстрируется данный способ:

```
export async function onChangeState(sender: Layout, e: CancelableEventArgs
<ICardStateChangingEventArgs>) {
        let description = sender.controls.get<TextArea>("Description");

        description.value = description.value + "\nЗарегистрировано"; ①

        e.wait(); ②
        await sender.layout.saveCard(); ③
        e.accept(); ④
}
```

① Изменяем значение поля карточки.

② Приостанавливаем обработку основного события.

③ Асинхронное сохранение.

④ Продолжаем обработку.

Обработка события сохранения карточки

При сохранении карточки последовательно вызываются события **Подготовка к сохранению карточки**, **Перед сохранением карточки** и **При сохранении карточки** корневого элемента разметки карточки. Данные события могут быть обработаны в пользовательской функции.

Событие **Перед сохранением карточки** является прерываемым, т. е. предоставляет возможность отменять сохранение изменений карточки в зависимости от условий:

```
function onSaveCard(sender: Layout, e: CancelableEventArgs<ICardSavingEventArgs>) {
    if (CheckPossibility()){ ①
        e.accept(); ②
        return;
    }

   MessageBox.ShowWarning("Сохранение отклонено");
   e.cancel(); ③
```

```
}
```

① Функция CheckPossibility проверяет выполнение неких требований

② Подтверждение сохранения. Необязательно.

③ Отмена сохранения, если условия не выполняются.



Если при сохранении карточки нужно изменить значение элемента управления разметки (и данное изменение должно быть включено в сохранение), изменение следует применять до сохранения в обработчике события **Подготовка к сохранению карточки**.

Обработка события элемента Кнопки сохранения

Другой способ обработать событие попытки сохранения карточки — подписаться на событие **Перед щелчком** элемента управления Кнопки сохранения. В аргументе события будет передан тип нажатой кнопки, который может использоваться при реализации логики обработчика.

```
export function onClicking(sender: SavingButtons, e: ICancelableEventArgs
<ISavingButtonClickEventArgs>) {
   if (e.data.button == SavingButton.Save){ ①
   }else{ ②
   }
}
```

① Обработка события, когда нажата кнопка Сохранить/ОК.

② Обработка события, когда нажата кнопка Отмена.

Прерывание выполнения операции

Функция, настроенная для обработки события, возникающего при выполнении операции, позволяет прервать её выполнение, если аргумент события имеет интерфейс ICancelableEventArgs. Данный интерфейс определяет два метода: cancel — прерывает выполнение операции; accept — продолжает выполнение операции.

Следующий код демонстрирует пример обработчика события сохранения карточки, прерывающего сохранение, если в названии карточки (cardName) содержится слово "договор".

```
export function onCardSaving(sender: Layout, e: ICancelableEventArgs<ISaveControlData>) {
```

```
①

    let cardName = sender.controls.get<TextBox>("cardName"); ②

    if (cardName.params.value.includes("договор")) { ③
        MessageBox.ShowError("Для оформления договоров используйте вид документа
'Договор'");
        e.cancel(); ④
    } else {
        e.accept(); ⑤
    }
}
```

① Метод `onCardSaving` настроен в качестве обработчика события, вызываемого перед сохранением карточки

Список параметров метода соответствует аргументам события `cardSaving` класса `LayoutParams`

② Получение элемента управления с названием карточки.

③ Если в названии используется слово "договор", сохранении карточки будет прервано.

④ Метод `cancel` прерывает операцию сохранения.

⑤ Метод `accept` продолжает операцию сохранения.

Обычно вызов метода `accept` не является обязательным. Однако `accept` или `cancel` обязателен в случае, если вызван `wait`.

Другой пример демонстрирует использование метода `wait`, который позволяет отложить решение до момента выполнения асинхронной операции (например, запроса к серверу или показа диалога подтверждения).

```
export async function onCardSaving(sender: Layout, e: ICancelableEventArgs
<ISaveControlData>) {

    e.wait(); ①
    try {
        await MessageBox.ShowConfirmation("Вы действительно хотите сохранить карточку?");
②

        e.accept(); ③
    } catch {
        e.cancel(); ④
```

```
    }
}
```

① Отменяем автоматический вызов `e.accept()`.

② Показ диалога с ожиданием решения пользователя.

③ Разрешаем выполнение операции.

④ Отменяем событие.



Чтобы определить, является ли событие прерываемым, обратитесь к его описанию в справочнике по JS API. События, возникающие при начале выполнения операции, реализуют типы `CancelableApiEvent` и `BasicApiEvent`.

Получение сервиса клиентского уровня

Сервисы представляют основной API взаимодействия с системой в части функциональности, не связанной с конкретными элементами управления. Они позволяют получить информацию о текущем контексте выполнения, выполнять запросы к серверу, показывать модальные окна, взаимодействовать с разметкой и использовать множество других функций системы.

Доступ к сервисам предоставляет функция `getService` разметки (тип `Layout`). Функция `getService` принимает тип сервиса, который нужно предоставить. Список сервисов клиентского уровня приведён в справочнике по JS API в разделе *Services*.

В Web-клиент версии 17 и выше функция `getService` есть у всех ЭУ. Рекомендуется получать сервис в ближайшем элементе управления.

Следующий код демонстрирует пример получения списка всех операций редактирования, зарегистрированных для карточки, с помощью сервиса `$EditOperationStore`.

```
export function onMyButtonClick(sender: CustomButton, e: IEventArgs) { ①
    let editOperationStore = sender.getService($EditOperationStore); ②

    let editOperations = editOperationStore.getAll(); ③
}
```

① Метод onMyButtonClick обрабатывает событие нажатия кнопки.

② Получаем разметку layout из элемента, вызвавшего событие. С помощью метода getService получаем сервис $EditOperationStore.

③ Вызываем метод getAll сервиса для получения списка всех операций редактирования.

Переопределение сервиса отрисовки элемента "Ход согласования"

Элемент управления **Ход согласования** представляет собой таблицу, в которой отображаются события заданий согласования (выполнение, делегирование и др.).

Программист может переопределить функцию отрисовки всей таблицы и отдельных элементов: заголовков и ячеек определённых столбцов. Для этого нужно передать в поле `stageRenderer` класса элемента управления `Ход согласования` (тип `AgreementHistory`) сервис с интерфейсом `IAgreementHistoryStageRenderer`:

```
interface IAgreementHistoryStageRenderer {
    renderStageTable?(props: IApprovalCycleInfoProps): React.ReactNode;
    renderStageHeaderCells?(): React.ReactNode;
    renderApproverCell?(stageItemProps: IApprovalStageItemRowProps): React.ReactNode;
    renderDecisionCell?(stageItemProps: IApprovalStageItemRowProps): React.ReactNode;
    renderDateCell?(stageItemProps: IApprovalStageItemRowProps): React.ReactNode;
    renderCommentCell?(stageItemProps: IApprovalStageItemRowProps): React.ReactNode;
    renderCorrectionsCell?(stageItemProps: IApprovalStageItemRowProps): React.ReactNode;
}
```

Интерфейс `IAgreementHistoryStageRenderer` определяет функции, обеспечивающие отрисовку определённых элементов таблицы `Ход согласования`:

- `renderStageTable` — отрисовка всей таблицы.
- `renderStageHeaderCells` — отрисовка заголовков таблицы.
- `renderApproverCell` — отрисовка ячеек столбца *Согласующий*.
- `renderDecisionCell` — отрисовка ячеек столбца *Решение*.
- `renderDateCell` — отрисовка ячеек столбца *Дата*.
- `renderCommentCell` — отрисовка ячеек столбца *Комментарий*.
- `renderCorrectionsCell` — отрисовка ячеек столбца *Правки*.

При реализации объекта `IAgreementHistoryStageRenderer` могут быть переопределены все или только требуемые функции интерфейса.

Ниже приведён пример реализации компонента, который переопределяет функцию отрисовки ячеек столбца `Комментарий`: ставит `-`, если комментария не было, и текст комментария — если был. Компонент реализуется в виде стандартного клиентского расширения.

Компонент включает две составляющие:

- Сервис, который будет передан в `stageRenderer`.
- Метод, который будет вызываться при открытии вкладки `Ход согласования` карточки *ДокументУД/Исходящий*, и передавать сервис в `stageRenderer`.

Ниже приведено содержимое файла `EventHandlers.tsx` (стандартное название основного файла клиентского расширения, создаваемого из шаблона).

```
import { IAgreementHistoryStageRenderer } from
"@docsvision/webclient/Approval/$AgreementHistoryStageRenderer";
import { serviceName } from "@docsvision/webclient/System/ServiceUtils";
import { IApprovalStageItemRowProps } from
"@docsvision/webclient/Approval/IApprovalStageItemRowProps";
import { IEventArgs } from "@{Docsvision/webclient/System/IEventArgs";
import { AgreementHistory } from "@docsvision/webclient/Approval/AgreementHistory";
import { Tab } from "@docsvision/webclient/Platform/Tab";
import { $RequestManager } from "@docsvision/webclient/System/$RequestManager";

import * as React from 'react' ①

export function loadRender(sender: Tab, e: IEventArgs) { ②

    if(sender.params.activeTabPage.key == "agreementHistoryTabPage"){ ③

        let agreementHistory = sender.layout.controls.get<AgreementHistory>(
"agreementHistory"); ④

        agreementHistory.params.stageRenderer = sender.layout.getService
($AgreementHistoryRenderService); ⑤
    }
}

export class AgreementHistoryRenderService implements IAgreementHistoryStageRenderer { ⑥
    constructor(private services: $RequestManager) {
    }

    renderCommentCell(stageItemProps: IApprovalStageItemRowProps): React.ReactNode{ ⑦

            if (!stageItemProps.stageItem.hasComment){
                return (
```

```
                <td key="comment"> - </td> ⑧
                )
            }

            return (
            <td key="comment">{stageItemProps.stageItem.comment}</td> ⑨
            )
    }
}

export type $AgreementHistoryRenderService = { renderService:
AgreementHistoryRenderService };
export const $AgreementHistoryRenderService = serviceName((s:
$AgreementHistoryRenderService) => s.renderService); ⑩
```

① Импорт зависимостей.

② Определяется функция, которая будет вызываться при открытии страниц элемента "Вкладки".

③ Следующий код применяется только для страницы с названием `agreementHistoryTabPage`.

④ Получение элемента управления `Ход согласования`.

⑤ Передача сервиса отрисовки.

⑥ Реализация сервиса отрисовки, наследуемого от `IAgreementHistoryStageRenderer`.

⑦ Переопределяется метод отрисовки ячеек столбца "Комментарий".

⑧ Если комментария нет, пишется `-`.

⑨ Если комментарий есть, пишется его текст.

⑩ Экспорт сервиса AgreementHistoryRenderService.



Инструкция по разработке и регистрации клиентских сервисов приведена в пункте Разработка расширения с клиентским сервисом.

В файле `Index.ts` нужно добавить регистрацию сервиса `$AgreementHistoryRenderService`.

```
import * as EventHandlers from "./EventHandlers";
import { $AgreementHistoryRenderService, AgreementHistoryRenderService } from
"./EventHandlers";
import { Service } from "@docsvision/webclient/System/Service";
import { $RequestManager } from "@docsvision/webclient/System/$RequestManager";
import { extensionManager } from "@docsvision/webclient/System/ExtensionManager";
```

```
extensionManager.registerExtension({
    name: "Template web extension",
    version: "6.1",
    globalEventHandlers: [ EventHandlers ],
    layoutServices: [
        Service.fromFactory($AgreementHistoryRenderService, (services: $RequestManager)
=> new AgreementHistoryRenderService(services))
    ]
})
```

После загрузки клиентского расширения на сервер Web-клиента нужно указать обработчик для переключения страниц элемента Вкладки карточки *Документ*:

1. Откройте программу Конструктор Web-разметок и перейдите к разметке просмотра карточки *ДокументУД/Исходящий*.
2. Перейдите к свойствам элемента contentTab.
3. В событии **После переключения активной вкладки** укажите обработчик loadRender.
4. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

   Будет выполнена загрузка скриптов.

Чтобы проверить пример, откройте исходящий документ с согласованием и перейдите на страницу Ход согласования. В столбце *Комментарий* у событий согласования с комментарием будет указан комментарий, у событий без комментария — знак -.



Чтобы получить другие примеры, включая пример получения списка файлов для отображения столбца *Правки*, обратитесь к исходному коду элемента управления Ход согласования в каталоге установки Web-клиента: /lib/docsvision/webclient/Content/App/Approval/Controls/AgreementHistory.

Разработка расширения с локализованными ресурсами

При необходимости использовать локализованные ресурсы в расширении можно их добавить при регистрации расширения в Index.ts.

Чтобы создать расширение, добавляющее новые локализованные ресурсы, выполните приведенную далее инструкцию.

1. Создайте проект клиентского расширения.
2. Создайте в папке `src` файл TypeScript (`.ts`, `.tsx`) со следующим содержимым:

```
import { ILocalizationsMap } from "@docsvision/webclient/System/ILocalizationsMap"; ①

export function getLocalizedResources(): ILocalizationsMap {
    let cultureMap: ILocalizationsMap = {};
    cultureMap["en"] = {
        "Resource_One": "Sample",
        "Resource_Two": "Sample 2"
    };
       cultureMap["ru"] = {
        "Resource_One": "Пример",
        "Resource_Two": "Пример 2"
    };
    return cultureMap;
}
```

① Импортируем необходимые сущности из модуля.

В данном коде выполняется добавление двух ресурсов `Resource_One` и `Resource_Two` для языков русский (`ru`) и английский (`en`).

Функция `getLocalizedResources` (можно использовать другое название) возвращает массив добавляемых локализованных ресурсов, в котором каждый элемент соответствует определённой локали. Например, следующий участок кода отвечает за добавление ресурсов для локали `ru` (русский):

```
cultureMap["ru"] = {
    "Resource_One": "Пример",
    "Resource_Two": "Пример 2"
};
```

3. Измените содержимое файла `Index.ts`.

```
import { getLocalizedResources } from "./LocalizedResource"; ①
import { extensionManager } from "@docsvision/webclient/System/ExtensionManager";
import { ILocalizationsMap } from "@docsvision/webclient/System/ILocalizationsMap";

extensionManager.registerExtension({ ②
```

```
    name: "Client extension with localization",
    version: "1.0",
    getLocalizations(): ILocalizationsMap {
        return getLocalizedResources();
        }
})
```

① Добавьте строку импорта функции getLocalizedResources из добавленного в 2 файла (в примере — LocalizedResource).

② Добавьте в registerExtension функцию getLocalizations, которая должна вызывать импортированную в 1 функцию.

Пример использования ресурсов

```
export function onButtonClick(sender: LayoutControl, args: IEventArgs) {
     const messageWindow - sender.getService($MessageWindow);
     messageWindow.showInfo(resoures.Resource_One);
}
```

Разработка расширения с клиентским сервисом

Данный раздел посвящён созданию новых сервисов клиентского уровня. Сервисы предоставляют публичные методы, которые могут быть вызваны (с помощью функции получения сервиса) из клиентских расширений (скриптов, компонентов элементов управления). Программист может реализовать сервисы различного назначения.



Рекомендуется использовать сервис в одном клиентском расширении. Использование разработанного сервиса в другом расширении сопряжено со сложностями настройки сборки.

Создание расширения

Чтобы создать расширение, добавляющее новые сервисы, выполните приведенную далее инструкцию.

1. Создайте проект клиентского расширения.
2. Создайте в папке `src` файл TypeScript (`.ts`, `.tsx`) со следующим содержимым:

```
import { $RequestManager } from "@docsvision/webclient/System/$RequestManager";
import { urlStore } from "@docsvision/webclient/System/UrlStore";
import { serviceName } from "@docsvision/webclient/System/ServiceUtils";
```

```
export class CardService {
    constructor(private services: $RequestManager) {
    }
    exists(cardId: string): Promise<boolean> { ①
        let url = urlStore.urlResolver.resolveApiUrl("Exists", "CardService"); ②
        url = url + "?cardId=" + cardId;
        return this.services.requestManager.get<boolean>(url);
    }
}

export type $CardService = { cardService: CardService };
export const $CardService = serviceName((s: $CardService) => s.cardService); ③
```

① Функция `exists` проверяет существование карточки.

② Проверка выполняется с помощью серверной функции `Exists` контроллера `CardService`.

③ Объявление сервиса `$CardService`.

> В данном коде реализован сервис `CardService`, содержащий единственную функцию `exists`, возвращающую признак существования карточки в системе Docsvision.
>
> При реализации собственного сервиса, используйте его объявление, по аналогии с приведенным примером, заменив `CardService` на название собственного сервиса.

3. Измените содержимое файла `Index.ts`.

```
import { $CardService, CardService } from "./WebService"; ①
import { extensionManager } from "@docsvision/webclient/System/ExtensionManager";
import { Service } from "@docsvision/webclient/System/Service";
import { $RequestManager } from "@docsvision/webclient/System/$RequestManager";

extensionManager.registerExtension({
    name: "Client extension with service",
    version: "1.0",
    layoutServices: [ ②
        Service.fromFactory($CardService, (services: $RequestManager) => new
CardService(services)) ③
        ]
})
```

① Добавьте строку импорта тип/константы $CardService и класса CardService из добавленного в 2 файла (в примере — WebService).

② Добавьте в registerExtension поле layoutServices, в значении которого должна быть специальная конструкция.

③ Service.fromFactory(**<Тип сервиса>**, (services: $RequestManager) ⇒ new **<Класс реализации сервиса>**(services)).

4. Соберите проект клиентского расширения командой `npm run build`.
5. Скопируйте полученный файл `%BuildDir%/extension.js` на сервер Web-клиента в папку `/lib/docsvision/webclient/Content\/%Каталог Решения%`.



Обратитесь к разделу Создание и публикация клиентского расширения, чтобы получить больше информации, связанной со сборкой проекта.

Получить разработанный сервис в клиентском расширении можно с помощью метода `getService`. Дополнительная информация по использованию функции getService приведена в пункте Получение сервиса клиентского уровня.

```
let cardService = layout.getService($CardService); ①
```

① Получение сервиса $CardService.

Проверка работы расширения

Чтобы проверить работу сервиса CardService:

1. Добавьте в данное расширение обработчик событий, вызывающий функцию `$CardService.exists`.

В следующем коде в метод exists сервиса `$CardService` передается идентификатор карточки, введённый в строку `cardId`. Результат проверки отображается на экране. Реализованный метод `onMyButtonClick` должен быть назначен обработчиком события нажатия **Кнопки**.

```
export async function onMyButtonClick(sender: CustomButton, e: IEventArgs) {

let cardId = sender.layout.controls.get<TextBox>("cardId").value;
```

```
let result = await sender.layout.getService($CardService).exists(cardId);

if (result)
MessageBox.ShowInfo('Карточка существует');
else
MessageBox.ShowInfo('Карточка не существует');
}
```

2. Разработайте серверное расширение, предоставляющее функцию `Exists`. Инструкция по разработке такого расширения приведена в разделе Разработка расширения с методами WebApi.
3. Опубликуйте клиентское и серверное расширения на сервере Web-клиента.
4. Перезапустите сервер Web-клиента.
5. Настройте разметку карточки, указав для любой **Кнопки** обработчик `onMyButtonClick`. Добавьте в разметку *Строку* с названием *cardId*.
6. Откройте карточку с настроенной разметкой. Введите идентификатор существующей карточки в *Строку* и нажмите **Кнопку**. На экране отобразится сообщение: `Карточка существует`.

Настройка страницы авторизации

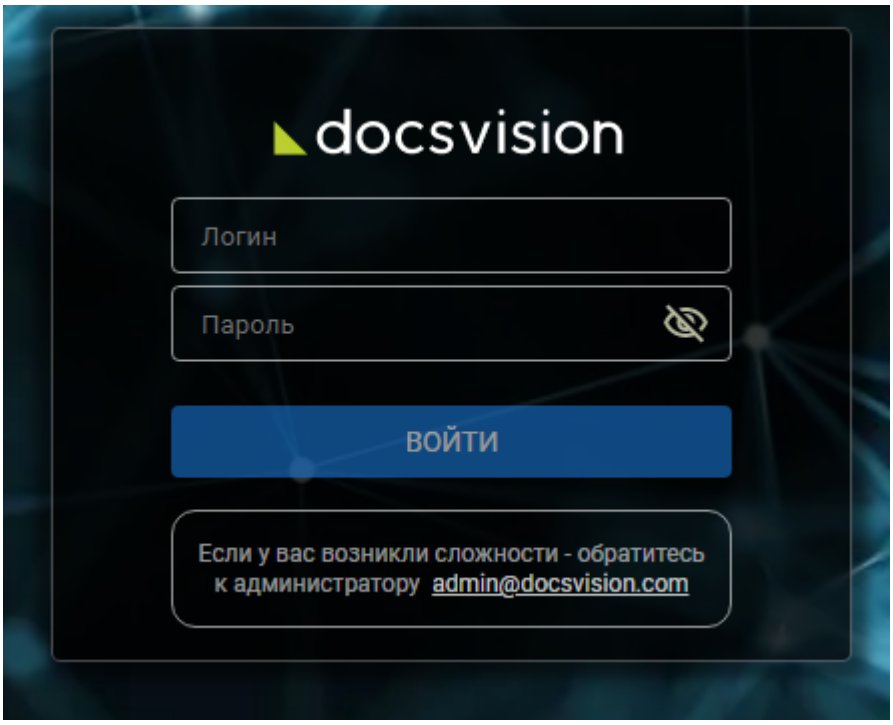


Рисунок 705. Страница входа с указанным адресом электронной почты администратора

Данный раздел описывает расширение, которое содержит пример внедрения

своего кода на страницу входа Web-клиента, а именно:

1. Выполнение кода при определённых событиях — тип `LoginEvents` в модуле `@docsvision/webclient/Account`.
2. Подмена текста на странице — объект `defaultLoginResources` в модуле `@docsvision/webclient/Account`.
3. Замена компонентов страницы — объект `LoginView` в модуле `@docsvision/webclient/Account/LoginContainer`.
4. Добавление стилей на страницу.



Данный раздел документации описывает настройку внешнего вида страницы авторизации. Инструкцию по добавлению расширений авторизации см. в документации по настройке Docsvision.

Чтобы изменить вид страницы входа, разработайте собственные скрипты По инструкции ниже и поместите их в папку `/lib/docsvision/webclient/Content/Account/Scripts`.

Ссылка на пример на GitHub: LoginPage.

Дополнительная информация

В примере используется UMD система модулей.

в результате сборки получается файл с расширением `.js`, который нужно поместить в папку `DocsVision/WebClient/Content/Account/Scripts`, код будет выполнен перед загрузкой приложения и выполнением `ReactDOM.render()`.

Чтобы добавить стили, нужно создать файл или файлы с расширением `.css` и положить в папку `DocsVision/WebClient/Content/Account/Styles`, стили загружаются после основных стилей страницы.

Входной точкой для внедрения кода на страницу является объект `loginExtensionManager` доступный глобально через объект `windows`.

Код расширения будет выполнен перед загрузкой приложения и выполнением `ReactDOM.render()`, но после загрузки библиотек `React` и `ReactDOM`.

Методы `loginExtensionManager`, а также возможные ресурсы, компоненты и события для замены обозначены в виде типов и интерфейсов в модуле `@docsvision/webclient/Account`.

Процесс настройки

Чтобы настроить вид страницы авторизации, необходимо:

1. Скопируйте проект на компьютер для разработки.
2. Откройте консоль в папке `ClientScripts/TypeScript/LoginPage` и выполните команду `npm install`, затем выполнить `npm run build:prod`.
3. Скопируйте каталог `SamplesOutput/Content/Account/Scripts` в каталог `/lib/docsvision/webclient/Content/Account/Scripts`.
4. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Пример кода расширения указан ниже и в файле `Index.ts` в репозитории примеров Web-клиента.

```
import React from 'react';
import { windowWithLoginExtensionManager, ILoginPlugin, ILoginExtensionManager,
LoginEvents } from '@docsvision/webclient/Account';
import { LoginView } from '@docsvision/webclient/Account/LoginContainer';
import { LoginLogic } from '@docsvision/webclient/Account/LoginLogic';
import "index.scss"

const loginExtensionManager: ILoginExtensionManager = (window as
windowWithLoginExtensionManager).loginExtensionManager;
loginExtensionManager.setResources({ Login_AdminMail: 'admin@docsvision.com' }); ①
if (loginExtensionManager.currentLocale === 'ru') {
    loginExtensionManager.setResources({ Login_Greeting: 'Приветствуем,' });
} else {
    loginExtensionManager.setResources({ Login_Greeting: 'Hello,' });
}

const plugin: ILoginPlugin = {
    order: 1,
    eventName: "LogicEventConstructor",
    fn: (data) => {
②
    },
    name: 'myPlugin',
};
const plugin2: ILoginPlugin = {
    order: 2,
    eventName: "EventContainerUseEffect",
    fn: (data) => {
③
    },
    name: 'myPlugin2',
```

```
};

④
loginExtensionManager.addPlugin(plugin);
loginExtensionManager.addPlugins([plugin2]);

⑤
loginExtensionManager.onSetupView = function (view: LoginView): LoginView {
⑥
    view.logo = React.memo((logic: LoginLogic): React.ReactElement => {
        return <img src=
'https://www.google.com/images/branding/googlelogo/2x/googlelogo_color_92x30dp.png'
className="my-custom-logo" />;
    });

⑦
    view.authBoxBefore = (logic: LoginLogic): React.ReactElement => null;
    view.authBoxAfter = (logic: LoginLogic): React.ReactElement => null;
    view.formBefore = (logic: LoginLogic): React.ReactElement => null;
    view.formAfter = (logic: LoginLogic): React.ReactElement => <a href="#" className=
"my-custom-element" >Забыли пароль?</a>;
    view.providersBefore = (logic: LoginLogic): React.ReactElement => null;
    view.providersAfter = (logic: LoginLogic): React.ReactElement => null;
    return view;
};
```

① Замените текст.

② Добавьте код выполнения некоторого действия.

③ Добавьте код выполнения ещё одного действия.

④ Добавьте события.

⑤ Измените компоненты представления.

⑥ Замените логотип.

⑦ Точки расширения.

Проверка примера

1. Перейти на страницу логина в браузере `http://company.com:5004/account/login`.
2. В результате на странице входа будут изменены логотип и приветствие, а также добавлена почта администратора.

Расширение серверных вызовов

Пример серверного расширения для взаимодействия с панелью настроек

таблицы. Пример содержит клиентский скрипт позволяющий добавить элемент в список контекстного меню с возможностью открыть карточку в модальном окне или скрыть существующий элемент.

Ссылка на пример на GitHub: ExtendServerQueryData.

```
extensionManager.registerExtension({ ①
    name: "ExtendServerQueryData",
    version: "6.1",
    globalEventHandlers: [ExtendServerQueryData]
});
```

① Регистрация расширения позволяет корректно установить все обработчики событий, сервисы и прочие сущности web-приложения.

На следующем шаге необходимо клонировать и заменить собственный сервис контроллера. Ниже приведён код обработчика события, в котором показано, как это сделать:

```
export function extendServerQueryData(sender: Layout) {
    const services = cloneServices(sender.params.services);
    replaceService(services, { tasksTreeController: new TasksTreeControllerCustom(sender
.params.services)});
    sender.params.services = services;
}
```

Затем на основе метода базовой реализации стандартного класса сформировать новые данные:

```
export class TasksTreeControllerCustom extends GenControllers.TasksTreeController {
    constructor(services) {
        super(services);
    }
    async getTasksTable(request: GenModels.TasksTableLoadOptions, options?:
RequestOptions) {
        var data = await super.getTasksTable(request, options);
        if (data && data.rows) {
            data.rows.forEach(row => {
                if (row.indicators.includes(GenModels.TasksIndicators.Overdue)) {
                    row.indicators.push(GenModels.TasksIndicators.HighPriority)
                }
            })
        };
```

```
        return data;
    }
}
```

Проверка примера

1. Запустите конструктор разметок.
2. Импортируйте решение из файла `ExtendServerQueryData.xml`.
3. Сделайте решение `ExtendServerQueryData` активным для карточки типа *Документ* вида *ДокументУД/Исходящий*.
4. В примере используется стандартная разметка просмотра исходящего документа. На событие **`Перед открытием карточки`** ЭУ `Layout` разметки назначается обработчик `extendServerQueryData`.
5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
6. Откройте исходящий документ, создайте подчиненное задание с просроченной датой исполнения.
7. Убедитесь, что в таблице исполнения просроченные задания автоматически помечаются индикатором высокого приоритета.

Управление диалогом подписания через API

Управлять диалогом подписания можно при помощи специального API.

*Свойства интерфейса `ISignatureDialogProps` в событии **`Перед открытием диалога подписи`** (`beforeSignatureDialogShown`) позволяют:*

- Скрывать выбор метки.
- Скрывать выбор сертификата.
- Отключать простую подпись.
- Скрывать кнопку настройки способов подписания.
- Скрывать сам диалог: диалог открывается, но оказывается невидим на экране для пользователя.
- Добавлять содержимое в нижней и верхней части диалога.
- Обрабатывать доступные способы подписания: фильтровать, модифицировать, добавлять свои.
- Управлять отрисовкой кнопок: добавлять свои, заменять стандартные.

*Управление диалогом через интерфейс `IDigitalSignatureDialogManagement` события **`После открытия`***

диалога подписи *(`afterSignatureDialogShown`) позволяет получать информацию о загруженных метках и способах подписания:*

- Программно выбирать метку и способ подписания
- Изменять видимость диалога для пользователя
- Динамически менять свойства диалога (см. п. 3)
- Программно подтверждать или отменять операцию подписания
- Также событие `beforeSignatureDialogShown` позволяет подменять логику формирования подписи и прикрепления к документу. Это можно использовать для подключения сторонних провайдеров подписания.

Входная точка — сервис `$DigitalSignature`. Для настройки диалога следует использовать события `beforeSignatureDialogShown` и `afterSignatureDialogShown` указанного сервиса. В первом событии можно настроить свойства диалога, во втором получить доступ к управлению диалогом. В событиях доступна информация о карточке, из которой осуществляется подпись: вид, тип, состояние и другая информация.

Следующий пример кода демонстрирует автоматический выбор сертификата для документов вида *Исходящий* и подписание без показа диалога:

```
export function initExtension(app) {
        app.digitalSignature.beforeSignatureDialogShown.watch(args => { ①
            if (args.cardInfo.kindId == '8e40f327-9517-4a43-998d-bf2bd619588d') {
                 args.dialogProps.showHidden = true;
            }
        });
        app.digitalSignature.afterSignatureDialogShown.watch(async (args) => { ②
            let methods = await args.management.getSignatureMethods();
            let outgoingMethod = methods.find(x => x?.certificateInfo?.friendlyName
.includes('для исходящего'));
            if (outgoingMethod) {
                args.management.selectSignatureMethod(outgoingMethod);
                args.management.accept();
            } else {
                const props = args.management.getDialogProps();
                props.footer = "Не удалось найти сертификат для исходящих документов.";
                args.management.update(props);
                args.management.setDialogVisible(true);
            }
        });
}
```

① По умолчанию скрываем диалог для документов вида *Исходящий*.

② Автоматически выбираем сертификат и нажимаем подписать.

Можно вставить в метод `init` при вызове `registerExtension`, либо просто скопировать в консоль браузера и вызвать `initExtension(WebClient.app)`.

Пример интеграции со сторонним облаком:

```
export function initExtension(app) {
    app.digitalSignature.beforeSignatureDialogShown.watch(args => {
       args.options.onCreateSignature = (options) => { return signByCustomCloud(options);
}
    });
}
```

Свойства диалога объекта `dialogProps` в событии `beforeSignatureDialogShown`:

```
    hideLabelSelection?: boolean; ①
    hideSignatureMethodsSetup?: boolean; ②
    hideSimpleSign?: boolean; ③
    hideSignatureMethodsSelect?: boolean; ④

    showHidden?: boolean; ⑤
    footer?: React.ReactNode; ⑥
    header?: React.ReactNode; ⑦

    onSign?: (options: IDigitalSignOptions) => Promise<void>; ⑧
    onCancel?: () => void; ⑨
    onPrepareSignatureMethods?: (methods: GenModels.SignatureMethod[]) => GenModels
.SignatureMethod[]; ⑩
    onRenderButtons?: (props: ISignatureDialogButtonsProps, defaultButtons: JSX.
Element[]) => JSX.Element[]; ⑪
```

① Не отображать выбор метки.

② Не отображать кнопку настройки вариантов подписания.

③ Скрыть метод простой подписи.

④ Не отображать раскрывающийся список выбора способа подписания. В таком случае будет выбран первый вариант по умолчанию.

⑤ Чтобы не отображать диалог на экране, используйте API `@link IDigitalSignatureDialogManagement`.

⑥ Сообщение, показываемое в нижней части диалога

⑦ Сообщение, показываемое в верхней части диалога

⑧ Вызывается при подписи.

⑨ Вызывается при отмене подписи.

⑩ Вызывается для обработки отображаемого списка доступных способов подписания.

⑪ Вызывается при отрисовке кнопок диалога.

Свойства объекта `management` *в событии* `afterSignatureDialogShown`*:*

```
    close(reason?: unknown): Promise<void>; ①
    accept?(data?: IDigitalSignOptions): Promise<void>; ②
    getSignatureMethods(): Promise<GenModels.SignatureMethod[]>; ③
    getLabels(): Promise<ISignatureLabel[]>; ④
    selectSignatureMethod(method: GenModels.SignatureMethod): void; ⑤
    selectLabel(label: ISignatureLabel): void; ⑥
    getDialogVisible(): boolean; ⑦
    setDialogVisible(visible: boolean): void; ⑧
    getDialogProps(): ISignatureDialogProps; ⑨
    update(props?: ISignatureDialogProps): void; ⑩
```

① Закрыть диалог без сохранения данных.

② Применить изменения и закрыть диалог. Параметр не обязателен.

③ Возвращает список доступных сертификатов и других способов подписания.

④ Возвращает список доступных меток подписи.

⑤ Выбирает в раскрывающемся списке указанный способ подписания.

⑥ Выбирает в раскрывающемся списке указанную метку подписи.

⑦ Возвращает значение свойства видимости диалога для пользователя.

⑧ Устанавливает видимость диалога для пользователя.

⑨ Возвращает текущие свойства диалога подписания.

⑩ Обновляет диалог с указанными свойствами. См. `@link getDialogProps`.

Примеры тестирования

Скрытие раскрывающегося списка для выбора метки в диалоге подписания:

```
WebClient.app.digitalSignature.beforeSignatureDialogShown.watch(args => {

    args.dialogProps.hideLabelSelection = true;
```

```
});
```

Отображение сообщения в нижней части диалога подписания:

```
WebClient.app.digitalSignature.afterSignatureDialogShown.watch(async (args) => {

    const props = args.management.getDialogProps();

    props.footer = "Добрый день!";

    args.management.update(props);

});
```

Автоматический выбор метки для карточек определенного вида:

```
WebClient.app.digitalSignature.afterSignatureDialogShown.watch(async (args) => {

    let labels = await args.management.getLabels();

    if (args.cardInfo.kindId == '8e40f327-9517-4a43-998d-bf2bd619588d') {

        args.management.selectLabel(labels[1]);

    }

});
```

Реализация собственной логики формирования подписи:

```
WebClient.app.digitalSignature.beforeSignatureDialogShown.watch(args => {

       args.options.onCreateSignature = (options) => { return alert('custom sign create
'); }

});
```

Работа со строкой данных элемента управления "Список файлов" (fileList)

Пример взаимодействия со строками элемента управления Список файлов (`fileList`). Проект представляет собой клиентское расширение. Расширение добавляет к строке элемента управления красную обводку при скачивании файла.

Основной код расширения содержит скрипт `fileListHighlightRow` в качестве

обработчика события **После загрузки версии файла** элемента управления `fileList` в конструкторе разметок.

EventHandler.ts

```
export async function fileListHighlightRow(sender: FileListControl, event:
IFileVersionDownloadingArgs) {
    const fileItem = event.fileItem;
    const customClass = "file-list-row_red-border";

    if (!event.fileItem?.cssClasses?.includes(customClass)) {
        sender.toggleRowClass(fileItem, customClass);
    }
}
```

Данный скрипт представляет пример получения объекта `fileItem`, представляющего информацию о файле из аргументов события. Выполняется исследование его свойства, после чего вызывается метод `toggleRowClass` ЭУ `FileListControl` для переключения стилей файла.

Index.ts

```
extensionManager.registerExtension({ ①
    name: "FileListHighlightRow",
    version: "1.0",
    globalEventHandlers: [ EventHandler ]
})
```

① Регистрация расширения позволяет корректно установить все обработчики событий, сервисы и прочие сущности web-приложения.

Проверка примера

1. Откройте папку с элементом управления и добавленным обработчиком события.
2. Скачайте версию файла.

Настройка страницы входа

Данный раздел содержит описание примера кастомизации страницы входа в Web-клиент.

В этом примере демонстрируется возможность внедрения своего кода на страницу входа в Docsvision Web-клиент, а именно:

- Выполнение кода при определённых событиях(тип `LoginEvents` в модуле `@docsvision/webclient/Account`).
- Подмена текста на страницы(объект `defaultLoginResources` в модуле `@docsvision/webclient/Account`).
- Замена компонентов страницы(объект `LoginView` в модуле `@docsvision/webclient/Account/LoginContainer`).
- Добавление стилей на страницу.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке `ClientScripts\TypeScript\LoginPage` и выполните команду `npm install`, затем `npm run build:prod`.
2. Скопируйте каталог `SamplesOutput\Content\Account\Scripts` в каталог `Путь к сайту Web-клиента\Content\Account\Scripts`.
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Дополнительная информация

- Рекомендуется осуществлять сборку в виде umd модуля.
- Для корректной работы расширения нужно создать файл или файлы с расширением `.js` и положить в папку `Путь к сайту Web-клиента\Content\Account\Scripts`, код будет выполнен перед загрузкой приложения и выполнением `ReactDOM.render()`.
- Чтобы добавить стили, нужно создать файл или файлы с расширением `.css` и положить в папку `Путь к сайту Web-клиента\Content\Account\Styles`, стили загружаются после основных стилей страницы.
- Входной точкой для внедрения кода на страницу является объект `loginExtensionManager` доступный глобально через объект `windows`.
- Код расширения будет выполнен перед загрузкой приложения и выполнением `ReactDOM.render()`, но после загрузки библиотек React и ReactDOM.
- Методы `loginExtensionManager`, а также возможные ресурсы, компоненты и события для замены обозначены в виде типов и интерфейсов в модуле `@docsvision/webclient/Account`.

Блок сообщений

Данный раздел содержит описание примера последовательного вывода нескольких информационных сообщений. Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт с функцией `showMessages` реализующей последовательный вывод `MessageBox`.

Ссылка на пример на GitHub: MessageBox.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке `MessageBoxWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
2. Скопируйте каталог `SamplesOutput\Content\Modules\MessageBoxWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок
2. Импортируйте разметку из файла `MessageBoxLayout.xml` без условий использования. При этом будет добавлена разметка в тип карточки документ, вид *ДокументУД\Исходящий*
3. Задайте для разметки условия использования, поменяв также порядок разметок в условиях использования, чтобы разметка стала первой разметкой для редактирования
4. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
5. Откройте новый документ *ДокументУД\Исходящий* и нажмите кнопку **Редактировать**. При этом должна открыться разметка с кнопкой вызова сообщений.
6. При нажатии на кнопку `ShowMessages` последовательно появляются диалоговые окна: информационное сообщение с текстом `Это информационное сообщение`, предупреждающее сообщение с текстом `Это предупреждающее сообщение`, сообщение об ошибке с текстом `Это сообщение об ошибке`, диалог подтверждения с текстом `Выберите кнопку`. В случае нажатия той или иной кнопки выводится

сообщение.

Модальное окно

Данный раздел содержит описание примера работы с модальным окном. Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт с функцией `showModalDialog` реализующей отображение `ModalDialog`. `ModalDialogComponent` — React компонент, отвечающий за представление модельного окна и его событий.

Ссылка на пример на GitHub: ModalDialog.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке `ModalDialogWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
2. Скопируйте каталог `SamplesOutput\Content\Modules\ModalDialogWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок
2. Импортируйте разметку из файла `ModalDialogLayout.xml` без условий использования. При этом будет добавлена разметка в тип карточки документ, вид *ДокументУД\Исходящий*
3. Задайте для разметки условия использования, поменяв также порядок разметок в условиях использования, чтобы разметка стала первой разметкой для редактирования
4. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
5. Откройте новый документ *ДокументУД\Исходящий* и нажмите кнопку **Редактировать**. При этом должна открыться разметка с кнопкой вызова модального окна и текстовым полем.
6. При нажатии на кнопку **ShowModalDialog** отображается модальное окно, в котором присутствует текстовое поле для ввода. После ввода сообщения и

нажатия на кнопку **ОК**, значение из этого поля копируется в текстовое поле на разметке и сохраняется в качестве имени документа.

Боковая панель

Данный раздел содержит описание примера работы с боковой панелью. Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт с функцией `showModalSidebar` реализующей отображение `ModalSidebar`. `ModalSidebarComponent` — React компонент, отвечающий за представление боковой панели и её событий.

Ссылка на пример на GitHub: ModalSidebar.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке `ModalSidebarWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
2. Скопируйте каталог `SamplesOutput\Content\Modules\ModalSidebarWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок
2. Импортируйте разметку из файла `ModalSidebarLayout.xml` без условий использования. При этом будет добавлена разметка в тип карточки документ, вид *ДокументУД\Исходящий*.
3. Задайте для разметки условия использования, поменяв также порядок разметок в условиях использования, чтобы разметка стала первой разметкой для редактирования.
4. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
5. Откройте новый документ *ДокументУД\Исходящий* и нажмите кнопку **Редактировать**. При этом должна открыться разметка с кнопкой вызова боковой панели и текстовым полем.

6. При нажатии на кнопку **ShowModalSidebar** отображается боковая панель, в котором присутствует текстовое поле для ввода. После ввода сообщения и нажатия на кнопку **ОК**, значение из этого поля копируется в элемент управления `textBox1` разметки.

Отображение незаполненных обязательных полей

Данный раздел содержит описание примера отображения незаполненных обязательных полей. Проект-расширение клиентской части Web-клиент. Расширение отображает окно со списком незаполненных обязательных полей при создании или сохранении карточки.

Ссылка на пример на GitHub: ShowRequiredFields.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке `ShowRequiredFields` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
2. Скопируйте каталог `SamplesOutput\Content\Modules\ShowRequiredFields` в каталог `Путь к сайту Web-клиента\Content\Modules`.
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
4. Добавьте скрипт `showRequiredFields` в качестве обработчика события **`Перед сохранением карточки`** `root` для разметок создания и/или редактирования в конструкторе разметок.

Проверка примера

1. Создайте карточку с обязательными полями и добавленным обработчиком события.
2. Не заполняя обязательные поля, нажмите на кнопку создать.

Проверка данных ЭУ

Данный раздел содержит описание примера проверки, изменения и отмены изменения состояния карточки. Подписаться на событие изменения состояния, провести проверку какого-нибудь поля и случае, если проверка не прошла, выдать предупреждающее сообщение и состояние не менять.

Ссылка на пример на GitHub: ValidateOnStateChanging.

Проект-расширение клиентской части Web-клиент. Содержит клиентский скрипт с функцией-обработчиком события `validateOnStateChanging`, проверку данных ЭУ при смене состояния.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке `ValidateOnStateChangingWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
2. Скопируйте каталог `SamplesOutput\Content\Modules\ValidateOnStateChangingWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Импортируйте решение из файла `ValidateOnStateChangingLayout.xml`.
3. Сделайте решение `ValidateOnStateChangingLayout` активным для карточки типа Документ вида *ДокументУД/Исходящий*.
4. На разметке расположен ЭУ `Текст` и кнопки операций (Start approving и др.) по изменению состояния карточки.
5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
6. Откройте новый документ *ДокументУД\Исходящий* и нажать кнопку редактировать. При этом должна открыться разметка с ЭУ `Текст` и кнопками изменения состояния. Часть кнопок будет неактивна, это зависит от настроек ролевой модели.
7. Если в ЭУ `Текст` нет текста или длина текста превышает 100 символов, то при нажатии на кнопку изменения состояния возникает сообщение об ошибке и состояние не изменяется.
8. В случае, если ЭУ `Текст` проходит валидацию, то состояние карточки меняется. Например при выполнении операции `Start approving` с состояния

`Drafting` на состояние `Is approving`.

9. При этом кнопки перехода исчезают, т.к. в состоянии `Is approving` операции `Start approving` и др. не доступны.



Данный пример демонстрирует только доступ к элементу разметки. При реализации расширений рекомендуется использовать экспорт разметок в виде Решения.

Расширения нового грида

Руководство по разработке решений Web-клиента

Расширения нового грида

Описание примеров разработки расширений нового грида.

- [WebClient:client-grid-context]
- [WebClient:client-grid-group-by-icon]
- [WebClient:client-grid-toolbar-button]
- Больше примеров категории в навигационном меню слева…

Расширить контекстное меню

Грид позволяет расширить функциональность контекстного меню при помощи расширений. Данный раздел описывает пример добавления нового пункта контекстного меню. Новый пункт контекстного меню позволяет открыть карточку в модальном окне или скрыть существующий элемент.

Ссылка на пример на GitHub: GridContextMenuItem.

Начальный код расширения должен импортировать прямо или косвенно все остальные строки, чтобы затем объединить код в один пакет.

```
extensionManager.registerExtension({ ①
    name: "GridContextMenuItem",
    version: "6.1",
    initialize() {
        app.folderPluginProvider.addFactory(new GridContextMenuItemPluginFactory());
    },
```

```
})
```

① Регистрация расширения позволяет корректно установить все обработчики событий, сервисы и прочие сущности web-приложения.

Добавить новый пункт контекстного меню

Расширение добавляет пункт контекстного меню, который открывает карточку в модальном окне.

```
export const AddContextMenuItemProvider: TablePlugins.ServiceProvider<$ContextMenuOptions
& $FolderGrid & $LayoutCardController & $DeviceType & $LayoutManager> = {
    name: "AddContextMenuItemProvider",
    description: "Добавляет пункт контекстного меню, который открывает карточку в
модальном окне",
    feature: GridContextMenuItemFeature,
    composition: TableCompositionNames.Root,
    addServices: (composition) => {
        const { contextMenuOptions, folderGrid } = composition.services;

        if (contextMenuOptions) {

            const isAdded = contextMenuOptions.isAddedProvider
(CONTEXT_MENU_ITEM_CUSTOM_PROVIDER_ID, TableCompositionNames.TableCell); ①

            if (!isAdded) {
                const provider = {
                    id: CONTEXT_MENU_ITEM_CUSTOM_PROVIDER_ID, ②
                    composition: TableCompositionNames.TableCell, ③
                    getItemsToAdd: () => ([ ④
                        {
                            id: "CONTEXT_MENU_CUSTOM_ITEM", ⑤
                            name: "Предпросмотр карточки", ⑥
                            action: (context: TableCompositions.Cell) => showCard(
context, composition.services), ⑦
                            blockId: folderGrid?.getContextMenuBlockNames().
linkOperation, ⑧
                            order: 0, ⑨
                        }
                    ])
                } as IDataItemProvider;

                contextMenuOptions.addItemProvider(provider);
            }
        }
    }
```

```
};
```

① Поскольку метод `addServices` вызывается при каждой прорисовке, проверим не был ли добавлен ранее наш провайдер.

② Уникальный ID провайдера, содержащего элементы списка контекстного меню.

③ Композиция, где будет вызвано контекстное меню.

④ Метод, который возвращает массив элементов контекстного меню.

⑤ Уникальный ID элемента контекстного меню.

В данном примере с помощью ID осуществляется доступ к скрытию элемента контекстного меню.

⑥ Отображаемое название элемента контекстного меню.

⑦ Метод, который будет вызван после клика на элемент контекстного меню.

⑧ Указание на расположение элемента в блоке элементов контекстного меню.

⑨ Порядковый номер расположения элемента в блоке элементов контекстного меню.

Скрыть пункт контекстного меню

Следующий пример скрывает пункт "Открыть" в контекстном меню грида.

```
export const HideContextMenuItemProvider: TablePlugins.ServiceProvider
<$ContextMenuOptions> = {
    name: "HideContextMenuItemProvider",
    description: "Скрывает пункт «Открыть» в контекстном меню",
    feature: GridContextMenuItemFeature,
    composition: TableCompositionNames.Root,
    addServices: (composition) => {
        const { contextMenuOptions } = composition.services;

        if (contextMenuOptions) {

            const isAdded = contextMenuOptions.isAddedFilter
(CONTEXT_MENU_ITEM_CUSTOM_FILTER_ID, TableCompositionNames.TableCell); ①

            if (!isAdded) {
                const filter = {
                    id: CONTEXT_MENU_ITEM_CUSTOM_FILTER_ID, ②
                    composition: TableCompositionNames.TableCell, ③
                    getItemsToHide: () => [CONTEXT_MENU_FOLDER_OPEN_ID] ④
```

```
                } as IDataItemFilter;

                contextMenuOptions.addItemFilter(filter);
            }
        }
    }
};
```

① Поскольку метод addServices вызывается при каждой прорисовке, проверим не был ли добавлен ранее наш фильтр.

② Уникальный ID фильтра.

③ Композиция, где будет вызвано контекстное меню.

④ Метод, который возвращает массив ID элементов контекстного меню.

Отображение модального окна

Данный пример демонстрирует создание и отображение модального окна.

```
export async function showCard(context: TableCompositions.Cell, services:
$ContextMenuOptions & $FolderGrid & $LayoutCardController & $DeviceType & $LayoutManager)
{
    const modalHost = new CommonModalDialogHost("grid-context-menu__card-layout",
services); ①

②

    const dialogProps = modalHost.dialogProps as CommonModalDialogProps;

    dialogProps.header = "Предпросмотр"; ③

    dialogProps.maxWidth = "80vw"; ④

    dialogProps.maxHeight = "80vh"; ⑤

    dialogProps.maximizeButtonEnabled = true;
    dialogProps.onMaximize = () => {
        dialogProps.maximized = !dialogProps.maximized;
        modalHost.forceUpdate(); ⑥
    }

    modalHost.onCancelCallback = () => services.layoutManager.destroy
(GRID_CONTEXT_MENU_SHOW_CARD_LOCATION_NAME); ⑦

    const renderLoading = (state: LoadingState) => <LoadingViewWithText state={state}
className="padding-20 margin-30" />; ⑧
```

```
    const loadLayout = async (): Promise<GenModels.LayoutViewModel> => { ⑨
        const model = await services.layoutCardController.view(context.data.row.entityId,
PlatformModeConditionTypes.VIEW);
        return model;
    }

    const dialogServices = addServices(services, modalHost.service, true); ⑩

    modalHost.renderCallback = () => (
        <LocationContainerControl
            async={true}
            renderLoadingState={renderLoading}
            customLayoutLoader={loadLayout} ⑪
            locationName={GRID_CONTEXT_MENU_SHOW_CARD_LOCATION_NAME}
            services={dialogServices}
            name={GRID_CONTEXT_MENU_SHOW_CARD_LOCATION_NAME} />
    );

    await modalHost.showDialog(); ⑫
}
```

① Модальное окно.

② Конфигурируем модальное окно.

③ Заголовок модального окна.

④ Максимальная ширина.

⑤ Максимальная высота.

⑥ Изменение отображения модального окна (свернуть / развернуть).

⑦ Обработчик закрытия модального окна.

⑧ Отображение текста загрузки разметки.

⑨ Получение разметки карточки в режиме просмотра.

⑩ Сервисы модального окна.

⑪ Тело модального окна.

⑫ Показываем модальное окно.

Зарегистрируйте плагин в `GridContextMenuItemPluginFactory`:

```
export class GridContextMenuItemPluginFactory implements IFolderPluginFactory {
    id: string = "GridContextMenuItemPluginFactory";
```

```
    getTablePlugins(): ITablePlugins[] {
        return [
            GridContextMenuItemPlugins
        ]
    }
}
```

Проверка примера

1. Откройте папку с таблицей и вызовите контекстное меню у произвольной карточки.
2. Кликните на пункт "Предпросмотр карточки" в контекстном меню.
3. Убедитесь, что открылось модальное окно "Предпросмотр", в котором отображается разметка просмотра выбранной карточки.

Применить группировку кнопкой в ячейке

Данный раздел описывает расширение, добавляющее элемент в панель инструментов грида. Элемент будет отображаться в ячейках таблицы при наведении на них курсора мыши. Нажатие на элемент будет выполнять группировку по столбцу.

Ссылка на пример на GitHub: GridHoverPanelGrouping.

```
extensionManager.registerExtension({
    name: "HoverPanelGroping",
    version: "6.1",
    initialize() {
        app.setService($HoverPanel, new CustomHoverPanelService()); ①
    },
})
```

① Заменяем реализацию стандартного сервиса `$HoverPanel` на `CustomHoverPanelService`.

Следующим шагом добавляем в ячейку таблицы элемент, который будет выполнять функцию группировки:

```
export const HoverPanelGroupingByColumn: HeaderHoverPanelPlugin<$Grouping & $SidePanel> =
{
    name: "HoverPanelGroupingByColumn",
    description: `
        Добавляет элемент в панель при наведении, который находится в ячейке шапки
```

```
таблицы.
        При клике на элемент, происходит группировка по столбцу.
    `,
    feature: CustomHoverPanelFeature,
    composition: HeaderHoverPanelName,
    placement: PluginPlacement.AfterContent,
    component: (props) => {
①
        const { data: { column }, services: { grouping, sidePanel } } = props.
composition;



②
        const groupingPath = useStore(grouping.$groupingPath);

        const groupByColumn = () => {
③
            if (grouping.canAddColumnGrouping(column.id) && !sidePanel.$isOpen.
getState()) {
                sidePanel.toggle();
            }

            grouping.addColumnGrouping(column.id); ④
        };

        const isGrouped = Boolean(groupingPath.find(path => path.columnId == column.id));
⑤

        if (grouping.$groupingAvailable.getState() && (isGrouped || !grouping
.canAddColumnGrouping(column.id))) {
            return <></>; ⑥
        }

        return (
            <Icon
                iconClass="dv-ico context-menu-group hover-panel-custom-grouping-by-
column"
                onClick={groupByColumn}
                dv-tooltip="Сгруппировать по столбцу" /> ⑦
        );
    }
};
```

① Деструктуризируем объект `props`, получая все необходимые данные для реализации группировки по столбцу

② С помощью хука useStore получаем данные из хранилища `$groupingPath`.

Хук предоставляет возможность перерисовывать компонент при изменении состояния хранилища `$groupingPath`.

Так как мы следим за состоянием хранилища и перерисовываем компонент, мы можем учитывать следующую логику:

- Если столбец уже сгруппирован, то не показывать иконку
- Если столбец не сгруппирован, то показывать иконку

③ Проверяем доступна ли колонка для группировки. Если колонка доступна и боковая панель закрыта, открываем боковую панель с группировкой.

④ Группируем по колонке.

⑤ Проверяем сценарий, что группировка по колонке уже применена.

⑥ Если группировка доступна и уже применена или недоступна для этой колонки, то ничего не отрисовываем.

⑦ Если группировка доступна, рисуем иконку. Группировка будет применена при нажатии на иконку.

```
export class CustomHoverPanelService implements IHoverPanelService { ①


    getPlugins(defaultPlugins: HeaderHoverPanelPlugin<any>[]): HeaderHoverPanelPlugin<
any>[] { ②

③

        defaultPlugins.push(HoverPanelGroupingByColumn); ④

        return defaultPlugins;
    }
}
```

① Создаём сервис алогичный `IHoverPanelService`..

② При инициализации плагинов для `HoverPanel`, будет вызван этот метод.

Внутри него мы можем убрать, добавить интересующие нас плагины для панели при наведении и вернуть новый массив плагинов.

③ К примеру:

- `defaultPlugins = defaultPlugins.filter(plugin => plugin.name !== target)`

- `defaultPlugins.push(ownPlugin)`

④ Добавляем плагин для группировки по колонке.

Проверка примера

1. Откройте папку с таблицей.
2. Наведите курсор на ячейку в столбце таблицы, которая доступна для группировки.
3. После нажатия на иконку "Сгруппировать по столбцу", должна открыться боковая панель, если она была закрыта, и добавиться колонка в группируемых элементах.

Отобразить количество непрочитанных карточек

Расширение в данном примере предназначено для добавления кнопки на панель инструментов грида. Расширение получает список карточек и по нажатию на кнопку отображает общее количество карточек, а также количество непрочитанных.

Ссылка на пример на GitHub: GridToolbarButton.

```
extensionManager.registerExtension({ ①
    name: "GridToolbarButton",
    version: "6.1",
    initialize() {
        app.folderPluginProvider.addFactory(new ToolbarButtonPluginFactory());
    },
})
```

① Регистрация расширения позволяет корректно установить все обработчики событий, сервисы и прочие сущности web-приложения.

Следующим шагом необходимо получить количество карточек:

```
export function ToolbarButtonMessageContent(props: IToolbarButtonMessageContent) {
    const { data, services } = props;

    useStore(services.tableData.$data); ①

    const loadedRows = data.rows.filter(x => getRowLoaded(x).loaded).length; ②
    const unreadRows = data.rows.filter(x => getUnreadRow(x).isUnread).length; ③

    return (
        <>
```

```
            <LabeledText label={"На текущий момент загружено карточек"} text={loadedRows}
/>
            <LabeledText label={"Из них не прочитано"} text={unreadRows} />
        </>
    )
}
```

① Обновляем компонент при изменении хранилища `$data` из `services.tableData`.

② Количество загруженных карточек.

③ Количество непрочитанных карточек.

Затем нужно получить плагины грида:

```
export class ToolbarButtonPluginFactory implements IFolderPluginFactory {
    id: string = "ToolbarButtonPluginFactory";

    getTablePlugins(): ITablePlugins[] { ①
        return [ ToolbarButtonPlugins ];
    }
}
```

① Возвращаем плагины грида.

Добавляем кнопку, отображающую информацию в модальном окне при нажатии:

```
export const ToolbarButtonButtonPlugin: TablePlugins.Toolbar.Component<$MessageWindow &
$TableData> = {
    name: "ToolbarButtonButtonPlugin",
    description: "Отображает кнопку в панели инструментов таблицы",
    feature: ToolbarButtonFeature,
    composition: TableCompositionNames.Toolbar,
    order: CompositionPluginOrder.Priority,
    component: (props) => {
        const { data, services } = props.composition;

        const showInfo = () => {
            services.messageWindow.showInfo(() => <ToolbarButtonMessageContent data={
data} services={services} />); ①
        };

        return <IconButton iconClassName="dv-ico ico-modal-info" onClick={showInfo} />;
②
    }
};
```

① Вызывается модальное окно с информацией.

② Кнопка, при нажатии на которую открывается модальное окно с информацией.

Проверка примера

1. Откройте папку с таблицей и нажмите на информационную кнопку в панели настроек таблицы.
2. Убедитесь, что открылось модальное окно с информацией. В окне должно отображаться количество загруженных карточек на данный момент и количество непрочитанных из них.

Изменить цвет строки по кнопке

В данном примере создаётся новая колонка таблицы грида. В новую колонку добавляется элемент, нажатие на который открывает диалог выбора цвета. При изменении цвета в диалоге, будет изменяться фон строки, в которой находится ячейка.

```
extensionManager.registerExtension({ ①
    name: "BackgroundRows",
    version: "6.1",
    initialize() {
        app.folderPluginProvider.addFactory(new BackgroundRowsPluginFactory());
    },
})
```

① Регистрация расширения позволяет корректно установить все обработчики событий, сервисы и прочие сущности web-приложения.

Создание кнопки, открывающей диалог выбора цвета:

```
export interface IBackgroundRowsButton extends PropsWithChildren<ICompositionPluginProps
<"TableCell", ICellData, $BackgroundRows>> {
}

export function BackgroundRowsButton (props: IBackgroundRowsButton) { ①
    const { services: { backgroundRows }, data: { row } } = props.composition;

    const [ color, setColor ] = useState( backgroundRows.$rows.getState()[row.entityId]
?? "#ffffff" ); ②

    const changeColor = (ev: ChangeEvent<HTMLInputElement>) => {
        backgroundRows.setBackground({ id: row.entityId, color: ev.target.value }); ③
```

```
        setColor(ev.target.value); ④
    };

    const click = (ev: React.MouseEvent) => {
        ev.stopPropagation(); ⑤
    };

    if (!getRowLoaded(row).loaded) { ⑥
        return <></>
    }

    return (
        <input
            className="background-rows__button"
            type="color"
            value={color}
            onClick={click}
            onChange={changeColor} />
    )
};
```

① Предоставляет возможность отобразить палитру цветов

② Исходный цвет: белый.

③ Обновляем хранилище.

④ Обновляем локальное состояние `input` с `type=color`.

⑤ Прекращает дальнейшую передачу текущего события.

⑥ Если строка ещё не загружена, то ничего не показываем.

Создаём сервис, который будет хранить и обновлять цвет строки:

```
export class BackgroundRowsService implements IBackgroundRowsService {
    $rows: Store<IBackgroundRows>;
    setBackground: Event<IBackgroundRow>;

    constructor() {
        const domain = createDomain("BackgroundRows"); ①

        this.setBackground = domain.event("setBackground");
        this.$rows = domain.store({}, { name: "$rows" })
            .on(this.setBackground, (store, background) => ({...store, [background.id]:
background.color }));
    }
}
```

① Инициализируем сущности необходимые для хранения и обновления хранилища.

Добавляем интерфейс сервиса:

```
export interface IBackgroundRows {
    [id: string]: string;
}

export interface IBackgroundRow {
    id: string;
    color: string;
}

export interface IBackgroundRowsService { ①
    readonly $rows: Store<IBackgroundRows>;
    setBackground: Event<IBackgroundRow>;
}
```

① Интерфейс сервиса.

Инициализируем сущности, необходимые для хранения и обновления хранилища:

```
export class BackgroundRowsService implements IBackgroundRowsService {
    $rows: Store<IBackgroundRows>;
    setBackground: Event<IBackgroundRow>;

    constructor() {
        const domain = createDomain("BackgroundRows");

        this.setBackground = domain.event("setBackground");
        this.$rows = domain.store({}, { name: "$rows" })
            .on(this.setBackground, (store, background) => ({...store, [background.id]:
background.color }));
    }
}
```

Подключаем плагины грида:

```
export class BackgroundRowsPluginFactory implements IFolderPluginFactory {
    id: string = "BackgroundRowsPluginFactory";

    getDataLoadingPlugins(): IFolderDataLoadingPlugin[] {
        return [
            new BackgroundRowsResponseResolver()
        ]
    }
```

```
    getTablePlugins(): ITablePlugins[] {
        return [
            BackgroundRowsPlugins
        ];
    }
}
```

Добавляем новую колонку для элемента переключения цвета:

```
export class BackgroundRowsResponseResolver implements IFolderDataLoadingPlugin {
    id: string = "BackgroundRowsResponseResolver";
    description: string = "Добавляет новую колонку.";
    order: PluginOrder = PluginOrder.Normal;

    async resolveResponse(data: ITableData, response: GenModels.GridViewModelEx):
Promise<void | ResponseResolveResult> { ①
        if (data.columns.length && !data.columns.find(x => x.id ==
BackgroundRowsColumnId)) { ②

            const backgroundRowsColumn = {
                id: BackgroundRowsColumnId, ③
                name: "Фон строки", ④
            } as IColumn;

            data.columns.push(backgroundRowsColumn);
        }
    }
}
```

① Вызывается после каждого ответа загрузки данных таблицы.

② Сделаем проверку на существование столбцов в таблице, а также проверим не был ли добавлен столбец, который мы хотим создать.

③ Уникальный идентификатор столбца.

④ Отображаемое название столбца.

Настраиваем элемент в ячейке:

```
export const BackgroundRowsColumnId = "backgroundRows";
export const BackgroundRowsFeature = "BackgroundRowsFeature";

export const BackgroundRowsServiceProvider: TablePlugins.ServiceProvider<$BackgroundRows>
= {
    name: "BackgroundRowsServiceProvider",
    description: "Добавляет сервис $BackgroundRows.",
```

```
    feature: BackgroundRowsFeature,
    composition: TableCompositionNames.Root,
    addServices: (composition) => {
        if (!composition.services.backgroundRows) { ①
            composition.services.backgroundRows = new BackgroundRowsService();
        }
    }
};

export const BackgroundRowsCellButtonPlugin: TablePlugins.Cell.Component<$BackgroundRows>
= {
    name: "BackgroundRowsCellButtonPlugin",
    description: "Отображает кнопку изменения фона строки.",
    feature: BackgroundRowsFeature,
    composition: TableCompositionNames.TableCell,
    shouldRender: (composition) => composition.data.column.id == BackgroundRowsColumnId,
②
    component: BackgroundRowsButton ③
};

export const BackgroundRowsMountEffect: TablePlugins.Row.MountEffect<$BackgroundRows> = {
    name: "BackgroundRowsMountEffect",
    description: "Обновляет композицию при изменении $rows",
    feature: BackgroundRowsFeature,
    composition: TableCompositionNames.TableRow,
    compositionDidMount: (composition) => { ④
        const { backgroundRows } = composition.services;

        const optimalUpdate = throttle({ source: backgroundRows.$rows, timeout: 100 });
⑤

        const update = optimalUpdate.watch(() => composition.update()); ⑥
        return () => update.unsubscribe(); ⑦
    }
};

export const BackgroundRowDecorator: TablePlugins.Row.Decorator<$BackgroundRows> = {
    name: "BackgroundRowDecorator",
    description: "Изменяет фон строки.",
    feature: BackgroundRowsFeature,
    composition: TableCompositionNames.TableRow,
    jsxDecorator: (node, composition) => { ⑧
        const { services, data } = composition;
        const backgroundRows = services.backgroundRows.$rows.getState(); ⑨

        return decorate(node, { style: { background: backgroundRows[data.row.entityId]
}}); ⑩
    }
```

```
};

export const BackgroundRowsPlugins: ITablePlugins = {
    serviceProviders: [ BackgroundRowsServiceProvider ],
    row: {
        mountEffects: [ BackgroundRowsMountEffect ],
        containerDecorators: [ BackgroundRowDecorator ]
    },
    cell: {
        content: [ BackgroundRowsCellButtonPlugin ]
    }
};
```

① Поскольку `addServices` вызывается при каждой отрисовке, сделаем проверку на существование сервиса, который хотим добавить.

② Компонент будет отображён, если условие правдиво.

③ Компонент, который будет отображён в ячейке добавленной колонки.

④ Вызывается при монтировании компоненты.

⑤ Поскольку хранилище `backgroundRow.$rows` может меняться достаточно быстро (зависит от скорости изменения цвета в палитре), необходимо установить ограничение обновления композиции.

В данном случае, `optimalUpdate` будет срабатывать не чаще, чем раз в `100` мс для оптимизации браузера.

⑥ Когда сработает событие, композиция таблицы будет обновлена.

⑦ Отписка от события, когда компонент будет размонтирован.

⑧ Метод, с помощью которого появляется возможность декорирования `ReactNode`.

⑨ Получаем хранилище, которое содержит информацию о цвете фона строк.

⑩ В данном методе вторым аргументом является объект свойств `ReactNode`, как в `React.createElement`.
В данном случае, если хранилище содержит информацию о фоне строки, она будет применена, в противном случае стиль не применится.

Проверка примера

1. Откройте папку с таблицей. В колонке "Фон строки" должен отображаться созданный элемент.
2. Нажмите на любой элемент в колонке и измените цвет.

Взаимодействие колонки таблицы со строками

Данный раздел содержит описание примера взаимодействия новой колонки таблицы с её строками. Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт позволяющий добавить новую колонку в таблицу, отрисовать элемент в её ячейках, при нажатии на который открывается палитра. Изменяя палитру, будет изменяться фон строки, в которой находится ячейка.

Ссылка на пример на GitHub: GridColumnBackgroundRows.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}

Сборка и установка

1. Откройте консоль в папке `GridColumnBackgroundRowsWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
2. Скопируйте каталог `SamplesOutput\Content\Modules\GridColumnBackgroundRowsWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
3. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Откройте папку с таблицей, где в колонке *Фон строки*, нажмите на любое поле ввода и изменить цвет.



Данный пример демонстрирует только взаимодействие новой колонки, её ячеек и фон строки, в которой они находятся, с помощью скрипта. Сохранение выбранного цвета строки не реализовано, поэтому при обновлении страницы цвет сбросится. При необходимости сохранение можно добавить в классе `BackgroundRowsService`.

Расширение возможностей серверной части Web-клиента

Серверное расширение — загружаемый в процесс сервера Web-клиента компонент, реализующий дополнительную функциональность серверного уровня.

Серверное расширение может добавлять:

- Методы WebApi.
- Поддержку новых типов карточек.
- Преобразователи значений элементов управления (binding-конвертер).
- Расширение или замена стандартных сервисов, с дополнительной логикой.
- Поддержка новых условий выбора разметок.
- Методы жизненного цикла карточек и строк.
- И другие точки расширения стандартной функциональности.

Создание и публикация серверного расширения

Серверное расширение представляет собой библиотеку DLL, разработанную на платформе .NET.

Ссылка на пример на GitHub: TemplateServerExtension.

Перечень необходимых инструментов

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.
- Docsvision WebClient SDK.

Описание файлов проекта

- В корне репозитория расположено решение для Visual Studio (`Samples.sln`) и ключ для подписания сборок примеров (`StrongNameKey.snk`). Рекомендуется использовать Visual Studio 2022.
- `Assemblies` — сборки, необходимые для использования примеров
- `ServerExtension.cs` — входная точка расширение, в которой регистрируются сервисы и прочие сущности.
- `Sign.snk` — файл подписи сборки. Для установки сборки должны быть подписаны. Рекомендуется сгенерировать новый файл подписи в настройках проекта.
- `Resource.resx`, `Resource.ru.resx` — локализации, используемые в расширении.
- `Feature1` — папка, содержащая реализацию некоторой функциональности.

- `Feature1/Feature1Controller.cs` — класс, регистрирующий конечные точки Web API.



При разработке собственных конечных точек рекомендуется всегда получать текущую сессию, т.к. при этом вызове происходит проверка доступа (в частности, проверка пользовательских cookie).

- `Feature1/IFeature1Service.cs` и `Feature1/Feature1Service.cs` — интерфейс и реализация сервиса, реализующего логику обработки запроса.
- `Feature1/Models` — классы моделей, используемых в сервисе и контроллере.

Сборка и установка

Чтобы создать серверное расширение:

1. Убедитесь, что установлен Docsvision WebClient SDKK.

   В этом случае в переменной окружения `DocsvisionWebClientSDK` содержится путь к его папке, например, `C:\Program Files\Docsvision\WebClient\SDK`.

2. Установите шаблон серверного расширения и создайте на его основе новый проект с названием `LearnDocvisionServerExtension`.
3. Обновите в проекте ссылки на зависимости.

   После крупного обновления версии Docsvision, необходимо повторно скомпилировать расширение с новыми версиями компонентов. В остальных случаях такой необходимости нет.

   Это необходимо, чтобы версии подключаемых компонентов соответствовали версиям компонентов, используемых сервером Web-клиента, для которого разрабатывается расширение.

4. Реализуйте в проекте функции серверного расширения.

   Примеры кода серверных расширений приведены в следующих пунктах раздела, а также в репозитории "ДоксВижн".

5. Настройте ядро серверного расширения.

Ядром серверного расширения называется файл ServerExtension.cs в проекте расширения. Ядро является производным от WebClientExtension классом, содержащим ссылки на реализуемые функции расширения.

```
public class ServerExtension : WebClientExtension
{
    public ServerExtension(IServiceProvider serviceProvider)
        : base()
    {
    }

    public override string ExtensionName ①
    {
        get { return Assembly.GetAssembly(typeof(ServerExtension)).GetName().
Name; }
    }

    public override Version ExtensionVersion ②
    {
        get { return new Version(FileVersionInfo.GetVersionInfo(Assembly
.GetExecutingAssembly().Location).FileVersion); }
    }

    public override void InitializeServiceCollection(IServiceCollection services)
③
    {
④
        services.AddSingleton<IFeature1Service, Feature1Service>();
    }

    protected override List<ResourceManager> GetLayoutExtensionResourceManagers()
    {
        return new List<ResourceManager>
        {

        };
    }
⑤

}
```

① Название расширения.

② Версия расширения.

③ Метод регистрации добавляемых расширением сервисов,

преобразователей, конвертеров и т.п. в IoC контейнере.

④ Примеры регистрации.

⑤ Далее идут функции, возвращающие реализованные функции серверного расширения.

Описание изменений, которые необходимо выполнить в ядре расширения, приведено в описании примеров расширений, добавляющих обработчики событий и локализованные ресурсы.

В серверном расширении Web-клиента используется IoC контейнер, реализованный с использованием библиотеки Autofac. Описание данной библиотеки и её возможностей см. на сайте autofac.org.

6. Скомпилируйте проект.

В результате будет получен файл DLL с ядром серверного расширения и реализуемыми функциями, а также файлы DLL с локализованными ресурсами.

7. Скопируйте `safeprojectnamesafeprojectname.dll` и `safeprojectnamesafeprojectname.dll.pdb` из папки **bin** › **WebClient** › **Bin** в **Каталог-установки-Web-клиента** › **bin**.

Расширения можно устанавливать в папки **Каталог-установки-Web-клиента** › **Extensions** и **Каталог-установки-Web-клиента** › **bin**. В ходе разработки расширения можно удобнее устанавливать в папку **Каталог-установки-Web-клиента** › **bin**, т.к. это избавляет от необходимости перезапускать службу Сервера Web-клиента при установке сборок.



В ресурсной поставке рекомендуется устанавливать сборки в папку **Каталог-установки-Web-клиента** › **Extensions**, чтобы проще было видеть, какие расширения установлены в Web-клиенте. Однако этот способ требует перезапуска службы Сервера Web-клиента при установке или обновлении сборки.

При обновлении существующих файлов достаточно обновить страницу в браузере с очисткой кэша (например `CTRL` + `SHIFT` + `R`)

8. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка работы

1. Откройте Web-клиент.
2. Откройте диалог "О программе" и убедитесь, что расширение присутствует в списке подключенных расширений.
3. Откройте произвольный документ и в консоли браузера выполните:

```
$ await
layoutManager.cardLayout.getService("requestManager").post("api/Feature1/Action1",
JSON.stringify({ documentId: layoutManager.cardLayout.getService("cardId") }));
```

4. Обновите документ, убедиться, что в названии добавился знак.

Разработка расширения с методами WebApi

Данный раздел содержит пример добавления новых web-методов, реализующих функции взаимодействия с сервером Docsvision (создание и изменение карточек, изменение данных, запуск бизнес-процессов), а также выполнения других задач, для которых необходимы возможности сервера Web-клиента.

Чтобы создать расширение, добавляющее новые методы WebApi, выполните приведенную далее инструкцию.

1. Создайте проект серверного расширения.
2. Добавьте в проект класс WebApi контроллера (производный от `System.Web.Http.ApiController`).
3. Реализуйте в классе контроллера веб-методы.

   Следующий код демонстрирует пример реализации контроллера с методом, проверяющим существование карточки в Docsvision.

```
using DocsVision.BackOffice.ObjectModel;
using DocsVision.Platform.WebClient.Helpers;
using System;
using System.Web.Http;

namespace WebApiServerExtension
{
    public class CardServiceController : ApiController
```

```
    {
        private readonly ICurrentObjectContextProvider
currentObjectContextProvider; ①

        public CardServiceController(ICurrentObjectContextProvider
currentObjectContextProvider) ②
        {
            this.currentObjectContextProvider = currentObjectContextProvider;
        }

        [HttpGet]
        public bool Exists(Guid cardId) ③
        {
            var objectContext = currentObjectContextProvider
.GetOrCreateCurrentSessionContext().ObjectContext; ④

            try
            {
                var card = objectContext.GetObject<BaseCard>(cardId); ⑤

                return (card != null); ⑥
            }
            catch
            {
                return false;
            }
        }
    }
}
```

① В конструкторе контроллера получаем ссылку на сервис-провайдер,

② Из сервис-провайдера можно получать различные сервисы объектной модели, а также сервис для получения контекста объектов.

③ Метод, проверяющий существует ли карточка с идентификатором cardId. Если карточка существует, метод возвращает true, иначе — false.

④ Получаем пользовательский контекст объектов.



Всегда нужно получать контекст объектов заново. Дополнительная информация в пункте Получение контекста объектов в серверном расширении.

⑤ Получаем карточку с указанным идентификатором.

⑥ Возвращаем true, если при получении карточки из контекста объектов не было ошибки и полученный объект не null.

4. Скомпилируйте проект и скопируйте полученную сборку на сервер Web-клиента в папку `/lib/docsvision/webclient/Extensions/%Каталог Решения%`.
5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Для проверки работоспособности примера, откройте в веб-браузере страницу `http://%WebCinstallDir%/DocsvisionWebClient/api/CardService/Exists?cardId=<CardID>`. Вместо *CardID* укажите идентификатор любой существующей карточки. Метод вернет значение **true** (если карточки не существует — **false**).

Пример клиентского расширения, использующего метод `Exists`, приведён в пункте Разработка расширения с клиентским сервисом.

Получение контекста объектов в серверном расширении

Программист может использовать в серверном расширении контекст объектов Docsvision для доступа к данным системы.

Разработчику доступен пользовательский контекст объектов. Пользовательский контекст создаётся для каждого пользователя при его входе в Web-клиент. Права на работу с объектами соответствуют правам данного пользователя в системе Docsvision.

Сервис `ICurrentObjectContextProvider` предоставляет методы для получения контекста объектов. `ICurrentObjectContextProvider` может быть получен следующим способом:

```
private readonly ICurrentObjectContextProvider _currentObjectContextProvider;

public DocumentRegistrationController(ICurrentObjectContextProvider
currentObjectContextProvider)
{
    _currentObjectContextProvider = currentObjectContextProvider;
}
```

Получить контекст объектов можно используя следующие функции сервиса

`ICurrentObjectContextProvider`.

- `GetOrCreateCurrentSessionContext` — возвращает контекст объектов текущего пользователя.
- `GetOrCreateExclusivePoolSessionContext` — возвращает контекст объектов из отдельного пула сессий Web-клиента.

Web-клиент периодически закрывает неиспользуемые пользовательские сессии, из-за чего контекст объектов, сохранённый в статический объект класса (серверного расширения или контроллера), спустя время станет недействительным. **Всегда получайте контекст объектов заново** (с помощью функции `GetOrCreateCurrentSessionContext`).

Примеры работы с контекстом объектов рассмотрены в Руководстве разработчика. Следующий код демонстрирует пример получения объектной модели карточки *Документ*.

```
Document document = objectContext.GetObject<Document>(Guid.Parse("9E11083F-D0EC-4F9F-8981-285498280BD9")); ①
```

① В метод GetObject передаются тип объекта (Document) и идентификатор объекта (ID карточки).



Не используйте метод `GetOrCreateApplicationPoolSessionContext`, это приведет к блокировкам потоков под нагрузкой.

После использования полученный через `GetOrCreateExclusivePoolSessionContext` контекст необходимо освобождать, т.к. он арендует сессию из отдельного пула сессий Web-клиента. После завершения работы сессия должна быть возвращена обратно в пул.

Количество одновременно арендованных сессий ограничено пятью. Если полученные контексты не освобождать, новые запросы сессии могут завершиться ошибкой из-за исчерпания лимита.

```
using (var pooledSessionContext = currentObjectContextProvider
.GetOrCreateExclusivePoolSessionContext())
{
    var sessionContext = pooledSessionContext.Context; ①
}
```

① Работа с `SessionContext`.

Добавление сервисов и поддержки новых типов карточек в контекст объектов

По умолчанию в контекст объектов, предоставляемый в серверном расширении, загружаются сервисы и преобразователи данных (см. документ Руководство разработчика Docsvision), позволяющие работать с карточками библиотек *Базовые объекты* и *Конструктор согласований*, системными карточками и карточками приложения *Управление документами*: `SystemCardsMapperFactory`, `BackOfficeMapperFactory`, `ApprovalDesignerMapperFactory`, `SystemCardsServiceFactory`, `BackOfficeServiceFactory`, `DocumentsManagementServiceFactory` и `ApprovalDesignerServiceFactory`.

Если требуется поддержать собственные типы карточек в объектной модели Docsvision, программист должен добавить в контекст объектов преобразователи данных для этих типов, и сервисы для работы с данными карточками. Инструкция по разработке фабрик сервисов и фабрик преобразователей данных приведена в Руководстве разработчика Docsvision.

Выполните указанные ниже действия для добавления сервисов и преобразователей данных в контекст объектов. Все действия выполняются в проекте серверного расширения (нового или существующего).

1. Подключите к проекту сборки, в которых реализованы классы разработанных фабрик сервисов и преобразователей данных.
2. Откройте файл ядра расширения.
3. Переопределите метод `OnObjectContextCreate`, вызываемый при создании контекста объектов, добавив в него код регистрации фабрик сервисов и преобразователей данных.

   Следующий код демонстрирует пример добавления в контекст объектов фабрики сервисов `CustomLibraryServiceFactory` и фабрики преобразователей данных `CustomLibraryMapperFactory`.

```
public override void OnObjectContextCreate(ObjectContext objectContext)
 {
    base.OnObjectContextCreate(objectContext);

①

    IObjectMapperFactoryRegistry mapperFactoryRegistry = objectContext.GetService
<IObjectMapperFactoryRegistry>(); ②
```

```
    mapperFactoryRegistry.RegisterFactory(typeof(CustomLibraryMapperFactory)); ③

    IServiceFactoryRegistry serviceFactoryRegistry = objectContext.GetService
<IServiceFactoryRegistry>(); ④

    serviceFactoryRegistry.RegisterFactory(typeof(CustomLibraryServiceFactory)); ⑤
}
```

① Далее следует стандартный код добавления фабрик в контекст объектов (см. Руководство разработчика Docsvision).

② Получаем из существующего контекста объектов сервис регистрации фабрик преобразователей данных.

③ Регистрируем фабрику преобразователей данных, реализованную в классе CustomLibraryMapperFactory.

④ Получаем из существующего контекста объектов сервис регистрации фабрик сервисов.

⑤ Регистрируем фабрику сервисов, реализованную в классе `CustomLibraryServiceFactory`.
Добавленный в контекст объектов сервис можно получить стандартным образом:

```
ISomeService someService = objectContext.GetService<ISomeService>();
```

+ . Опубликуйте серверное расширение.

+ NOTE: Сборки, содержащие классы фабрик сервисов и преобразователей данных, нужно скопировать в каталог со сборкой данного серверного расширения.

Методы управления жизненным циклом объектов Web-клиента

В данном пункте рассмотрены механизмы API Web-клиента, которые позволяют влиять на функции создания, изменения и удаления карточек и строк секций из разметок, а также сервис `ILifeCycleService`, предоставляющий доступ к данным механизмам.

Реализация методов для работы с карточками из разметок

Программисту предоставляется возможность добавить новую функциональность в механизмы создания и работы с карточками из разметок. Данная возможность

реализуется с помощью специального класса с интерфейсом ICardLifeCycleEx. Интерфейс ICardLifeCycleEx предоставляет методы, которые вызываются на различных этапах работы с карточкой:

```
    public interface ICardLifeCycleEx
    {
        Guid CardTypeId { get; } ①
        Guid Create(SessionContext sessionContext, CardCreateLifeCycleOptions options);
②
        bool Validate(SessionContext sessionContext, CardValidateLifeCycleOptions
options, out List<ValidationResult> validationResults); ③
        void OnSave(SessionContext sessionContext, CardSaveLifeCycleOptions options); ④
        bool CanDelete(SessionContext sessionContext, CardDeleteLifeCycleOptions options,
out string message); ⑤
        void OnDelete(SessionContext sessionContext, CardDeleteLifeCycleOptions options);
⑥
        string GetDigest(SessionContext sessionContext, CardDigestLifeCycleOptions
options); ⑦
    }
```

① Идентификатор типа карточек.

② `Create` — Вызывается при получении разметки создания карточки.

③ `Validate` — Вызывается перед сохранением изменений карточки в БД с возможностью прервать сохранение.

④ `OnSave` — Вызывается при сохранении изменений карточки. Позволяет добавить дополнительную логику в сохранение карточки, но без возможности прервать сохранение.

⑤ `CanDelete` — Вызывается перед удалением карточки. Позволяет прервать удаление.

⑥ `OnDelete` — Вызывается при удалении карточки. Позволяет добавить дополнительную логику в удаление карточки, но без возможности прервать удаление.

⑦ `GetDigest` — Вызывается при получении дайджеста карточки при её сохранении. Позволяет добавить собственную логику формирования дайджеста. Метод `GetDigest` используется, только когда способ формирования дайджеста не настроен в Docsvision и карточка сохраняется из пользовательского интерфейса (при нажатии кнопки Сохранить, сохранении по месту или скриптом).

Указанные методы используются **исключительно** при выполнении стандартных

операций со строками из разметок или с использованием JS WebAPI разметок. В других случаях методы интерфейса `ICardLifeCycleEx` не вызываются и не влияют на работу сервера Web-клиента с карточками.

Следующий код демонстрирует пример реализации класса, в котором определены собственные функции создания и сохранения карточки.

```
using System;
using DocsVision.BackOffice.ObjectModel.Services;
using DocsVision.BackOffice.ObjectModel;
using DocsVision.Platform.WebClient;
using DocsVision.WebClientLibrary.ObjectModel.Services.EntityLifeCycle;
using DocsVision.WebClientLibrary.ObjectModel.Services.EntityLifeCycle.Options;
using DocsVision.BackOffice.CardLib.CardDefs;


internal class DocumentCardLifeCycle : BaseCardLifeCycleEx
{
    public override Guid CardTypeId => CardDocument.ID;

    public override Guid Create(SessionContext sessionContext, CardCreateLifeCycleOptions
options)
    {
        if (!options.CardKindId.HasValue)
            throw new ArgumentNullException(nameof(options.CardKindId));

        var objectContext = sessionContext.ObjectContext;
        var kindsCardKind = objectContext.GetObject<KindsCardKind>(options.CardKindId);
        var documentService = objectContext.GetService<IDocumentService>();
        var document = documentService.CreateDocument(null, kindsCardKind);
        document.Description = document.SystemInfo.CardKind.Name;
        document.MainInfo.Name = "Наименование документа";

        objectContext.AcceptChanges();
        return document.GetObjectId();
    }


    public override void OnSave(SessionContext sessionContext, CardSaveLifeCycleOptions
options)
    {
        base.OnSave(sessionContext, options);
        var objectContext = sessionContext.ObjectContext;
        var document = objectContext.GetObject<Document>(options.CardId);
        document.MainInfo.Name = "Принудительно установленное наименование документа";
```

```
        objectContext.AcceptChanges();
    }
}
```

В приведенном коде класс `DocumentCardLifeCycle` наследуется от базовой реализации интерфейса `ICardLifeCycleEx` — класса `BaseCardLifeCycle` с абстрактными методами. Помимо класса `BaseCardLifeCycle` с базовой реализацией, API включает реализующий интерфейс `ICardLifeCycleEx` класс `DefaultCardLifeCycle` с реализацией по умолчанию методов `Create` и `CreateFromTemplate`.

Метод `Create` в приведенном выше коде создаёт новую карточку *Документ* с предустановленным названием (`document.MainInfo.Name`). Метод `CreateFromTemplate` создаёт карточку *Документа* по шаблону. Метод `Save` изменяет значение поля `document.MainInfo.Name` при сохранении карточки.

Тип с интерфейсом `ICardLifeCycleEx` нужно добавить в IoC-контейнер в ядре расширения:

```
public override void InitializeServiceCollection(IServiceCollection services)
{
    services.AddTransient<ICardLifeCycleEx, DocumentCardLifeCycle>();
}
```

Серверное расширение должно быть опубликовано стандартным образом.

Реализация методов для работы со строками секций из разметок

Программисту предоставляется возможность добавить новую функциональность в механизмы создания и работы со строками секциями из разметок. Данная возможность реализуется с помощью специального класса с интерфейсом `IRowLifeCycleEx`. Интерфейс `IRowLifeCycleEx` предоставляет методы, которые вызываются на различных этапах работы со строкой секции:

```
public interface IRowLifeCycleEx
{
    Guid SectionId { get; } ①
    Guid Create(SessionContext sessionContext, RowCreateLifeCycleOptions options); ②
    bool Validate(SessionContext sessionContext, RowValidateLifeCycleOptions options, out
List<ValidationResult> validationResults); ③
    void OnSave(SessionContext sessionContext, RowSaveLifeCycleOptions options); ④
    bool CanDelete(SessionContext sessionContext, RowDeleteLifeCycleOptions options, out
string message); ⑤
```

```
    void OnDelete(SessionContext sessionContext, RowDeleteLifeCycleOptions options); ⑥
}
```

① Идентификатор секции, для строк которой предоставляется реализация интерфейса IRowLifeCycleEx.

② `Create` — Вызывается при получении разметки создания новой строки секции.

③ `Validate` — Вызывается перед сохранением изменений строки в БД, с возможностью прервать сохранение.

④ `OnSave` — Вызывается при сохранении изменений строки — позволяет добавить дополнительную логику в сохранение строки, но без возможности прервать сохранение.

⑤ `CanDelete` — Вызывается перед удалением строки секции — позволяет прервать удаление.

⑥ `OnDelete` — Вызывается при удалении строки секции — позволяет добавить дополнительную логику в удаление строки, но без возможности прервать удаление.

Указанные методы используются **исключительно** при выполнении стандартных операций со строками из разметок или с использованием JS WebAPI разметок. В других случаях методы интерфейса `IRowLifeCycleEx` не вызываются и не влияют на работу сервера Web-клиента со строками секций.

Следующий код демонстрирует пример реализации класса, в котором определена собственная функция проверки данных строки секции перед сохранением изменений (`Validate`).

```
using System.Collections.Generic;
using DocsVision.BackOffice.CardLib.CardDefs;
using DocsVision.Platform.WebClient;
using DocsVision.WebClientLibrary.ObjectModel.Services.EntityLifeCycle;
using DocsVision.WebClientLibrary.ObjectModel.Services.EntityLifeCycle.Options;

public class RefPartnersCompaniesRowLifeCycle : DefaultRowLifeCycleEx ①
{
    public RefPartnersCompaniesRowLifeCycle()
        : base(RefPartners.Companies.ID)
    {
    }

    public override bool Validate(SessionContext sessionContext,
RowValidateLifeCycleOptions options, out List<ValidationResult> validationResults)
```

```
    {
        var isValid = base.Validate(sessionContext, options, out validationResults);
        var rowData = sessionContext.AdvancedCardManager.GetRowData(options.CardId,
RefPartners.Companies.ID, options.RowId);
        var companyName = rowData.GetString(RefPartners.Companies.Name);
        if (string.IsNullOrEmpty(companyName))
        {
            isValid = false;
            var validationResult = new ValidationResult
            {
                Message = "Наименование компании не заполнено"
            };
            validationResults.Add(validationResult);
        }
        return isValid;
    }
}
```

① `RefPartnersCompaniesRowLifeCycle` наследуется от стандартной реализации интерфейса `IRowLifeCycleEx` — класса `DefaultRowLifeCycle`, в котором уже реализованы необходимые методы. Помимо класса `DefaultRowLifeCycle` с реализацией по умолчанию API включает реализующий интерфейс `IRowLifeCycleEx` базовый класс `BaseRowLifeCycle` с абстрактными методами.

Метод `Validate` проверяет строку `rowId` на заполненность поля `RefPartners.Companies.Name`, и, если данное поле пустое, возвращает ошибку `validationResult`.

Полученный тип с интерфейсом `IRowLifeCycleEx` нужно добавить в IoC-контейнер в ядре расширения:

```
public override void InitializeServiceCollection(IServiceCollection services)
{
    services.AddTransient<IRowLifeCycleEx, RefPartnersCompaniesRowLifeCycle>();
}
```

Серверное расширение должно быть опубликовано стандартным образом.

Работа с сервисом жизненного цикла объектов Web-клиента

Рассмотренные в пунктах [WebClient:server-card-factory] и [WebClient:server-row-factory] компоненты, которые позволяют внести изменения в механизмы создания, изменения и удаления карточек и строк секций из разметок, доступны программисту через специальный сервис `ILifeCycleServiceEx`. Данный сервис

определяет методы:

- `GetCardLifeCycle` — Возвращает реализацию интерфейса `ICardLifeCycleEx` для указанного типа карточек.
- `GetRowLifeCycle` — Возвращает реализацию интерфейса `IRowLifeCycleEx` для указанной секции.

Следующий код демонстрирует способ получения сервиса `ICardLifeCycleEx` и его использования в веб-методе для создания карточки по шаблону:

```
public class CardServiceController : ApiController
{
    private readonly ILifeCycleServiceEx lifeCycleService;
    private readonly ICurrentObjectContextProvider currentObjectContextProvider;

    public CardServiceController(ILifeCycleServiceEx lifeCycleService,
ICurrentObjectContextProvider currentObjectContextProvider) ①
    {
        this.lifeCycleService = lifeCycleService;
        this.currentObjectContextProvider = currentObjectContextProvider;
    }

    [HttpGet]
    public Guid CreateByTemplate(Guid templateId)
    {
        var sessionContext = currentObjectContextProvider
.GetOrCreateCurrentSessionContext();
        var documentCardLifeCycle = lifeCycleService.GetCardLifeCycle(CardDocument.ID);
②
        var newDocumentId = Guid.Empty;
        if (templateId != Guid.Empty)
        {
            newDocumentId = documentCardLifeCycle.Create(sessionContext, new
CardCreateLifeCycleOptions()
            {
                TemplateId = templateId ③
            });
        }
        return newDocumentId;
    }
}
```

① Принимаем требуемый сервис в конструкторе.

② Получаем интерфейс для работы с карточкой типа `CardDocument`.

③ Используем метод создания карточки по шаблону.

Изменение штампа электронной подписи

Пример демонстрирует вставку собственных изображений штампов электронной подписи в файл, приложенный к карточке Docsvision.

Ссылка на пример на GitHub: CSPSignatureVisualization.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Создайте описание расширения для Web-клиента, которое задано в текущей сборке

```
    public class CSPSignatureVisualizationExtension : WebClientExtension
    {
        public CSPSignatureVisualizationExtension() ①
        {
        }

        #region Свойства

        public override string ExtensionName ②
            => Assembly.GetAssembly(typeof(CSPSignatureVisualizationExtension)).
GetName().Name;

        public override Version ExtensionVersion ③
            => new Version(FileVersionInfo.GetVersionInfo(Assembly.
GetExecutingAssembly().Location).FileVersion);

        #endregion Свойства

        #region Обработчики событий

        public override void InitializeServiceCollection(IServiceCollection services) ④
        {
            services.AddSingleton<IImageGenerator, CSPSignatureImageGenerator>(); ⑤

            services.AddSingleton<IImagePositionSelector,
CSPSignatureImagePositionSelector>(); ⑥
        }
```

```
        #endregion Обработчики событий
    }
```

① Создаём новый экземпляр класса `CSPSignatureVisualizationExtension`.

Параметры:

- `serviceProvider` — сервис-провайдер.

② Получаем название расширения.

③ Получаем версию расширения.

④ Регистрируем типы в IoC контейнере.

Параметры:

- `containerBuilder` — контейнер.

⑤ Регистрируем тип `CSPSignatureImageGenerator` как реализацию интерфейса `IImageGenerator` в `containerBuilder`.

⑥ Регистрируем тип `CSPSignatureImagePositionSelector` как реализацию интерфейса `IImagePositionSelector` в `containerBuilder`.

Настройка штампа ЭП

Описание генератора изображений штампов электронной подписи выглядит следующим образом:

```
    public class CSPSignatureImageGenerator : IImageGenerator ①
    {
        private const double ViewBoxHeight = 88; ②

        private const double ViewBoxWidth = 314; ③

        private const double ContentTop = 64; ④

        private const double FontHeight = 14; ⑤

        private const int MaxStringsLength = 38; ⑥
⑦
```

① Высота изображения штампа электронной подписи.

② Ширина изображения штампа электронной подписи.

③ Левая верхняя координата информационного содержимого штампа электронной подписи.

④ Высота используемого шрифта в пикселях (Используется десятый размер шрифта).

⑤ Максимальная длина используемого в изображении штампа электронной подписи текста. При стандартном интервале между символами. Используется для разбиения текста на отдельные строки.

⑥ Полный код приведён в файле **CSPSignatureVisualization › CSPSignatureVisualizationServerExtension › DataVisualization › CSPSignatureImageGenerator.cs** в репозитории примера на GitHub.

Описание класса, вставляющего в PDF-файл изображения штампов электронной подписи:

```
namespace CSPSignatureVisualizationServerExtension.DataVisualization
{
    public class CSPSignatureImagePositionSelector : IImagePositionSelector
    {
        private static readonly double ImageMargin = ImageHelper.PixelsToUnits(5); ①

        private static readonly int PagePadding = 30; ②

        private const int PageNumber = 0; ③

        #region Реализация интерфейса IImagePositionSelector

        public GeneratedFileInfo Generate(SessionContext sessionContext,
ImageGeneratorType generatorType, List<byte[]> images, Stream fileStream) ④
        {
            ArgumentNullException.ThrowIfNull(images);
            ArgumentNullException.ThrowIfNull(fileStream);

            using var pdfDocument = PdfReader.Open(fileStream);
            var page = pdfDocument.Pages[PageNumber];

            var pageWidth = page.Width.Value;
            var pageHeight = page.Height.Value;

            double xIndent = PagePadding; ⑤
            double yIndent = PagePadding;

            double maxImageHeight = 0; ⑥

            foreach (var imageHelper in images.Select(x => new ImageHelper(x)))
            {
                if (yIndent + imageHelper.Height > pageHeight) ⑦
```

```
                    break;

                if (xIndent + imageHelper.Width > pageWidth) ⑧
                {
                    yIndent += maxImageHeight + ImageMargin;
                    xIndent = PagePadding;
                    maxImageHeight = imageHelper.Height;
                }
                else if (imageHelper.Height > maxImageHeight) ⑨
                {
                    maxImageHeight = imageHelper.Height;
                }

                InsertImage(pdfDocument, page, imageHelper, xIndent, yIndent); ⑩

                xIndent += imageHelper.Width + ImageMargin; ⑪
            }

            var outputStream = new MemoryStream(); ⑫
            pdfDocument.Save(outputStream);
            pdfDocument.Close();
            outputStream.Seek(0, SeekOrigin.Begin);

            return new GeneratedFileInfo { Stream = outputStream };
        }
⑬
```

① Расстояние между изображениями штампов электронной подписи.

② Координаты первого изображения штампа электронной подписи. 30 миллиметров.

③ Номер страницы, на которую требуется вставить изображение штампа.

④ Вставляет изображения штампов электронной подписи на первую страницу указанного PDF-файла.

Параметры:

- `sessionContext` — контекст сессии.
- `generatorType` — тип генератора изображений штампов электронной подписи.
- `images` — массив изображений штампов электронной подписи.
- `fileStream` — поток, содержащий PDF-файл, в который необходимо вставить указанные изображения.

Возвращаемое значение: модель файла со вставленными изображениями.

⑤ Задаём начальные координаты первого изображения штампов электронной подписи.

⑥ Максимальная высота изображения штампа в текущей строке (нужна для выравнивания изображений в строке).

⑦ Если верхняя левая координата текущего изображения больше высоты страницы PDF-документа, прекращаем вставку изображений.

⑧ Если ширина строки вставленных изображений превышает ширину страницы, переходим на следующую строку.

⑨ В противном случае вычисляем максимальную высоту изображения штампа в текущей строке.

⑩ Вставляем текущее изображение по указанным координатам.

⑪ Вычисляем горизонтальную координату следующего изображения.

⑫ Копируем в поток сформированный PDF-файл со вставленными изображениями штампов электронной подписи.

⑬ Полный код приведён в файле **CSPSignatureVisualization › CSPSignatureVisualizationServerExtension › DataVisualization › CSPSignatureImagePositionSelector.cs** в репозитории примера на GitHub.

Пример взаимодействия с сервисом конвертации

Этот пример серверного расширения демонстрирует взаимодействие с сервисом конвертации.

Ссылка на пример на GitHub: ConversionSample.

Необходимое ПО

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.



Для работы расширения требуется наличие в лицензии опции *Docsvision Сервис конвертации файлов*.

Сборка

1. Откройте `/Samples.sln`

2. Соберите проект **ServerExtensions** › **ConversionSample** › **ConversionSampleServerExtension**.
3. Откройте консоль в папке **ServerExtensions** › **ConversionSample** › **ConversionSampleWebExtension**, выполните `npm install`, `npm update` и `npm run build:prod`.
4. Скопируйте каталог `SamplesOutput\Content\Modules\ConversionSampleWebExtension` в каталог `Путь-к-установленному-Web-клиенту\Content\Modules`.
5. Скопируйте каталог `SamplesOutput\Extensions\ConversionSampleServerExtension` в каталог `Путь к установленному Web-клиенту\Extensions`.
6. Перезапустите службу **dvappserver** для Linux и **Docsvision Platform** для Windows.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Откройте Конструктор Web-разметок.
2. В разметке просмотра документа на событие `Перед стартом согласования` элемента управления `agreementManagement` назначьте обработчик `attachPdfa`.
3. Создайте карточку документа, приложите к ней `.docx`-файл и отправьте на согласование.
4. К документу будет добавлен файл, конвертированный в PDF/A.

Изменение штампа электронной подписи

Пример взаимодействия с сервисом конвертации

Изменение экспорта в Excel

Пример серверного расширения, позволяющего вмешаться в процесс экспорта представления в Excel.

Ссылка на пример на GitHub: ExcelExport.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Настраиваем собственные параметры экспорта:

```
namespace ExcelExportServerExtension.ExcelExport ①
{
    public class ExcelExportExampleService : ExcelExportService
    {
        private readonly Color oddRowColor = Color.LightBlue; ②

        protected override void ApplyWorkbookSettings(XLWorkbook workbook) ③
        {
            foreach (var worksheet in workbook.Worksheets)
            {
                for (var i = 1; i < worksheet.RangeUsed().RowCount(); i += 2) ④
                {
                    var row = worksheet.RangeUsed().Row(i);
                    row.Style.Fill.BackgroundColor = XLColor.FromColor(oddRowColor);
                }
            }
        }

        protected override string GetGridRowParamValueAsString(object paramValue, bool
noNullValues = false) ⑤
        {
            if (paramValue is DateTime dateTime) ⑥
            {
                return dateTime.ToLongDateString();
            }

            return base.GetGridRowParamValueAsString(paramValue);
        }

        protected override IEnumerable<GridColumn> GetGridColumns(GridViewModel
viewModel) =>
            viewModel.Columns.Select(gridColumn => CreateGridColumnModel(gridColumn,
viewModel))
                .OrderBy(column => column.UserColumn?.Order ?? 0)
                .Select(column => column.GridColumn); ⑦

        private static GridColumnModel CreateGridColumnModel(GridColumn gridColumn,
GridViewModel viewModel) => new GridColumnModel
        {
            GridColumn = gridColumn,
            UserColumn = viewModel.GridUserSettings
                .GetColumnsForCurrentPresentation()
                .FirstOrDefault(x => x.Name == gridColumn.Name)
        };
    }
```

```
}
```

① Описание класса, реализующего свой экспорт в Excel.

② Изменяем цвет нечётных строк.

③ Модифицирует документ Excel.

④ Меняем фоновый цвет нечётных строк в таблице.

⑤ Получаем строковое значение ячейки таблицы.

⑥ Для ячейки типа `DateTime` устанавливаем длинный формат.

⑦ Получаем все колонки из грида и сортируем их по параметру `Order`.

В методе `InializeContainer` серверного расширения зарегистрируйте сервис.

```
        public override void InitializeContainer(ContainerBuilder containerBuilder) ①
        {
            containerBuilder.RegisterType<ExcelExportExampleService>()
                .As<IExcelExportService>()
                .SingleInstance(); ②
        }
```

① Регистрируем типы в IoC контейнере.

② Регистрируем тип `ExcelExportExampleService` как реализацию интерфейса `IExcelExportService` в `containerBuilder`.

Проверка примера

1. Откройте папку в Web-клиенте Docsvision.
2. Нажмите кнопку **Экспортировать в Excel**.
3. Загрузится документ в формате `.xlsx`, содержащий выгрузку представления со всеми колонками и выделенными цветом нечетными строками.

Создание плагина для наполнения контрола табличных данных

ЭУ Контрол табличных данных отображает данные в виде таблицы, наполняемой при помощи плагина. На данной странице описан способ создания плагина.

Ссылка на пример на GitHub: DataGridControlExtension.

`Название плагина` указывается в конструкторе разметок, в свойствах элемента управления. При открытии разметки карточки в Web-клиенте из настроек ЭУ будет считано `Название плагина`. Затем отправляется запрос на сервер, который

обнаруживает нужную реализацию интерфейса и возвращает модель для отображения таблицы.

Чтобы реализовать собственный плагин:

1. Выполните наследование от интерфейса `IDataGridControlPlugin (DocsVision.Layout.WebClient.Services)`.

   Наименование плагина:

   ```
   string Name { get; }
   ```

   Метод формирования модели таблицы:

   ```
   TableModel GetTableData(SessionContext sessionContext, List<ParamModel> parameters);
   ```

2. Затем зарегистрируйте тип в IoC контейнере:

   ```
   public override void InitializeContainer(ContainerBuilder containerBuilder) {

   containerBuilder.RegisterType<TestTablePlugin>().As<IDataGridControlPlugin>().SingleIn
   stance();

   }
   ```

В случае, если это разметка карточки, то в массив параметров будет добавлен её идентификатор (`CurrentCardId`).

Создание карточки при помощи скрипта

Данный раздел содержит описание примера создания карточки скриптом с заполнением атрибутов.

Ссылка на пример на GitHub: CreateCard.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проект `ServerExtensions > CreateCard > CreateCardServerExtension`.
3. Откройте консоль в папке `ServerExtensions > CreateCard > CreateCardWebExtension` и выполнить команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
4. Скопируйте каталог `SamplesOutput\Content\Modules\CreateCardWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
5. Скопируйте каталог `SamplesOutput\Extensions\CreateCardServerExtension` в каталог `Путь к сайту Web-клиента\Extensions`.
6. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте любую разметку просмотра.
3. Выбрать условия использования этой разметки.
4. Откройте разметку и добавить в нее элемент управления `Кнопка`.
5. На событие `onClick` задать функцию `createOutgoingDocument`.
6. Сохраните разметку.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
8. Откройте карточку с этой разметкой.
9. Убедитесь, что появился новый элемент управления — кнопка, при нажатии на которую создается исходящий документ.
10. Должен открыться исходящий документ, с заданными атрибутами.

Проект "CreateCardServerExtension"

1. Проект-расширение для Web-клиент. Содержит бизнес-логику и скрипт для создания новой карточки.

```
namespace CreateCardServerExtension
{
    public class LayoutWebClientExtension : WebClientExtension ①
    {
        public LayoutWebClientExtension(IServiceProvider serviceProvider) ②
            : base()
        {
        }
```

```
        public override string ExtensionName ③
        {
            get { return Assembly.GetAssembly(typeof(LayoutWebClientExtension
)).GetName().Name; }
        }

        public override Version ExtensionVersion ④
        {
            get { return new Version(FileVersionInfo.GetVersionInfo(Assembly
.GetExecutingAssembly().Location).FileVersion); }
        }

        #region WebClientExtension Overrides

        public override void InitializeServiceCollection(IServiceCollection services)
⑤
        {
            services.AddSingleton<ISampleDocumentService, SampleDocumentService>();
        }

        #endregion
    }
}
```

① Задаёт описание расширения для Web-клиента, которое задано в текущей сборке.

② Создаёт новый экземпляр `LayoutWebClientExtension`. `serviceProvider` — сервис-провайдер.

③ Получить название расширения.

④ Получить версию расширения.

⑤ Регистрация типов в IoC контейнере.
Параметры:

- `containerBuilder` — сборщик контейнеров.

2. Демонстрирует расширение функционала с помощью добавления новых сервисов, контроллеров.
3. Реализован контроллер `SampleDocumentController` с методом `CreateOutgoingDocument`, который вызывает сервис `ISampleDocumentService`, для создания исходящего документа и заполнения его полей:

- "Исходящий.Дата создания" = текущая дата.
- "Исходящий.Регистратор" = текущий пользователь.
- "Исходящий.Автор" = текущий пользователь.
- "Исходящий.Организация-Контрагет" = Входящий.Организация-Отправитель.
- "Исходящий.Название" = "В ответ на " + <Входящий.Исходящий номер>.
- В ссылки Исходящего и исходного Входящего добавить ссылку друг на друга, тип "ответ" — "в ответ на".

```
namespace CreateCardServerExtension.Controllers
{
    public class SampleDocumentController : Controller
    {
        private readonly ICurrentObjectContextProvider currentObjectContextProvider;
        private readonly ISampleDocumentService sampleDocumentService;

        public SampleDocumentController(ICurrentObjectContextProvider
currentObjectContextProvider, ISampleDocumentService sampleDocumentService) ①
        {
            this.currentObjectContextProvider = currentObjectContextProvider;
            this.sampleDocumentService = sampleDocumentService;
        }

        public ActionResult CreateOutgoingDocument(Guid parentDocId) ②
        {
            var sessionContext = this.currentObjectContextProvider
.GetOrCreateCurrentSessionContext();
            var response = sampleDocumentService.CreateOutgoingDocument(
sessionContext, parentDocId);

            return Content(JsonHelper.SerializeToJson(response), "application/json");
        }
    }
}
```

① Создаёт новый экземпляр, см. `SampleDocumentController`.

② GET: `/LayoutCreateDocumentController/SendToAcquaintance`

Проект "CreateCardWebExtension"

Содержит клиентские скрипты, в которых при нажатии на кнопку с помощью сервиса `requestManager` отправляется запрос на сервер. Пользователю

отображается созданная карточка Исходящего документа в режиме просмотра.

Создание карточки пользовательского типа

Данный раздел содержит описание примера добавления в диалог создания карточки пользовательского типа.

Ссылка на пример на GitHub: CreateCardDialog.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Открыть `/Samples.sln`
2. Собрать проект `ServerExtensions > CreateCardDialog > CreateCardDialogServerExtension`
3. Открыть консоль в папке `ServerExtensions > CreateCardDialog > CreateCardDialogWebExtension` и выполнить команду `npm install`, затем `npm update` и в конце `npm run build:prod`
4. Скопировать каталог `SamplesOutput\Extensions\CreateCardDialogServerExtension` в каталог `Путь к сайту Web-клиента\Extensions`
5. Скопировать файл `SamplesOutput\Extensions\ru\CreateCardDialogServerExtension.resources.dll` в каталог `Путь к сайту Web-клиента\Extensions\ru`
6. Скопировать каталог `SamplesOutput\Content\Modules\CreateCardDialogWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустить конструктор разметок.
2. Выбрать библиотеку `ApprovalDesigner карточки`.
3. Выбрать тип *Этап согласования*.
4. Создать разметку создания и просмотра (Например, добавить элемент управления `Строка` и привязать его к свойству `MainInfo\Name`).

5. Выбрать условия использования этих разметок.
6. В конфигурационном файле Web-клиента `appsettings.json` в секции `Docsvision > Platform > CardTypes` добавить строку:

```
{
    {

        "CardTypeId": "0DB13C90-21B6-49D8-9070-8144DF97552A", ①
        "CssClass": "approval-stage"
    }
}
```

① `0DB13C90-21B6-49D8-9070-8144DF97552A` — идентификатор типа карточки этапа согласования.

7. Разрешить создание карточек вида *Этап согласования* в справочнике видов карточек.
8. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
9. Открыть Web-клиент и нажать на кнопку **Создать**.
10. Должен появиться новый тип карточки *Этап*, доступный для создания (фон — фиолетовый).
11. При выборе его, открывается созданная разметка, при этом фон боковой панели и шапки становится фиолетовым.

Проект "CreateCardDialogServerExtension"

Проект реализует серверное расширение, в котором регистрируется фабрика для создания типа карточки *Этап*, отличная от стандартной.

Проект "CreateCardDialogWebExtension"

Содержит клиентские скрипты реализующие веб-расширение, которое регистрирует тип карточки для его корректного отображения в Web-клиенте.

Примечания

В базовом варианте, чтобы подключить к Web-клиенту карточку нового типа, достаточно внести изменения в конфигурационный файл Web-клиента, как описано в инструкции проверки примера, а так же описать стиль класса `approval-stage-color`, например как:

```
.approval-stage-color {
  background-color: black;
}
```

Этот стиль применяется к элементам добавляемого типа карточки в диалоге создания карточки.

Если необходимо изменять цвет основных элементов управления Web-клиента, например, основной кнопки, шапки, боковых панелей, следует реализовать стили по примеру, описанному в `approval-stage.scss`.

Локализованные наименования всех зарегистрированных типов карточек доступны на клиенте и содержатся в объекте `resources`. Наименование пользовательского типа доступно по ключу `CardType_{CardTypeName}_DisplayName`, где `{CardTypeName}` - наименование типа карточки в файле метаданных.

В собственном решении или элементе управления можно использовать цвета, связанные с каким-то типом карточки. Для этого следует воспользоваться системными CSS классами:

- `card-type-background-color` — стандартный фон элемента типа карточки.
- `card-type-background-color-hover` — стандартный фон элемента типа карточки при наведении мыши.
- `card-type-background-color-light` — светлый фон элемента типа карточки.
- `card-type-background-color-light-hover` — светлый фон элемента типа карточки при наведении мыши.
- `card-type-background-color-disabled` — заблокированный фон элемента типа карточки, например, когда кнопка недоступна для нажатия.
- `card-type-foreground-color` — стандартный цвет шрифта типа карточки.

Список может быть расширен, при необходимости обратитесь к документации разработчика.

Если определенная бизнес-логика реализована или необходимо её реализовать, например если требуется задать значение какого-то поля по умолчанию при создании карточки, можно подключить серверное расширение (см. `LayoutWebClientExtension.cs`) и реализовать в нем интерфейс `ICardFactory`, в котором реализуется собственная логика по созданию карточки или созданию карточки по шаблону.

Фрагмент файла "LayoutWebClientExtension.cs"

```
namespace CreateCardDialogServerExtension
{

    public class LayoutWebClientExtension : WebClientExtension ①
    {

        public LayoutWebClientExtension(IServiceProvider serviceProvider) ②
            : base()
        {
        }

        public override string ExtensionName ③
        {
            get { return Assembly.GetAssembly(typeof(LayoutWebClientExtension)).
GetName().Name; }
        }

        public override Version ExtensionVersion ④
        {
            get { return new Version(FileVersionInfo.GetVersionInfo(Assembly
.GetExecutingAssembly().Location).FileVersion); }
        }


        #region WebClientExtension Overrides

        public override void InitializeServiceCollection(IServiceCollection services) ⑤
        {
            services.AddTransient<ICardLifeCycleEx, ApprovalStageCardLifeCycle>();
        }

        protected override List<ResourceManager> GetLayoutExtensionResourceManagers() ⑥
        {
            return new List<ResourceManager>
            {
                { Resources.ResourceManager }
            };
        }

        #endregion
    }
}
```

① Задаёт описание расширения для Web-клиента, которое задано в текущей сборке.

② Создаёт новый экземпляр `LayoutWebClientExtension`.

Параметры:

- `serviceProvider` — сервис-провайдер.

③ Получить название расширения.

④ Получить версию расширения.

⑤ Регистрация типов в IoC-контейнере.

Параметры:

- `containerBuilder` — сборщик контейнеров.

⑥ Получает менеджеры ресурсов для расширения разметки.

Добавление условия выбора разметок

Данный раздел содержит описание примера серверного расширения. Пример демонстрирует добавление условия выбора разметок по группе, пользователю, роли.

Ссылка на пример на GitHub: CustomConditionTypes.

После сохранения изменений в конструкторе разметок, разметка отображается в соответствии с заданными условиями.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проект `ServerExtensions > CustomConditionTypes > CustomConditionTypesDesignerExtension`.
3. Соберите проект `ServerExtensions > CustomConditionTypes > CustomConditionTypesLayoutExtension`.
4. Соберите проект `ServerExtensions > CustomConditionTypes > CustomConditionTypesInterfaces`.
5. Скопируйте каталог `SamplesOutput\Plugins\CustomConditionTypesDesignerExtension` в каталог `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins`.

6. Скопируйте каталог `SamplesOutput\Plugins\CustomConditionTypesInterfaces` в каталог `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins`.
7. Скопируйте каталог `SamplesOutput\Extensions\CustomConditionTypesInterfaces` в каталог `Путь к сайту Web-клиента\Extensions`.
8. Скопируйте каталог `SamplesOutput\Extensions\CustomConditionTypesLayoutExtension` в каталог `Путь к сайту Web-клиента\Extensions`.
9. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите Конструктор Web-разметок.
2. Выберите или создайте любую разметку в локации **WebFrame** › **MainMenu**.
3. Откройте вкладку условия использования разметки.
4. Добавьте новое условие выбора или измените существующее.
5. Измените в условии выбора роль, группу, пользователя или все сразу.
6. Запустите Web-клиент.
7. Заданная разметка отображается, только если выполняются условия выбора разметки.

Проект "CustomConditionTypesWebLayoutsDesignerExtension"

Проект-расширение для конструктора разметок. Содержит класс `CustomConditionTypesProvider`, в котором реализованы методы для отображения новых условий выбора разметки по группе, пользователю, роли.

```
    [SupportedOSPlatform("windows")]
    internal class CustomConditionTypesProvider : IConditionTypesProvider
    {
        public Guid Id => Constants.ExtensionId;

        public List<Staff> GetStaffTree() ①
        {
            var staffTree = new List<Staff>();

            var sessionContext = this.SessionContextProvider
.GetOrCreateCurrentSessionContext();

            var staffService = sessionContext.ObjectContext.GetService<IStaffService>();
            var units = staffService.GetUnits(null, true).OrderBy(u => u.Name);
```

```
            foreach (var unit in units)
            {
                var nodeStaff = unit.Staff;
                staffTree.Add(nodeStaff);
            }

            return staffTree;
        }

        public List<StaffEmployee> GetUsersTree() ②
        {
            var usersTree = new List<StaffEmployee>();

            var sessionContext = this.SessionContextProvider
.GetOrCreateCurrentSessionContext();
            var staffService = sessionContext.ObjectContext.GetService<IStaffService>();
            var units = staffService.GetUnits(null, true).OrderBy(u => u.Name);
            foreach (var unit in units)
            {
                foreach (var employee in unit.Employees)
                    usersTree.Add(employee);
            }

            return usersTree;
        }

        private readonly IServiceProvider serviceProvider;
        private ICurrentObjectContextProvider sessionContextProvider;

        public CustomConditionTypesProvider(IServiceProvider serviceProvider)
        {
            this.serviceProvider = serviceProvider;
        }

        private ICurrentObjectContextProvider SessionContextProvider
        {
            get
            {
                if (this.sessionContextProvider == null)
                    this.sessionContextProvider = DocsVision.Platform.Tools.LayoutEditor
.ServiceUtil.GetService<ICurrentObjectContextProvider>(this.serviceProvider);
                return this.sessionContextProvider;
            }
        }
```

① Получение списка штатов сотрудников.

② Получение списка сотрудников.

Проект "CustomConditionTypesLayoutExtension"

Проект-расширение для Web-клиента, которое позволяет определить, выполняется условие выбора разметки или нет для текущего контекста.

Фрагмент "CustomConditionTypesLayoutExtension.cs"

```
    public class CustomConditionTypesLayoutExtension : ILayoutExtension
    {
        public CustomConditionTypesLayoutExtension()
        {
            this.LayoutExtensionDescription = new LayoutExtensionDescription
            {
                Id = Constants.ExtensionId,
                Name = "Custom condition types layout extension",
                Version = FileVersionInfo.GetVersionInfo(Assembly.GetExecutingAssembly
().Location).FileVersion,
                ConditionTypes = this.GetConditionTypes(),
                ModeConditionTypes = this.GetLayoutTypes()
            };
        }
```

При внесении изменений необходимо увеличивать номер версии расширения в `AssemblyFileVersion`.

Проект "CustomConditionTypesInterfaces"

Проект-расширение для конструктора разметок и серверного расширения, содержащий все константные значения для этих проектов.

Фрагмент "Constants.cs"

```
    public static class Constants
    {
        public static readonly Guid ExtensionId = new Guid("7BDD6F65-B6F2-414F-B97F-
13E87E972F85");
        public static readonly Guid LocationExtensionId = new Guid("7e794753-46ed-42f3-
85b2-2e172c00d045");

        public static class ConditionTypes
        {
            public static readonly Guid GroupsConditionType = new Guid("AC83CC41-CB31-
4804-B330-A180FA02BA9D");
            public static readonly Guid UnitConditionType = new Guid("103B4D34-594A-479B-
BE72-D481677C5136");
```

```
            public static readonly Guid RolesConditionType = new Guid("986E19CD-6101-
4979-A99B-77CA50C0D781");
        }

        public static class MainMenu
        {
            public const string Id = "DV.MainMenu";
        }
    }
```

Взаимодействие с пользовательским типом карточки

Данный раздел содержит описание примера серверного расширения. Пример демонстрирует взаимодействие с пользовательским типом карточки Docsvision.

Ссылка на пример на GitHub: CustomLibrary.

В библиотеке `Sample custom library` реализован тип карточки `Custom Directory`, который представляет справочник с полем `Counter`. Значение поля получается скриптом и выводится во всплывающем окне.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.
- Docsvision Resource Kit
- Docsvision DVExplorer

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проекты `ServerExtensions > CustomLibrary > CustomLibrary.ObjectModel` и `CustomLibraryServerExtension`.
3. Откройте консоль в папке `ServerExtensions > CustomLibrary > CustomLibraryWebExtension` и выполнить команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
4. Скопируйте каталог `SamplesOutput\Extensions\CustomLibraryServerExtension` в каталог `Путь к сайту Web-клиента\Extensions`.
5. Скопируйте каталог `SamplesOutput\Content\Modules\CustomLibraryWebExtension` в

каталог `Путь к сайту Web-клиента\Content\Modules`.



Перед выполнением следующего шага, рекомендуется сделать резервную копию базы данных Docsvision.

6. На сервере Docsvision установите *Docsvision Resource Kit*, установить обновление *DVExplorer*. С помощью утилиты *DVCardManager* следует загрузить в БД Docsvision библиотеку `CustomCardLib` из каталога `SamplesOutput\SamplesCardDefs\CustomLibrary`. Подробное описание процедуры загрузки пользовательской библиотеки карточек содержится в руководстве разработчика Docsvision.
7. С помощью утилиты *DVExplorer* подключитесь к БД, откройте карточку справочника `Custom Directory`, добавьте строку секции `MainInfo` и установите значение `777` в поле `Counter`.

   Сохраните изменения. Подробное описание работы с утилитой *DVExplorer* содержится в документации комплекта утилит Resource Kit.
8. Перезапустите службу **dvappserver** для Linux и **Docsvision Platform** для Windows.
9. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустить конструктор разметок.
2. Скопировать любую разметку просмотра.
3. Выбрать условия использования этой разметки.
4. В разметке просмотра на событие `OnCardOpened` элемента `root` указать обработчик с названием `getCustomData`.
5. Сохранить разметку.
6. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
7. Создайте карточку документа.
8. При открытии разметки просмотра появится всплывающее сообщение: `Custom data: 777`.

Каталог "CardDefs"

Содержит описание библиотеки `Sample custom library`, в которой описан тип

карточки `CustomDirectory` — справочник. В секции `MainInfo` справочника есть поле `Counter`. Подробности содержатся в документации разработчика Docsvision.

Проект "CustomLibrary.ObjectModel"

Проект содержит объектную модель карточки типа `CustomDirectory`. Подробности содержатся в руководстве разработчика Docsvision.

```
namespace CustomLibrary.ObjectModel
{

    public class CustomDirectory : BaseCard ①
    {

        public static readonly ObjectProperty MainInfoProperty = ObjectProperty.Register
("MainInfo", typeof(ObjectCollection<MainInfo>), typeof(CustomDirectory)); ②

        [DebuggerHidden]
        static CustomDirectory()
        {
        }

        internal protected CustomDirectory() ③
            : base()
        {
        }

        internal protected CustomDirectory(ObjectInitializationData data) ④
            : base(data)
        {
        }

        #region Properties

        public MainInfo MainInfo ⑤
        {
            get
            {
                if (((ObjectCollection<MainInfo>)GetValue(MainInfoProperty)).Count == 0)
                    ((ObjectCollection<MainInfo>)GetValue(MainInfoProperty)).Add(new
MainInfo());

                return ((ObjectCollection<MainInfo>)GetValue(MainInfoProperty)).First();
            }
        }

        #endregion
```

```
    }
}
```

① Представляет собой пользовательский справочник.

② Настройки базы данных.

③ Создаёт новый экземпляр `CustomDirectory`.

④ Создаёт новый экземпляр `CustomDirectory`.

⑤ Главная информация.

Проект "CustomLibraryServerExtension"

Проект-расширение для Web-клиент. Содержит сервис по работе со справочником `CustomDirectory`, методы контроллера `CustomLibrary` для взаимодействия с клиентскими скриптами веб-приложения.

```
namespace CustomLibraryServerExtension.Services
{
    public class CustomLibraryService : ICustomLibraryService ①
    {
        public CustomLibraryService(IServiceProvider provider) ②
        {
        }
        public int GetCustomData(SessionContext sessionContext) ③
        {
            var settingsDirectory = sessionContext.ObjectContext.GetObject
<CustomDirectory>(CustomLibrary.CustomLibrary.CardLib.CardDefs.CustomDirectory.ID);

            int count = settingsDirectory.MainInfo.Counter;

            return count;
        }
    }
}
```

① Представляет собой пример сервиса, использующего кастомную библиотеку

② Создаёт новый экземпляр <see cref="CustomLibraryService"/>

③ Получить кастомные данные

Проект "CustomLibraryWebExtension"

Содержит клиентский скрипт `getCustomData`.

Информация о расширенной карточке

Данный раздел содержит описание примера разработки и подключения собственного серверного расширения для чтения значения полей, загруженных и не загруженных на страницу.

Ссылка на пример на GitHub: ExtendedCardInfo.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Собрать проект `ServerExtensions > ExtendedCardInfo > ExtendedCardInfoServerExtension`.
3. Откройте консоль в папке `ServerExtensions > ExtendedCardInfo > ExtendedCardInfoWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
4. Скопировать каталог `SamplesOutput\Extensions\ExtendedCardInfoServerExtension` в каталог `Путь к сайту Web-клиента\Extensions`.
5. Скопируйте файл `SamplesOutput\Extensions\ru\ExtendedCardInfoServerExtension.resources.dll` в каталог `Путь к сайту Web-клиента\Extensions\ru`.
6. Скопируйте файл `SamplesOutput\Extensions\uk\ExtendedCardInfoServerExtension.resources.dll` в каталог `Путь к сайту Web-клиента\Extensions\uk`.
7. Скопируйте каталог `SamplesOutput\Content\Modules\ExtendedCardInfoWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
8. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте любую разметку просмотра.

3. Выберите условия использования этой разметки.
4. Откройте разметку и для ЭУ `RegDate` на событие **`onChanged`** привязать функцию `extendedCardCheckDates`.



Для разных видов карточки название может варьироваться, например, `RegistrationDate`.

5. Сохраните разметку.
6. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
7. Откройте карточку с этой разметкой.
8. Убедитесь, что при изменении даты появляется сообщение.

Проект "ExtendedCardInfoServerExtension"

Проект-расширение для Web-клиента. Содержит бизнес-логику для чтения значения полей, загруженных и не загруженных на страницу.

Демонстрирует расширение функционала с помощью добавления новых сервисов, контроллеров. Реализован контроллер `ExtendedCardController` с методом `Get`, который вызывает сервис `IExtendedCardService` для чтения не загружаемых на страницу полей документа:

- `CreateDate`
- `ChangeDate`
- `Description`
- `BarCode`

Проект "ExtendedCardInfoWebExtension"

Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт функции `extendedCardCheckDates`, которая вызывается на событие **`onChanged`** ЭУ `RegDate`. С помощью `requestManager.get` отправляем запрос на сервер для получения расширенной модели `IExtendedCardModel` и сравниваем поля.

Проверка опции лицензии

Данный раздел содержит описание примера проверки наличия дополнительной опции лицензионного ключа.

Ссылка на пример на GitHub: LicenseCheck.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`
2. Соберите проект `ServerExtensions > LicenseCheck > LicenseCheckServerExtension`
3. Откройте консоль в папке `ServerExtensions > LicenseCheck > LicenseCheckWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
4. Скопируйте каталог `SamplesOutput\Content\Modules\LicenseCheckWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
5. Скопируйте каталог `SamplesOutput\Extensions\LicenseCheckServerExtension` в каталог `Путь к сайту Web-клиента\Extensions`.
6. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте любую разметку просмотра.
3. Выберите условия использования этой разметки.
4. Откройте разметку и добавить в нее элемент управления `Кнопка`.
5. На событие `onClick` задать функцию `checkLicenseFeature`.
6. Сохраните разметку.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
8. Откройте карточку с этой разметкой.
9. Убедитесь, что появился новый элемент управления (кнопка).
10. Нажмите на кнопку. Должно появиться сообщение с результатом проверки.

Проект "LicenseCheckServerExtension"

Проект-расширение для Web-клиента. Содержит бизнес-логику и скрипт для проверки наличия дополнительной опции лицензионного ключа. В данном

примере это опция `Docsvision Card Builder`. Демонстрирует расширение функционала с помощью добавления новых сервисов, контроллеров. Реализован контроллер `LicenseCheckController` с методом `CheckFeature`, который вызывает сервис `ILicenseCheckService`.

Проект "LicenseCheckWebExtension"

Содержит скрипт, который при нажатии на кнопку с помощью сервиса `requestManager` отправляется запрос на сервер. Пользователю отображается сообщение о наличии или отсутствии опции.

Автоматическое изменение срока исполнения заданий

Данный раздел содержит описание примера серверного расширения. Пример демонстрирует реализацию автоматических изменений данных связанных карточек.

Ссылка на пример на GitHub: ShiftTasksEndDate.

После сохранения карточки Документа ищутся все его связанные задания, и срок исполнения для отправленных на исполнение заданий переносится на 3 дня вперед.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проект `ServerExtensions > ShiftTasksEndDate > ShiftTasksEndDateServerExtension`.
3. Откройте консоль в папке `ServerExtensions > ShiftTasksEndDate > ShiftTasksEndDateWebExtension` и выполнить команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
4. Скопируйте каталог `SamplesOutput\Extensions\ShiftTasksEndDateServerExtension` в каталог `Путь к сайту Web-клиента\Extensions`.
5. Скопируйте файл `SamplesOutput\Extensions\ru\ShiftTasksEndDateServerExtension.resources.dll` в

каталог “Путь к сайту Web-клиента\Extensions\ru”.

6. Скопируйте файл `SamplesOutput\Extensions\uk\ShiftTasksEndDateServerExtension.resources.dll` в каталог `Путь к сайту Web-клиента\Extensions\uk`.
7. Скопируйте каталог`SamplesOutput\Content\Modules\ShiftTasksEndDateWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
8. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте любую разметку редактирования.
3. Выберите условия использования этой разметки.
4. В разметке редактирования на событие `OnCardSaved` элемента `root` прикрепляем обработчик с названием `shiftTasksEndDate`.
5. Сохраните разметку.
6. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
7. Создайте карточку документа.
8. Добавьте дочерние задания (не группы заданий), отправить на исполнение.
9. Нажмите на редактирование карточки, изменить что-нибудь, сохранить.
10. Срок исполнения для отправленных на исполнение заданий сдвинулся на 3 дня.

Проект "ShiftTasksEndDateServerExtension"

Проект-расширение для Web-клиента. Содержит бизнес-логику изменения данных связанных карточек. Демонстрирует расширение функционала с помощью добавления новых сервисов, контроллеров. Реализован контроллер `AdvancedDocumentController` с методом `ShiftTasksEndDate`, который вызывает сервис `IShiftTasksEndDateService`.

Проект "ShiftTasksEndDateWebExtension"

Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт функции `shiftTasksEndDate`, которая вызывается на событие `OnCardSaved`. С помощью `requestManager.post` отправляем запрос на сервер для изменения даты.

Отображение данных из связанных карточек

Данный раздел содержит описание примера серверного расширения. Пример содержит серверное расширение для отображения данных из связанных карточек в табличном контроле. В разметку карточки *Документ УД/Исходящий* добавляется таблица контрагентов, в колонках которой автоматически показываются e-mail и телефон выбранной в строке организации, загружаемые с сервера отдельным запросом.

Ссылка на пример на GitHub: TableControl.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проект `ServerExtensions > TableControl > TableControlServerExtension`.
3. Откройте консоль в папке `ServerExtensions > TableControl > TableControlWebExtension` и выполнить команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
4. Скопируйте каталог `SamplesOutput\Content\Modules\TableControlWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
5. Скопируйте каталог `SamplesOutput\Extensions\TableControlServerExtension` в каталог `Путь к сайту Web-клиента\Extensions`.
6. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте разметку просмотра исходящего документа.
3. Выберите условия использования этой разметки.
4. Откройте разметку и добавить в нее элемент управления `Таблица` в режиме *Редактирование по месту* и 3 столбца. Для таблицы в поле *Источник данных* секцию `Partners Receivers`.

5. В первый столбец поместите элемент управления `Подразделение котрагента` в режиме *Редактирование по месту*, задайте имя `samplePartnersDepartment`. Значение *Поле данных* выберите `Partner company`.
6. Во второй столбец поместить элемент управления `Метка`, задайте имя `sampleDepartmentEmail`.
7. В третий столбец поместить элемент управления `Метка`, задайте имя `sampleDepartmentPhone`.
8. На событие `После открытия карточки` элемента `root` задайте функцию `sampleDocumentViewCardOpened`.
9. Сохранить разметку.
10. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
11. Откройте карточку с этой разметкой, добавить строку в таблицу, выбрать контрагента, обновить страницу.
12. Убедиться, что почта и телефон контрагента заполнились.

Проект "TableControlServerExtension"

Проект-расширение для Web-клиента. Содержит бизнес-логику и скрипт для отображения данных из связанных карточек в табличном контроле. Демонстрирует расширение функционала с помощью добавления новых сервисов, контроллеров. Реализован контроллер `SamplePartnersController` с методом `GetPartnersInfo`, который вызывает сервис `ISamplePartnersService`, для получения информации о контрагентах: `Phone`, `Email`, `Name`.

Проект "TableControlWebExtension"

Содержит клиентский скрипт с функцией `sampleDocumentViewCardOpened`, которая вызывается на событие `После открытия карточки`. Функция отправляет запрос на сервер `getPartnersInfo` и заполняет таблицу вернувшимися данными.

Расширение возможностей программы Конструктор Web-разметок

Конструктор Web-разметок предоставляет возможность дополнять собственные функциональные возможности с помощью расширений.

Ссылка на пример на GitHub: Docsvision Web-layout designer extension template.

Альтернативно расширение можно получит

Расширение программы Конструктор Web-разметок может добавлять:

- Типы элементов управления.
- Свойства (настройки) элементов управления.
- Редакторы для свойств элементов управления.
- Функции проверки доступности операций.
- Критерии для условий выбора разметок.
- Локализованные ресурсы.

Разработка расширения программы Конструктор Web-разметок

Процесс разработки расширения программы Конструктор Web-разметок может включать следующие шаги:

1. Создание в Visual Studio[8] проекта типа TemplateDesignerExtension.
2. Обновление ссылок на подключаемые компоненты (при необходимости).



Версии подключаемых компонентов в проекте должны соответствовать версиям компонентов, используемых Конструктором web-разметок, для которого разрабатывается расширение. При обновлении версии Web-клиента все разработанные компоненты должны быть скомпилированы повторно с использованием новых версий библиотек Web-клиента.

3. Разработка ядра расширения программы Конструктор Web-разметок.

   Ядро расширения программы Конструктор Web-разметок представляет собой производный от `WebLayoutsDesignerExtension` класс. В нём переопределяются методы получения описателей типов элементов управления, описателей типов свойств элементов управления, редакторов значений свойств и пр., в зависимости от назначения расширения программы Конструктор Web-разметок.

   При создании проекта из шаблона `TemplateDesignerExtension` ядро расширения программы Конструктор Web-разметок будет содержаться в файле `Extension\DesignerExtension.cs`.

   Примеры реализации отдельных элементов ядра расширения программы Конструктор Web-разметок рассмотрены в следующих разделах.

4. Компиляция проекта.

> В результате будет получен файл DLL — расширение программы Конструктор Web-разметок, а также файлы DLL с локализованными ресурсами.

Публикация расширения программы Конструктор Web-разметок

Последовательность публикации расширения программы Конструктор Web-разметок на сервере Web-клиента включает следующие шаги:

1. Компиляция проекта.

   > В результате будет получен файл DLL с ядром расширения программы Конструктор Web-разметок и реализуемыми функциями, а также файлы DLL с локализованными ресурсами.

2. Создание каталога для Решения на сервере Web-клиента: `Каталог-установки-Конструктора-Web-Разметок/Plugins/%Каталог Решения%`.
3. Копирование сборки расширения программы Конструктор Web-разметок в `%Каталог Решения%`. Файлы DLL с локализованными ресурсами копируются в существующие в каталоге `/ru/` (для русской локализации), `/uk/` (для английской локализации) и т.д.

   > Реализованная в расширении программы Конструктор Web-разметок функциональность будет доступна после запуска программы Конструктор Web-разметок.

4. Перезапуск Конструктор Web-разметок.

Создание и публикация расширения программы Конструктор Web-разметок

Серверное расширение представляет собой библиотеку DLL, разработанную на платформе .NET.

Ссылка на шаблон на GitHub: TemplateDesignerExtension.

Чтобы создать расширение программы Конструктор Web-разметок:

1. Создайте в Visual Studio[9] новый проект с типом TemplateDesignerExtension.
2. Обновите в проекте ссылки на зависимости.



После крупного обновления версии Docsvision, необходимо повторно скомпилировать расширение с новыми версиями компонентов. В остальных случаях такой необходимости нет.

Это необходимо, чтобы версии подключаемых компонентов соответствовали версиям компонентов, используемых сервером Web-клиента, для которого разрабатывается расширение.

3. Реализуйте в проекте функции расширения программы Конструктор Web-разметок.



Примеры кода расширений программы Конструктор Web-разметок приведены в следующих пунктах раздела, а также в репозитории "ДоксВижн".

4. Настройте ядро расширения.

Ядром расширения является класс, производный от `WebLayoutsDesignerExtension`, который содержится в файле `Extension/DesignerExtension.cs`.

```
class TemplateDesignerExtension : WebLayoutsDesignerExtension
{
public TemplateDesignerExtension(IServiceProvider provider)
: base(provider)
{
}
  protected override List<ControlTypeDescription> GetControlTypeDescriptions() ①
  {
      return new List<ControlTypeDescription>
      {
      };
  }

  protected override Dictionary<string, PropertyDescription>
GetPropertyDescriptions() ②
  {
      return new Dictionary<string, PropertyDescription>
      {
      };
  }

  protected override List<ResourceManager> GetResourceManagers() ③
  {
```

```
        return new List<ResourceManager>
        {
          Resources.ResourceManager
        };
    }
}
```

① Метод регистрации новых типов элементов управления.

② Метод регистрации новых свойств элементов управления.

③ Метод регистрации локализованных ресурсов.

Добавление нового элемента управления

Добавить новый тип элементов управления в Конструктор Web-разметок можно с помощью расширения.

Расширение предоставляет возможность настраивать разметку с использованием нового элемента управления, но не обеспечивает его функционирование на клиентской стороне.

Полностью вопрос разработки новых типов элементов управления будет рассмотрен в разделе Разработка элементов управления.

Основным способом описания элемента управления является текстовый или XML. Подробнее о создании текстового описателя элемента управления см. [WebClient:new-controls-create-text-descriptor].

Добавить элемент управления

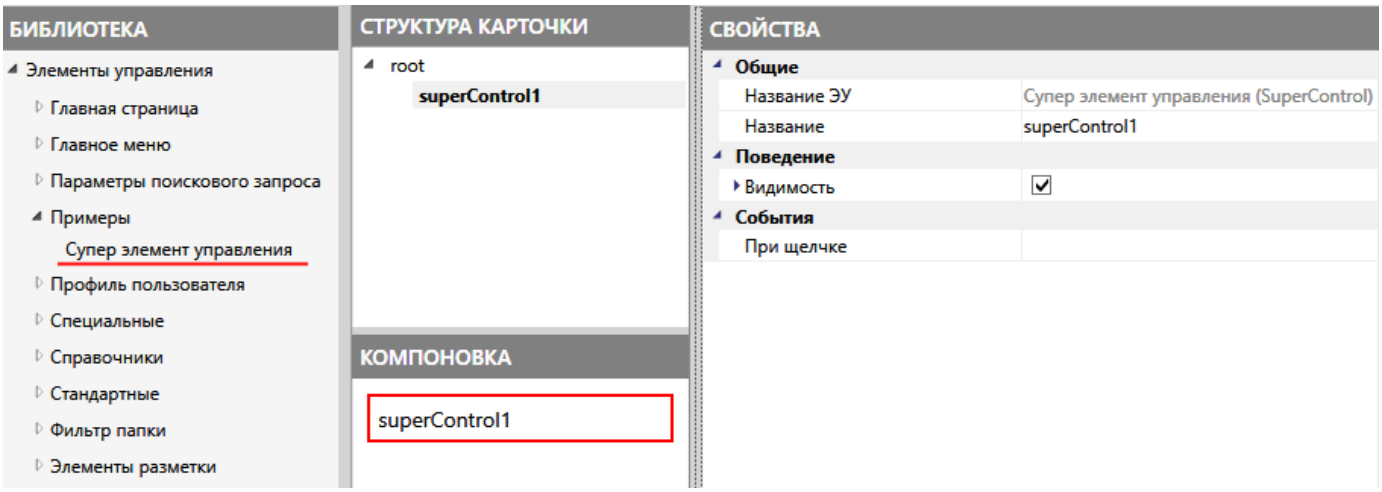


Рисунок 706. Элемент управления из расширения в редакторе разметок

Чтобы создать расширение, добавляющее новый тип элементов управления, выполните приведенную далее инструкцию.

1. Создайте проект расширения программы Конструктор Web-разметок.
2. Добавьте в ядро расширения (файл `DesignerExtension.cs`) функцию, предоставляющую *описатель элемента управления*.

   Описатель элемента управления — это объект, содержащий информацию об элементе управления: название типа, группа, перечень свойства (настроек). Описатель имеет тип `ControlTypeDescription`.

```
ControlTypeDescription GetSuperControlTypeDescription ()
{
  var controlTypeDescription = new ControlTypeDescription ("SuperControl") {
      DisplayName = "Супер элемент управления",
      ControlGroupDisplayName = "Примеры",
      PropertyDescriptions = {
          PropertyFactory.GetNameProperty(), ①
          PropertyFactory.GetVisibilityProperty(), ②
          PropertyFactory.GetClickEvent() ③
      }
  };
  return controlTypeDescription;
}
```

   ① Название.

   ② Видимость.

   ③ Событие `При щелчке`.

   При составлении описания элемента необходимо:

   i. Указать название типа элемента управления. Название передается непосредственно в конструктор класса `ControlTypeDescription`.

   Описатель элемента управления с типом "SuperControl"

```
var controlTypeDescription = new ControlTypeDescription ("SuperControl"){}
```

   ii. Указать отображаемое название типа элемента управления в DisplayName. Данное название приводится в строке `Название ЭУ` в блоке настроек элемента управления в программы Конструктор Web-разметок.

```
DisplayName = "Супер элемент управления",
```

iii. Указать группу элемента управления в ControlGroupDisplayName. Группа используется для группировки элементов управления в библиотеке элементов управления программы Конструктор Web-разметок.

```
ControlGroupDisplayName = "Примеры",
```

3. Укажите свойства элемента управления в `PropertyDescriptions`.

> Свойства элемента управления — его параметры, которые определяют внешний вид, связь с данными карточки, функции, вызываемые при работе элемента управления (обработчики событий). К свойствам относятся: видимость, текстовая метка, подсказка, (событие) при получении фокуса и др. Значения свойств (есть исключения) передаются в клиент, где могут быть использованы для формирования пользовательского интерфейса.

4. Переопределите метод `GetControlTypeDescriptions()` базового класса, чтобы он возвращал коллекцию с реализованным в пункте 2 методом:

```
protected override List<ControlTypeDescription> GetControlTypeDescriptions()
{
    return new List<ControlTypeDescription>
    {
        GetSuperControlTypeDescription()
    };
}
```

Добавление свойства в Конструктор Web-разметок

Данный раздел содержит описание примера добавления свойства `Url` в конструктор разметок. Проект-расширение для конструктора разметок. Содержит добавление собственного свойства, используя класс `PropretyDescription` (см. свойство `Url`).

Ссылка на пример на GitHub: ControlProperties/Url.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Открыть `/Samples.sln`.
2. Собрать проект `ControlProperties > Url > UrlPropertyDesignerExtension`.
3. Скопировать каталог `SamplesOutput\Plugins\UrlPropertyDesignerExtension` в каталог “Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins”.

Проверка примера

Для проверки примера необходима реализация ЭУ, который использует свойство Url (xml-файл или расширение конструктора)

1. Реализуйте и подключите ЭУ, который использует свойство `Url` одним из доступных способов:
   - Через `xml`, см. `Controls\Link\SampleLinkControlDescription.xml`
   - Через расширение, см. `Controls > Image > ImageDesignerExtension`.
2. Запустите конструктор разметок.
3. Скопируйте любую разметку.
4. Выберите условия использования этой разметки.
5. Откройте разметку и добавить в нее новый элемент управления.
6. Убедитесь, что присутствует свойство `Url`.
7. Сохраните разметку.
8. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
9. Откройте карточку с этой разметкой.
10. Убедитесь в работоспособности нового элемента управления.

Добавление нового свойства элементов управления

Ссылка на GitHub: TemplateDesignerExtension.

Элементы управления обладают набором свойств, которые определяют их поведение в веб-браузере пользователя: внешний вид, получение данных (для

разметки карточки), обработка событий.

Когда программист формирует список свойств элемента управления нового типа, он может ссылаться только на существующие стандартные свойства (Стандартные свойства и события элементов управления). Если стандартные свойства не позволяют описать все особенности элемента управления, программист может сформировать новое свойство.

Добавить новое свойство элементов управления в Конструктор Web-разметок можно с помощью *Расширения*.

Перечень необходимых инструментов

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.
- Docsvision WebClient SDK.

Добавление свойства

Чтобы создать расширение, добавляющее новый тип свойств элементов управления, выполните приведенную далее инструкцию.

1. Убедитесь, что установлен Docsvision WebClient SDK.

   В этом случае в переменной окружения `DocsvisionWebClientSDK` содержится путь к его папке, например, `C:\Program Files\Docsvision\WebClient\SDK`.

2. Создайте проект расширения программы Конструктор Web-разметок.
3. Добавьте в ядро расширения (файл `DesignerExtension.cs`) функцию, предоставляющую *описатель свойства* элемента управления.

   Описатель свойства элемента управления — это объект с типом `PropertyDescription`, содержащий информацию о свойстве: название свойства, тип, категория, редактор, который должен использоваться для изменения значения, возможность локализации и настройки значения для определённого типа устройств.

   ```
   PropertyDescription GetSuperPropertyDescription()
   {
   ```

```
    var propertyDescription = new PropertyDescription();
    propertyDescription.Name = "SuperProperty";
    propertyDescription.DisplayName = "Супер свойство";
    propertyDescription.Type = typeof(string); ①
    propertyDescription.Category = PropertyCategoryConstants.DataCategory; ②
    return propertyDescription;
}
```

① Строковый тип свойства.

② Указывает категорию свойства (`DataCategory` — Данные).

При составлении описания элемента необходимо:

i. Указать название свойства в Name.

```
propertyDescription.Name = "SuperProperty";
```

ii. Указать отображаемое название свойства в DisplayName. С данным названием свойство отображается в списке свойств элемента управления в программы Конструктор Web-разметок.

```
+propertyDescription.DisplayName = "Супер свойство";
```

iii. Указать тип значения свойства в Type.

```
propertyDescription.Type = typeof(string); ①
```

① Строковый тип свойства.

iv. Указать категорию свойства в Category. Категория используется для группировки свойств в настройках элемента управления в программе Конструктор Web-разметок. Категория выбирается из перечня в классе `PropertyCategoryConstants`.

```
propertyDescription.Category = PropertyCategoryConstants.DataCategory; ①
```

① Указывает категорию свойства (`DataCategory` — Данные).



Обратитесь к описанию класса `PropertyDescription`, чтобы

получить полный перечень возможностей описателя свойства элементов управления.

4. Переопределите метод `GetPropertyDescriptions` базового класса, чтобы он возвращал коллекцию с парой: ключ — имя. Возвращаемая коллекция позволяет получить описатель данного свойства. Используйте имя, указанное в `propertyDescription.Name` и метод, предоставляющий описатель свойства в качестве значения:

```
protected override Dictionary<string, PropertyDescription> GetPropertyDescriptions()
{
    return new Dictionary<string, PropertyDescription>
    {
        { "SuperProperty", GetSuperPropertyDescription()}
    };
}
```

Проверка свойства

Чтобы проверить пример:

1. Создайте новый тип элемента управления.
2. Добавьте **Супер свойство** в список свойств элемента управления (в `PropertyDescriptions`).



Для получения описателя свойства используйте метод `PropertyFactory.Create`, передав в него имя свойства, указанное при его регистрации в методе `GetPropertyDescriptions`.

```
PropertyDescriptions = {
    PropertyFactory.GetNameProperty(),
    PropertyFactory.GetVisibilityProperty(),
    PropertyFactory.Create("SuperProperty"), ①

    PropertyFactory.GetClickEvent() ②
}
```

① Получаем описатель свойства "SuperProperty".

② Событие "При щелчке".

3. Опубликуйте расширение с элементом управления на сервере Web-клиента.

4. Откройте для настройки любую разметку карточки.
5. Добавьте в разметку элемент управления с новым свойством. В его настройках будет *Супер свойство*.

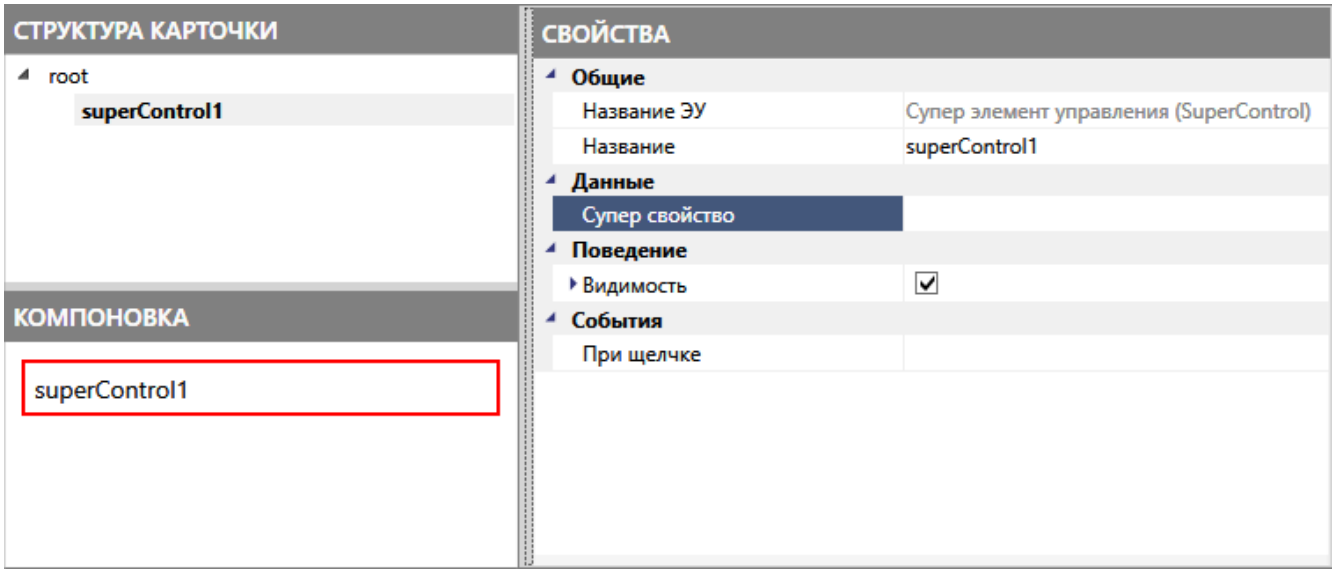


Рисунок 707. Свойство элемента управления из расширения в редакторе разметок

Добавление редактора для значения свойства элемента управления

Редактор предоставляет пользовательский интерфейс для изменения значения свойства в программе Конструктор Web-разметок.

Новый редактор нужно разрабатывать, когда требуется поддержать возможность изменения свойства со сложным типом значения, предоставить новый пользовательский интерфейс для изменения свойства или изменить логику работы со значением свойства.

Редактор представляет собой элемент управления с типом `System.Windows.Controls.UserControl`, в котором реализован интерфейс `Xceed.Wpf.Toolkit.PropertyGrid.Editors.ITypeEditor`. Рекомендуемые размеры элемента управления: `20x200px`. Если для изменения значения свойства требуется форма большего размера, разработайте её в отдельном компоненте с типом `System.Windows.Window`, а в основной компонент `UserControl` добавьте функцию, вызывающую данную форму.

Добавить новый редактор

Добавить новый редактор в Конструктор Web-разметок можно с помощью *Расширения*.

1. Создайте проект расширения программы Конструктор Web-разметок.
2. Добавьте в проект компонент с типом UserControl (WPF) и реализуйте в нем программный интерфейс ITypeEditor.

   Следующий код демонстрирует пример класса редактора.

```
public partial class SuperEditor : UserControl, ITypeEditor
{
    private string startValue;
    private PropertyItem propertyItem; ①

    public SuperEditor()
    {
        InitializeComponent();
    }

    public FrameworkElement ResolveEditor(PropertyItem prop) ②
    {
        propertyItem = prop; ③

        if (prop.Value == null) ④
            prop.Value = string.Empty;

        startValue = prop.Value.ToString(); ⑤

        Placeholder.Text = "{lines ...}"; ⑥

        return this; ⑦
    }

    private void ChangeLines_Click(object sender, RoutedEventArgs e) ⑧
    {
        var dialog = new DialogEditor(startValue);
        dialog.ShowDialog();

        propertyItem.Value = startValue = dialog.Value; ⑨
    }
}
```

   ① Переменные для хранения текущего значения свойства и его программного компонента (PropertyItem).

   ② Реализация единственного метода интерфейса ITypeEditor. Метод должен вернуть элемент управления, который будет отображаться на панели Свойства.

③ Сохраняем ссылку на компонент свойства, для возможности сохранения изменения свойства.

④ Поле `prop.Value` содержит значение элемента управления.

⑤ Значение свойства передается в `prop.Value`.

⑥ Текст, который будет отображаться в качестве значения свойства.

⑦ В качестве элемента управления для отображения на панели свойства используется текущий элемент управления.

⑧ Открытие диалогового окна редактирования значения свойства.

⑨ Новое значение получаем из переменной `Value`. Значение должно быть сохранено в `propertyItem.Value`.
В данном случае редактор представляет собой кнопку **ChangeLines**, при нажатии которой открывается диалоговое окно **DialogEditor** для ввода текстового значения, а также нередактируемый блок текста **Placeholder** для отображения статичного значения `{lines ···}`.

Ниже приведён код графической части редактора.

```
 <UserControl x:Class="TemplateDesignerExtension1.Editors.SuperEditor"
              xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
              xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
              xmlns:mc="http://schemas.openxmlformats.org/markup-
compatibility/2006"
              xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
              xmlns:local="clr-namespace:TemplateDesignerExtension1.Editors"
              mc:Ignorable="d"
              d:DesignHeight="20" d:DesignWidth="200">
        <Grid>
            <TextBlock x:Name="ValueString" HorizontalAlignment="Left" Margin="1"
VerticalAlignment="Stretch" Width="164"/>
            <Button x:Name="ChangeLines" Content="···" HorizontalAlignment="Right"
Margin="1" VerticalAlignment="Top" Width="22" Click="ChangeLines_Click" />;
        </Grid>
</UserControl>
```

Собственная реализация редактора может включать любые элементы управления и использоваться для отображения любых значений (с учётом рекомендуемых значений элемента управления).



Значение свойства можно получить из переменной `propertyItem.Value`. При изменении значения, его нужно

> сохранить в `propertyItem.Value`.

3. Переопределите метод `GetEditors` базового класса расширения, чтобы он возвращал коллекцию с парой: ключ — имя. Используя данный метод, можно получить редактор. Значением будет тип редактора:

```
protected override Dictionary<string, Type> GetEditors()
{
    return new Dictionary<string, Type>
    {
        { "SuperEditor", typeof(Editors.SuperEditor) }
    };
}
```

Проверка примера

1. Создайте новое свойство элемента управления в расширении с *Супер редактором*.
2. Укажите *Супер редактор* в качестве редактора свойства в параметре **Editor**.

```
PropertyDescription GetSuperPropertyDescription()
{
    var propertyDescription = new PropertyDescription();
    propertyDescription.Name = "SuperProperty";
    propertyDescription.Type = typeof(string); ①
    propertyDescription.Category = PropertyCategoryConstants.DataCategory;
    propertyDescription.DisplayName = "Супер свойство";
    propertyDescription.Editor = typeof(SuperEditor); ②
    return propertyDescription;
}
```

① Тип свойства должен быть string!

② Указываем тип "Супер редактора".

3. Добавьте свойство с редактором *Супер редактор* в описатель элемента управления. См. пример в пункте Добавление нового свойства элементов управления.
4. Опубликуйте расширение с элементом управления на сервере Web-клиента.
5. Откройте для настройки любую разметку.

6. Добавьте в разметку элемент управления, содержащий свойство с редактором. Для изменения значения свойства будет использован *Супер редактор*.

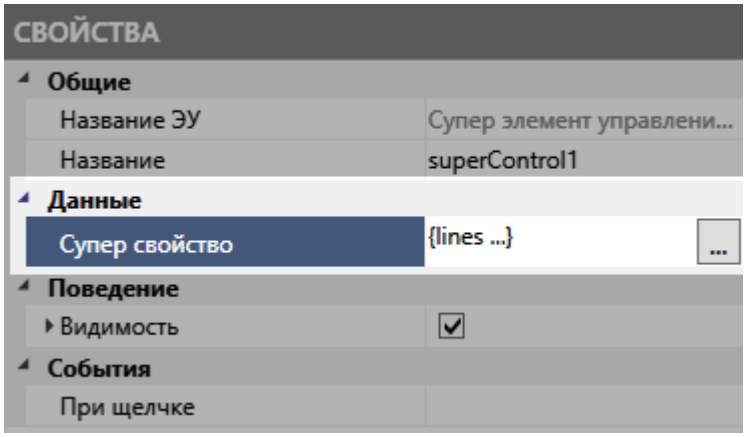


Рисунок 708. Супер свойство с собственным редактором

Изменение значения свойства осуществляется в диалоговом окне, открываемом при нажатии кнопки **...** (три точки).

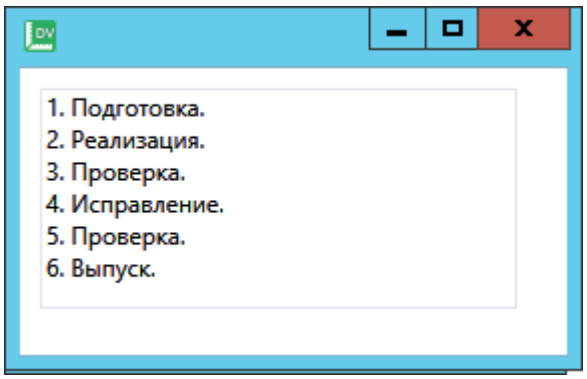


Рисунок 709. Диалоговое окно изменения значения свойства

Редактор для выбора поля карточки

Для выбора поля карточки, с которым связаны данные элемента управления, может использоваться стандартный редактор — `Docsvision.BackOffice.WebLayoutsDesigner.Editors.FieldMetadataEditor`. Данный редактор предоставляет возможность установки фильтра для ограничения доступных для выбора полей.

Следующий пример демонстрирует редактор, предоставляющий возможность выбирать только поля булева типа.

```
class BooleanMetadataEditor : ITypeEditor
{
    public FrameworkElement ResolveEditor(PropertyItem propertyItem)
```

```
    {
        var editor = new FieldMetadataEditor();

        editor.FieldFilter = (field) => { return field.FieldType == FieldType.Bool; }; ①
        return editor.ResolveEditor(propertyItem);
    }
}
```

① Устанавливаем ограничением для отображаемых типов. Текущее обрабатываемое поле передаётся в `field`.

Использование возможностей фильтра `FieldFilter` позволяет установить более точные критерии для отображения списка полей в редакторе.

Например, в следующем коде поля ограничиваются по типу и связанной карточки и секции, с которой должно быть связано поле (для полей ссылающихся на справочники).

```
public FrameworkElement ResolveEditor(PropertyItem propertyItem)
{
    var editor = new FieldMetadataEditor();

    editor.FieldFilter = (field) => { ①
        return (field.FieldType == FieldType.RefId || field.FieldType == FieldType
.UniqueId) &&
            field.LinkedCardTypeId == RefBaseUniversal.ID && ②
            field.LinkedSectionId == RefBaseUniversal.Items.ID; }; ③

    return editor.ResolveEditor(propertyItem);
}
```

① Устанавливаем фильтр.

② Поле должно быть связано с карточкой Конструктор справочников

③ Поле должно быть связано с секцией *Строка справочника*

Добавление локализованных ресурсов

В строковых переменных расширения программы Конструктор Web-разметок (отображаемые названия свойств, диалоги) можно использовать локализованные ресурсы, добавляемые расширением.

Чтобы создать расширение, добавляющее новые локализованные ресурсы, выполните приведенную далее инструкцию.

1. Создайте проект расширения программы Конструктор Web-разметок.
2. Добавьте в проект расширения ресурсные файлы (для английского, русского и др. языков), содержащие требуемые локализованные ресурсы.



Проект расширения, формируемый из шаблона `TemplateDesignerExtension`, содержит ресурсные файлы, которые можно использовать для формирования локализованных ресурсов.

3. Переопределите метод `GetResourceManagers` базового класса расширения, чтобы он предоставлял менеджер ресурсов для добавленных ресурсных файлов:

```
protected override List<ResourceManager> GetResourceManagers()
{
    return new List<ResourceManager>
    {
        Resources.ResourceManager
    };
}
```

4. Скомпилируйте проект и скопируйте полученную сборку на сервер Web-клиента в папку `Каталог-установки-Конструктора-Web-Разметок/Plugins/%Каталог Решения%`. Ресурсные сборки скопируйте в папки `Каталог-установки-Web-клиента/ru/` (для русской локализации), `Каталог-установки-Web-клиента/uk/` (для английской локализации) и т.д.

Ограничение выполнения действий с элементом управления

Программист может установить запрет на выполнение определённых действий с элементом управления в программе Конструктор Web-разметок.

Ограничение устанавливается "функцией проверки", которая возвращает набор флагов, разрешающих только определённые действия с элементом управления. Например, *функция проверки* может ограничить использование элемента управления определённым типом карточек.

Добавление функции проверки

Чтобы создать расширение, добавляющее функцию проверки, выполните приведенную далее инструкцию.

1. Создайте проект расширения программы Конструктор Web-разметок.

2. Добавьте в ядро расширения (файл DesignerExtension.cs) функцию проверки следующего вида:

```
public static AllowedOperationsFlag GetPossibleOperation(IServiceProvider
serviceProvider)
{
    var flags = AllowedOperationsFlag.All; ①

    return flags;
}
```

① По умолчанию доступны все операции
Функция проверки должна вернуть набор флагов AllowedOperationsFlag.

3. Добавьте в код функции проверки алгоритм, добавляющий нужные ограничения.

Следующий код демонстрирует пример функции проверки, запрещающей добавление элемента управления в разметку карточки, если тип карточки не *Документ*.

```
public static AllowedOperationsFlag GetPossibleOperation(IServiceProvider
serviceProvider)
{
    var flags = AllowedOperationsFlag.All; ①

    var selectedLayoutService = ServiceUtil.GetService<ISelectedLayoutService
>(serviceProvider); ②

    if (selectedLayoutService.SelectedCardTypeId != DocsVision.BackOffice.CardLib
.CardDefs.CardDocument.ID) ③
        flags &= ~AllowedOperationsFlag.Create; ④

    return flags;
}
```

① По умолчанию доступны все операции.

② Получение сервиса для работы с разметкой.

③ Из сервиса получаем идентификатор типа карточек, для которых настраивается разметка.

④ Если тип карточки не документ (CardDocument.ID) из возможных операций

удаляется флаг Create. Что фактически приводит к невозможности добавить элемент управления в разметку
В примере информация о контексте использования элемента управления предоставляется сервисом ISelectedLayoutService.

4. Переопределите метод GetAllowedOperations базового класса расширения, чтобы он возвращал коллекцию с парой: ключ — имя. Метод позволяет получить функцию проверки. Значение представлено функцией проверки:

```
protected override Dictionary<string, Func<AllowedOperationsFlag>>
GetAllowedOperations()
{
    return new Dictionary<string, Func<AllowedOperationsFlag>>
    {
        {"GetPossibleOperation", () => GetPossibleOperation(this.serviceProvider) } ①
    };
}
```

① Обратите внимание, что в функцию передается сервис-провайдер расширения.

Проверка примера

1. Создайте проект нового элемента управления.
2. В поле GetAllowedOperations (функция проверки) описателя элемента управления укажите функцию GetPossibleOperation.

```
ControlTypeDescription GetSuperControlTypeDescription()
{
    var controlTypeDescription = new ControlTypeDescription("SuperControl")
    {
        DisplayName = "Супер элемент управления",
        ControlGroupDisplayName = "Примеры",
        PropertyDescriptions = {
            PropertyFactory.GetNameProperty (),
            PropertyFactory.GetVisibilityProperty (),
            PropertyFactory.GetClickEvent ()
        },
        GetAllowedOperations = () => GetPossibleOperation(serviceProvider) ①
    };

    return controlTypeDescription;
```

```
}
```

① Передаем функцию проверки.
Функция проверки также может быть получена по имени из хранилища функций проверки:

```
GetAllowedOperations = AllowedOperationsStorage.GetAllowedOperations
("GetPossibleOperation")
```

Код получения хранилища функций проверки:

```
IAllowedOperationsStorage allowedOperationsStorage;
protected IAllowedOperationsStorage AllowedOperationsStorage
{
    get
    {
        return this.allowedOperationsStorage ?? (this.allowedOperationsStorage =
ServiceUtil.GetService<IAllowedOperationsStorage>(serviceProvider));
    }
}
```

3. Опубликуйте расширение с элементом управления на сервере Web-клиента.
4. Откройте для настройки разметку карточки *Документ*.

> Элемент управления `Супер элемент управления` будет представлен в библиотеке элементов управления.

5. Откройте любую другую разметку.

> Элемент управления `Супер элемент управления` будет отсутствовать в библиотеке элементов управления.

Разработка элементов управления

Элемент управления в Web-клиенте — дополнительный компонент Модуля, предназначенный для настройки разметки. Элемент управления включает два обязательных компонента:

1. Описатель элемента управления — добавляет в Web-клиент возможность использования данного элемента управления для настройки разметки в программе Конструктор Web-разметок.
2. Клиентский компонент элемента управления — добавляет в Web-клиент пользовательский интерфейс и функции элемента управления: внутреннюю логику работы, публичные функции.

Формирование и публикация описателя элемента управления

Описатель элемента управления — программный компонент, содержащий общую информацию об элементе управления, и список его свойств (настроек, которые можно изменять в программе Конструктор Web-разметок).

Описатель элемента управления может быть сформирован двумя способами:

1. В виде функции расширения программы Конструктор Web-разметок.
2. В виде файла XML.

Для простоты первый вариант описателя будем называть *бинарным описателем* элемента управления, второй — *текстовым описателем*.

Основное отличие: *текстовый описатель* можно составить с применением уже зарегистрированных редакторов, свойств (пример исключения приведён в пункте Формирование текстового описателя элемента управления) и функций проверки. Для *бинарного описателя* недостающие элементы можно включить в расширение программы Конструктор Web-разметок, в котором реализуется описатель. В остальном выбор способа формирования описателя определяется предпочтениями программиста.

Формирование бинарного описателя элемента управления

Обратитесь к инструкции в пункте [WebClient:layout-designer-add-new-control], чтобы ознакомиться с условиями формирования бинарного описателя элемента управления.

Формирование текстового описателя элемента управления

Ссылка на пример на GitHub: TemplateXmlDesignerExtension.

1. Создайте текстовый файл с расширением `.xml`, кодировкой `UTF-8`, и содержимым:

```
<?xml version="1.0" encoding="utf-8" ?>
```

```
<Controls>
  <Control Name="" DisplayName="" ControlGroupDisplayName=""> ①
    <Properties> ②
       <Property Type="" > ③
    </Properties>

    <Events> ④
       <Event Name="" ResourceKey=""> ⑤
    </Events>
  </Control>
</Controls>
```

① Описатель элемента управления.

② Свойства элемента управления.

③ Здесь перечисляются все свойства элемента управления.

④ События элемента управления.

⑤ Здесь перечисляются все события элемента управления.

2. Укажите основную информацию об элементе управления в атрибутах элемента `Control`.

Name

Название типа элемента управления — английский, без спецсимволов и пробелов.

DisplayName

Отображаемое в программе Конструктор Web-разметок название элемента управления. Если нужно использовать локализованное название, замените атрибут `DisplayName` на `ResourceKey`, и укажите в значении название локализованного ресурса.

Добавить локализованные ресурсы можно с помощью предназначенного для этого расширения программы Конструктор Web-разметок.

ControlGroupDisplayName

Название группы, в которой будет отображаться элемент управления на панели *Библиотека*. Можно указать название существующей группы или новой группы.

Если нужно использовать локализованное название, замените атрибут

`ControlGroupDisplayName` на `ControlGroupResourceKey`, и укажите в значении название локализованного ресурса.

AllowedOperations

Название функции проверки. Необязательный атрибут.

3. Перечислите свойства элемента управления в элементе `Properties`.

   Свойства обеспечивают возможность настройки элемента управления в программе Конструктор Web-разметок, и передаются в клиент при загрузке.

4. Добавьте дочерний элемент `Property`.
5. Укажите информацию о свойстве в аргументах элемента `Property`.

Type

Название типа свойства, зарегистрированного в программе Конструктор Web-разметок.

Типы стандартных свойств перечислены в пункте Стандартные свойства и события элементов управления.

DefaultValue

Значение по умолчанию (необязательное).

6. Перечислите события элемента управления в элементе `Events`.
7. Добавьте дочерний элемент `Event`.
8. Укажите информацию о событии в аргументах элемента `Event`.

Name

Название события. Названия стандартных событий перечислены в пункте Стандартные свойства и события элементов управления.

DisplayName

Отображаемое название события. Если нужно использовать локализованное название, замените атрибут `DisplayName` на `ResourceKey`, и укажите в значении название локализованного ресурса.

DefaultValue

Обработчик по умолчанию (необязательное).

1. Сохраните файл `.xml` на сервер Web-клиента в папку `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins`.

9. Перезапустите Конструктор Web-разметок.

Добавление нового типа свойств в текстовом описателе

Новый тип свойств можно добавить с помощью серверного расширения (см. Добавление нового свойства элементов управления) или описать непосредственно в текстовом описателе элемента управления.

Добавление нового типа свойства

Чтобы добавить новый тип свойств:

1. Добавьте дочерний элемент `Property` в секцию `Properties` текстового описателя.
2. Добавьте в элемент `Property` аргументы:

Name

Название нового типа свойств.

DataType

Тип значения свойств. В `DataType` может быть указан только стандартный тип — должно быть указано полное название типа (например, `System.Boolean`).

ResourceKey

Название локализованного ресурса для отображаемого названия.

Category

Категория свойства. Может иметь значения: `GeneralCategory`, `DataCategory`, `BehaviorCategory`, `AppearanceCategory`, `EventsCategory`, `DefaultValueCategory`.

Editor

Имя редактора значения, которое указано при его регистрации в методе GetEditors расширения программы Конструктор Web-разметок.

BindingConverter

Имя конвертера в серверном расширении для свойства `Type="Binding"`.

BindingResolver

Имя резолвера биндинга в серверном расширении для свойства `Type="Binding"`.

ItemsSource

Варианты значений свойства, которые будут отображаться в виде раскрывающегося списка.

В значении атрибута должно быть указано полное имя типа класса, реализующего интерфейс `IItemsSource` (см. ниже).

DefaultValue

Значение по умолчанию.

DeviceDependent

Флаг, указывающий на то, что значение свойства зависит от типа клиентского устройства.

Localized

Флаг, указывающий на то, что свойство является локализуемым.

При объявлении свойств можно указывать одновременно `Type` и другие параметры. Свойство, указанное в `Type`, станет основой для нового, которое может быть дополнительно настроено.

Следующий пример демонстрирует объявление свойства `superBinding`, основанного на стандартном свойстве `Binding`, с переопределённым конвертером (`BindingConverter`):

```
<Property Type="Binding" Name="superBinding" BindingConverter="SuperConverter"
/>
```

Примеры объявления свойств

- Свойство, получающее значения из источника значений (`ItemsSource`):

```
<Property Name="SuperMode" DisplayName="Супер свойство с вариантами"
    Category="AppearanceCategory" DataType="System.Int32" DefaultValue="0"
    ItemsSource="SuperExtension.SuperVariantsSource" />
```

Пример реализации класса `SuperExtension.SuperVariantsSource`, предоставляющего значения для свойства:

```
namespace SuperExtension
{
    public class SuperVariantsSource : IItemsSource
    {
        public ItemCollection GetValues()
        {
            ItemCollection values = new ItemCollection();
            values.Add(0, "Вариант 1"); ①
            values.Add(1, "Вариант 2"); ②
            values.Add(2, Resource.LocalizedVariant3); ③
            return values;
        }
    }
}
```

① При выборе этого варианта, значение свойства будет 0.

② При этом варианте, значение 1.

③ Имя варианта можно объявить в ресурсах с целью локализации.

- Свойство логического типа, зависящее от устройства (`DeviceDependent`), с категорией `Внешний вид` (`AppearanceCategory`) и значением по умолчанию ***false***:

```
<Property Name="ShowPreview" ResourceKey="History_ShowPreview"
    Category="AppearanceCategory" DataType="System.Boolean" DefaultValue="false"
DeviceDependent="true" />
```

- Свойство числового типа с категорией `Поведение`(`BehaviorCategory`) и значением по умолчанию `20`:

```
<Property Name="RecordsOnPage" Editor="PositiveIntegerEditor"
    Category="BehaviorCategory" DataType="System.Int32" DefaultValue="20"/>
```

- Свойство строкового типа:

```
<Property Name="DurationControlName" Category="BehaviorCategory"
DataType="System.String"/>
```

- Локализованное свойство (`Localized`) строкового типа со стандартным редактором (`Editor`) `LocalizedPropertyEditor`:

```
<Property Name="ButtonText" Editor="LocalizedPropertyEditor" Localized="true"
    Category="AppearanceCategory" DataType="System.String"/>
```

Разработка и публикация клиентского компонента элемента управления

Клиентский компонент элемента управления — программный компонент, обеспечивающий загрузку и функционирование элемента управления на стороне клиента. Клиентский компонент предоставляет графический интерфейс элемента управления.

Клиентский компонент элемента управления является вариантом клиентского расширения.

Чтобы создать клиентский компонент элемента управления, выполните приведенную далее инструкцию.

1. Создайте проект клиентского расширения. Не публикуйте его.
2. Создайте в папке `src` файл TypeScript (`.ts`, `.tsx`) с кодом клиентского компонента.

   Клиентский компонент состоит из трёх блоков кода:

   i. Класс параметров:

```
export class SuperControlParams extends BaseControlParams {
    @r firstProperty?: string;
    @rw secondProperty?: boolean;

    @apiEvent firstEvent?: BasicApiEvent<IEventArgs>;
}
```

> Класс параметров является классом, расширяющим `BaseControlParams`. В данном классе перечисляются публичные свойства и события элемента управления, указанные в его описателе, которые

отсутствуют в базовом классе `BaseControlParams`. В данном классе также перечисляются любые другие параметры, значение которых должно быть доступным извне.

Если свойство, указанное в описателе, не требуется на клиентском уровне, его можно не объявлять в классе параметров.

В приведенном выше коде объявляются два свойства и событие: `firstProperty` (строкового типа), `secondProperty` (булева типа) и `firstEvent` (событие).

При объявлении свойства или события нужно указать декоратор:

- `@r` — если свойство доступно только для чтения.
- `@rw` — если свойство доступно для чтения и записи.
- `@apiEvent` — если свойство является событием.

Значение свойства будут загружено в поле, только если поле класса связано со свойством элемента управления.

Чтобы поле класса могло быть связано со свойством элемента управления:

☑ Имя поля класса должно совпадать с названием свойства/события элемента управления с условием.

Первая буква в названии поля класса должна быть строчной. Например, если в описателе свойство называется `SomeParam`, то поле должно назваться `someParam`.

☑ Тип поля класса должен совпадать с типом свойства элемента управления.

Для событий должен быть установлен тип:

- `BasicApiEvent<IEventArgs>` — для непрерываемых событий.
- `CancelableApiEvent<IEventArgs>` — для прерываемых событий.

Если в описателе элемента управления у свойства указано значение по умолчанию, данное значение должно быть указано в **классе параметров**.

ii. Интерфейс состояния:

```
export interface SuperControlState extends SuperControlParams, BaseControlState {
internalFirstState: boolean;
}
```

Интерфейс состояния является интерфейсом, расширяющим `BaseControlParams`.

В данном интерфейсе перечисляются свойства, которые требуются для описания внутреннего состояния элемента управления (частные переменные).

Если при работе элемента управления хранить внутреннее состояние не требуется, интерфейс состояния не будет содержать полей.

```
export interface SuperControlState extends SuperControlParams,
BaseControlState {
}
```

iii. Класс реализации:

```
export class SuperControlImpl extends BaseControlImpl<SuperControlParams,
SuperControlState> {
    constructor(props: SuperControlParams, state: SuperControlState) { ①
        super(props, state);

        this.state.firstEvent = SimpleEvent.Create(this.state.wrapper);
    }

    renderControl() { ②
        return (
            <div />
        );
    }
}
```

① Инициализация компонента.

② VirtualDOM элемента управления.

Класс реализации является производным от контейнерного класса `BaseControlImpl`, в который передаются `класс параметров` и `интерфейс`

состояния. Класс реализации возвращает виртуальный DOM, описывающий пользовательский интерфейс элемента управления.

Класс реализации содержит два обязательных метода:

constructor

В данном методе должны быть выполнены следующие действия:

- Вызван конструктор базового класса с передачей ему публичных и внутренних свойств ЭУ:

```
super(props, state);
```

- События, объявленные в классе параметров, должны быть проинициализированы, а внутренние обработчики событий связаны с текущим контекстом. Подробности в пункте Работа с событиями в клиентском компоненте.

renderControl

Метод должен вернуть VirtualDOM элемента управления.

Пользовательский интерфейс элемента управления может быть сформирован с использован стандартных элементов React, стандартных элементов управления Web-клиента или разработанных элементов управления (см. дополнительные условия в пункте [Использование вложенных элементов управления](UseNestedControls.md)).

iv. Интерфейсный класс:

```
export class SuperControl extends BaseControl<SuperControlParams,
SuperControlState> {
    protected createParams() {
        return new SuperControlParams(); ①
    }

    protected createImpl() { ②
        return new SuperControlImpl(this.props, this.state);
    }
}
```

① Инициализация параметров.

② Инициализация класса реализации.

> Интерфейсный класс является производным от контейнерного класса `BaseControl`, в который передаются *класс параметров* и *интерфейс состояния*.
>
> Интерфейсный класс обеспечивает жизненный цикл элемента управления: связывает элемент управления с его значением, инициализирует класс параметров, а также предоставляет проинициализированную сущность класса реализации.
>
> *Интерфейсный класс должен содержать два обязательных метода:*
>
> - `createParams` — метод должен вернуть экземпляр *класса параметров*;
> - `createImpl` — метод должен вернуть экземпляр *класса реализации*.
>
> Помимо указанных методов, интерфейсный класс может содержать функции, обрабатывающие загрузку значения свойства элемента управления с сервера Web-клиента.
>
> ```
> @handler("binding")
> private set Binding(binding: IBindingResult<boolean>) {
>     this.state.readOnly = editOperationAvailable(this.state.services,
> binding) == false;
> }
> ```

В случае простого элемента управления, лишенного сложной внутренней логики, реализация элемента управления может быть помещена в интерфейсный класс.

```
export class SuperSimpleControl extends BaseControl<SuperControlParams,
SuperControlState> {
    protected createParams() {
        return new SuperControlParams();
    }

    protected createImpl() { ①
        return new ControlImpl(this.props, this.state, this.renderControl.bind(this));
    }
```

```
    renderControl() { ②
        return (
            <div />
        );
    }
}
```

① Метод `createImpl` в данном случае возвращает экземпляр стандартного типа `ControlImpl`, в которые передана функция, возвращающая VirtualDOM элемента управления

② Метод может иметь любое название, кроме "render".

3. Измените содержимое файла `Index.ts`.

```
import { extensionManager } from "@docsvision/webclient/System/ExtensionManager";

import { SuperControl } from "./Controls/SuperControl"; ①

extensionManager.registerExtension({ ②
    name: "Client extension with controls",
    version: "1.0",
    controls: [{ controlTypeName: "SuperControl", constructor: SuperControl }] ③ ④
})
```

① Добавьте строку импорта элемента управления.

② Добавьте в `registerExtension` поле `controls`, в значении которого укажите описание элемента управления.

③ `controlTypeName` — название типа элемента управления, которое указано в описателе элемента управления

④ `constructor` — интерфейсный класс элемента управления.

4. Соберите проект клиентского расширения командой `npm run build`.
5. Скопируйте полученный файл `%BuildDir%\extension.js` на сервер Web-клиента в папку `/lib/docsvision/webclient/Content/Modules/%Каталог Решения%`.
6. После завершения отладки, соберите и опубликуйте на сервере Web-клиента "релизную версию" клиентского расширения. "Релизная версия" собирается командой `npm run build:prod`.

Объявление событий элемента управления в клиентском компоненте

Чтобы корректно объявить обрабатываемое событие в клиентском компоненте элемента

управления, нужно выполнить два условия:

☑ Генерируемые элементом управления события должны быть проинициализированы.

☑ Событие должно быть сгенерировано.

Инициализация событий

Для примера рассмотрим инициализацию события `onShowed`, поддерживаемого элементом управления:

```
export class SuperControlParams extends BaseControlParams { ①
    @apiEvent onShowed?: BasicApiEvent<IEventArgs>;
}
```

① Класс параметров с объявлением события `onShowed`.

Для инициализации события `onShowed`, нужно добавить в конструктор класса реализации следующий код:

```
this.state.onShowed = SimpleEvent.Create<IEventArgs>(this.state.wrapper);
```

Событие `onShowed` является непрерываемым (имеет тип `BasicApiEvent`). Для прерываемого события, инициализация будет выглядеть следующим образом:

```
this.state.onShowing = CancelableEvent.Create<IEventArgs>(this.state.wrapper);
```

Таким образом конструктор класса реализации может выглядеть так:

```
construct() {
    super.construct();

    this.state.onShowed = SimpleEvent.Create<IEventArgs>(this.state.wrapper);
    this.state.onShowing = CancelableEvent.Create<IEventArgs>(this.state.wrapper);
}
```

Генерация события

```
SimpleEvent.cast(this.state.onShowed).trigger({} as IEventArgs);

await CancelableEvent.cast(this.state.onShowing).trigger({} as IEventArgs).promise;
```

Связать обработчики с контекстом

Связывание обработчика с контекстом выполняется в конструктор класса реализации, что демонстрируется в следующем коде, в котором выполняется связывание с контекстом обработчика события `onClick`.

```
construct() {
    super.construct();

    this.clickHandler = this.clickHandler.bind(this); ①
}
```

① Метод `clickHandler` является внутренним обработчиком события `onClick`.

Работа элемента управления с данными карточки

Чтобы элемент управления мог работать с данными карточки, его необходимо определённым образом подготовить:

1. Добавьте в описатель элемента управления свойства: `DataSource`, `DataField` и `Binding`.

   > Первые два свойства предоставляют возможность выбрать в настройках элемента управления секцию (`DataSource`) и поле карточки (`DataField`) с данными. Свойство `биндинг` (`Binding`) — не отображаемое в настройках свойство, с которым значение элемента управления передается в его клиентский компонент.
   >
   > Элемент управления также может получать данные из карточки, связанной с текущей. Для возможности выбора типа источника (текущая карточки или определённый дополнительный источник) в описатель также нужно добавить свойство `ExtendedDataSource`.

   Текстовый описатель:

   ```
   ...
       <Properties>
           ...
           <Property Type="DataSource" />
           <Property Type="DataField" />
           <Property Type="Binding" />
           <Property Type="ExtendedDataSource" /> ①
   ```

```
    </Properties>
```

① Не обязательно.

Бинарный описатель:

```
...
    PropertyDescriptions =
    {
        ...
        PropertyFactory.GetDataSourceProperty(),
        PropertyFactory.GetDataFieldProperty(),
        PropertyFactory.GetBindingProperty(),
        PropertyFactory.GetExtendedDataSourceProperty(),
    }
```

2. Добавить в клиентский компонент элемента управления в класс параметров поле для хранения значения элемента управления. Тип поля должен соответствовать типу значения элемента управления.

```
export class SomeControlParams extends BaseControlParams {
    @rw value?: string;
}
```

3. Добавить в клиентский компонент элемента управления в интерфейс состояния поле для хранения биндинга. Тип поля должен быть `IBindingResult<string>`. Тип `string` нужно заменить на тип значения элемента управления.

```
export interface SomeControlState extends SomeControlParams, BaseControlState {
    binding: IBindingResult<string>; ①
}
```

① Если элемент управления предназначен только для чтения значения (изменять не требуется), свойство `binding` не требуется.

4. Добавить в клиентский компонент элемента управления в интерфейсный класс set-метод, загружающий значение элемента управления. Метод должен быть помечен декоратором `@handler("binding")`.

```
@handler("binding")
private set binding(binding: IBindingResult<string>) {
```

```
    this.state.value =  binding && binding.value;
    this.state.binding = binding;
}
```

5. Добавить в клиентский компонент элемента управления в интерфейсный класс метод, возвращающий биндинги.

   "Биндинг" формируется методом `getBindingResult`, в который нужно передать:

   - Текущий "биндинг" (из свойства класса, в которое "биндинг" был сохранен при получении с сервера.
   - Текущее значение элемента управления.
   - Поле, в котором содержится ассоциированное с ЭУ название элемента карточки.

   Переданное значение будет записано в историю изменений карточки: `Выполнилась операция "Редактирование". Значение поля {Значение аргумента}` было изменено с ⋯ на ⋯.

   ```
   protected getBindings() {
       return [getBindingResult(this.state.binding, this.params.value, () => at
   (SomeControlParams).controlTitle)];
   }
   ```

   Метод `getBindings` должен вернуть `IBindingResult`, для получения которого вызывается метод `getBindingResult`, принимающий:

   - Биндинг, полученный при загрузке элемента управления.
   - Новое значение элемента управления (в примере, оно хранится в поле `this.params.value`).
   - Поле, в котором содержится ассоциированное с элементом управления название элемента карточки. Данное значение используется при формировании истории изменений карточки: `Выполнилась операция "Редактирование". Значение поля {Значение аргумента}` было изменено с ⋯ на ⋯.

6. В класс реализации добавить функцию, сохраняющую значение элемента

управления в поле value.

```
changedHandler = (event) => {
    this.state.value = event.target.value;
    this.forceUpdate();
}
```

В составе биндинга может быть передана операция редактирования, настраиваемая в программе Конструктор Web-разметок. Операцию редактирования можно использовать, чтобы ограничить возможность изменять значение элемента управления или его скрытия.

Чтобы использовать операцию редактирования, необходимо:

☑ Добавить в описатель элемента управления свойство EditOperation:

```
<Properties>
①
        <Property Type="DataSource" />
        <Property Type="DataField" />
        <Property Type="Binding" />

        <Property Type="EditOperation" /> ②
</Properties>
```

① Свойства, перечисленные ранее, также должны быть указаны.

② Добавляем операцию редактирования.

☑ Добавить код проверки операции редактирования в функцию загрузки значения элемента управления (binding):

```
@handler("binding")
protected set binding(binding: IBindingResult<boolean>) {
    this.state.canEdit = !binding || editOperations.available(binding.
editOperation);
}
```

Проверка прав пользователя на изменение значения

Возможность изменения значения ЭУ (или выполнения других операций) проверяется по доступности пользователю операции редактирования.

Чтобы добавить возможность настраивать для ЭУ операцию редактирования,

нужно добавить в описатель элемента управления свойство EditOperation.

```
<?xml version="1.0" encoding="utf-8" ?>
<Controls>
    <Control Name="Somecontrol" DisplayName="Somecontrol">
        <Properties>
            <Property Type="Name" />

            <Property Type="DataSource" /> ①
            <Property Type="DataField" />
            <Property Type="Binding" />
            <Property Type="EditOperation" /> ②
        </Properties>
    </Control>
</Controls>
```

① Свойства, используемые для связывания ЭУ с данными карточки.

② Свойство Операция редактирования.

Свойство Операция редактирования передается в представительную часть в биндинге, поэтому свойство Binding также должно быть в описателе элемента управления.

Чтобы добавить проверку операции редактирования в представительную часть ЭУ, нужно:

1. Добавить в "интерфейс состояния" поле, в которое будет сохраняться результат проверки.
2. В "интерфейсном классе" в методе, отвечающем за загрузку биндинга, организовать проверку операции редактирования:

```
@handler("binding")
protected set binding(binding: IBindingResult<boolean>) {
    this.state.canEdit = !binding || this.props.services.editOperations.available
(binding.editOperation) ①
}
```

① Сохраняем результат проверки доступности операции в поле canEdit.



В приведенном коде проверка доступности операции осуществляется с помощью клиентского сервиса $EditOperationStore. Подробности в пункте Получение сервисов в клиентском компоненте.

Конвертация значения элемента управления

Программист может внедрить в механизм получения значения элемента управления функцию конвертации, которая требуемым образом подготовит данные для использования в клиентском компоненте элемента управления.

Чтобы внедрить функцию конвертации, нужно:

1. Указать в описателе свойства `Binding` название функции конвертации:

   ☑ в текстовом описателе — в атрибуте `BindingConverter`:

   ```
   <Property Type="Binding" BindingConverter="SuperConverter" />
   ```

   ☑ в бинарном описателе — в методе `GetBindingLoader`:

   ```
   var bindingProperty = PropertyFactory.GetBindingProperty();
   bindingProperty.GetBindingLoader = (() => new SimpleBindingLoader
   {
       ConverterType = "SuperConverter"
   });
   ```

2. Добавить класс конвертер в серверное расширение.

   Класс конвертера является производным от `BaseBindingConverter`, в котором переопределяется метод `ConvertForDisplay`. Данный метод должен привести значение элемента управления к требуемому виду.

   Следующий код демонстрирует пример класса конвертера, в котором значение элемента управления преобразуется с использованием функции `ConvertForDisplay`.

   ```
   internal class SuperConverter : BaseBindingConverter
   {
       public SuperConverter(IServiceProvider serviceProvider) ①
           : base(serviceProvider, "SuperConverter")
       {
       }

       public override BindingResult ConvertForDisplay(ControlContext controlContext,
   Binding binding, BindingResult bindingResult) ②
       {
           var data = SuperMethod(bindingResult.Value); ③
   ```

```
        return bindingResult.Clone(data); ④
    }
}
```

① В конструктор базового класса должно быть передано название конвертера, которое указано данных свойства.

② Метод `ConvertForDisplay` должен возвращать преобразованные данные свойства ЭУ для использования в клиентском расширении ЭУ.

③ Здесь метод `SuperMethod` требуемым образом преобразует текущее переданное значение ЭУ — `bindingResult.Value`.

④ Метод `Clone` создаёт копию `bindingResult`, в поле `Value` которой записывается преобразованное значение из `data`.
Конвертер вызывается автоматически при загрузке значения элемента управления, при условии совпадения названия конвертера в описателе и в расширении.

Обратите внимание

При разработке конвертера следует помнить следующую особенность: в конвертер всегда поступают данные текущей карточки. Если элемент управления связан с данными связанной карточки (не текущей), то получить её данные можно с помощью методов:

- `LayoutContextHelper.TryGetCardDataSource` — сохраняет в `cardId` идентификатор карточки, с данными которой связан элемент управления.
- `LayoutContextHelper.TryGetDataSource` — сохраняет идентификаторы метаданных карточки, с которыми связан элемент управления: в `cardId` — идентификатор карточки, в `sectionId` — идентификатор секции, в `rowId` — идентификатор строки.

```
public override BindingResult ConvertForDisplay(ControlContext controlContext,
Binding binding, BindingResult bindingResult)
{
   CardKindModel model = null;
if (LayoutContextHelper.TryGetCardDataSource(controlContext, binding, out var
cardId))
   {
         model = ExSuperMethod(controlContext.LayoutContext.SessionContext,
cardId);
```

```
    }
    return bindingResult.Clone(model);
}
```

3. Конвертер нужно добавить в IoC-контейнер в ядре серверного расширения:

```
public override void InitializeContainer(ContainerBuilder containerBuilder) {
    containerBuilder.RegisterOrderedType<SuperConverter, IBindingConverter>();
}
```

Расширение должно быть опубликовано на сервере Web-клиента.

Пример. Элемент управления "SuperControl"

В данном примере будут разработаны компоненты элемента управления `SuperControl`: текстовый описатель элемента управления и клиентский компонент.

Элемент управления `SuperControl` является ссылкой, адрес которой указывается в программе Конструктор Web-разметок. Для элемента управления доступно свойство **Видимость** и событие **При щелчке**.

1. Формируем текстовый описатель элемента управления.

   i. Создаем новый файл `SuperControl.xml` со следующим содержимым:

```
<?xml version="1.0" encoding="utf-8" ?>
<Controls>
    <Control Name="SuperControl" DisplayName="Супер элемент управления">
        <Properties>
            <Property Type="Name" /> ①
            <Property Type="Visibility" /> ②

            <Property Name="Url" Category="BehaviorCategory" DataType=
"System.String" /> ③
        </Properties>
        <Events>
            <Event Name="Click" ResourceKey="ControlTypes_ClickEventProperty"/>  ④
        </Events>
    </Control>
</Controls>
```

① Название.

② Видимость.

③ Свойство строкового типа для указания адреса ссылки.

④ Событие **При щелчке**.

ii. Файл `SuperControl.xml` копируем в папку `Каталог-установки-Конструктора-Web-Разметок/Plugins/%Каталог-Решения%`.

2. Формируем клиентское расширение с клиентским компонентом элемента управления.

   i. Создаем новое клиентское расширение.

   ii. Добавляем в папку расширения файл `src/Controls/SuperControl.tsx` со следующим содержимым:

```
import { BaseControlParams, BaseControlState, BaseControl } from
"@docsvision/webclient/System/BaseControl";
import { r } from "@docsvision/webclient/System/Readonly";
import { apiEvent } from "@docsvision/webclient/System/Event";
import { BasicApiEvent } from "@docsvision/webclient/System/ApiEvent";
import { IEventArgs } from "@docsvision/webclient/System/IEventArgs";
import { BaseControlImpl } from "@docsvision/webclient/System/BaseControlImpl";
import { SimpleEvent } from "@docsvision/webclient/System/SimpleEvent";
import React from "react"; ①

export class SuperControlParams extends BaseControlParams { ②
    @r url?: string; ③
    @apiEvent onClick?: BasicApiEvent<IEventArgs>;
}

export interface SuperControlState extends SuperControlParams, BaseControlState {
④
}

export class SuperControlImpl extends BaseControlImpl<SuperControlParams,
SuperControlState> { ⑤
    construct() {
        super.construct();

        this.state.onClick = SimpleEvent.Create<IEventArgs>(this.state.wrapper); ⑥

        this.clickHandler = this.clickHandler.bind(this); ⑦
    }
```

```
    clickHandler() { ⑧
        SimpleEvent.cast(this.state.onClick).trigger(); ⑨
        window.open(this.props.url, "_blank");
    }

    renderControl() { ⑩

        if (this.props.visibility) { ⑪
            return (
                <a onClick={this.clickHandler}>{this.props.url}</a> ⑫
            );
        }
    }
}

export class SuperControl extends BaseControl<SuperControlParams,
SuperControlState> {
    protected createParams() { ⑬
        return new SuperControlParams();
    }

    protected createImpl() {
        return new SuperControlImpl(this.props, this.state);
    }
}
```

① Импорт используемых модулей.

② Класс параметров. Объявляем отсутствующее в `BaseControlParams` свойство `url` и непрерываемое событие `onClick`.

③ Название свойства `url` здесь должно быть с маленькой буквы!

④ Интерфейс состояния. Внутренние состояния не используются.

⑤ Класс реализации.

⑥ Инициализируем событие `onClick`.

⑦ Связываем внутренний обработчик с событием `clickHandler`.

⑧ При нажатии ссылки будет выполнен переход по ней, а также вызван обработчик события `onClick`.

⑨ Вызываем обработчик события `onClick`, который указан в Конструкторе web-разметок.

⑩ Возвращаем VirtualDOM элемента управления.

⑪ Если элемент управления невидим (`Visibility` в значении `FALSE`), то

ничего не возвращаем.

⑫ При нажатии ссылки вызываем внутренний обработчик `clickHandler`

⑬ Интерфейсный класс.

3. Настраиваем клиентское расширение в файле `src/Index.tsx`.

```
import { extensionManager } from "@docsvision/webclient/System/ExtensionManager";
import { SuperControl } from "./Controls/SuperControl";

extensionManager.registerExtension({
    name: "Client extension with SuperControl",
    version: "1.0",
    controls: [{ controlTypeName: "SuperControl", constructor: SuperControl }]
})
```

4. Собираем клиентское расширение с помощью команды `npm run build`.
5. Полученный файл `extension.js` копируем в папку `{wcd/Content/Modules/SuperControlExtension/` на сервере Web-клиента.
6. Следующие действия выполняются после выполнения всех действий:
   i. Перезапустить службу Сервера Web-клиента, чтобы загрузились скрипты с элементом управления.
   ii. Откройте Конструктор Web-разметок и настроить разметку с использованием `Супер элемента управления`.

При настройке элемента управления должны быть доступны свойства **`Url`** и **`Видимость`**, а также событие **`При щелчке`**, для которого можно настроить обработчик.

В веб-браузере `Супер элемент управления` представляет собой ссылку (тег `a`), при нажатии на которую будет выполнен переход по адресу, указанному в **`Url`**, а также вызван обработчик, указанный в событии **`При щелчке`**.

Групповая операция подписания

В данном разделе описан пример реализации групповой операции *Подписание документа*.

Ссылка на пример на GitHub: DocumentSignBatchOperation.

Описание файлов проекта

1. `copy-path.js` — константы с именем расширения и параметрами сборки (в частности, путь, куда копировать файлы после сборки).
2. `gulpfile.js` — конфигурация сборки стилей для Gulp
3. `package.json` — файл конфигурации NPM. Содержит определение команд сборки, а также используемые npm-зависимости.
4. `package-lock.json` — файл, который автоматически генерируется NPM при выполнении команды `npm install`. Это служебный файл NPM, он необходим для того, чтобы зависимости всегда устанавливались в одной и той же последовательности.
5. `rollup.config.js` — файл конфигурации Rollup, который используется для сборки файлов скриптов в один бандл.
6. `tsconfig.json` — файл конфигурации TypeScript
7. `node_modules` — служебная папка NPM, которая появляется после выполнения команды `npm install`. Содержит npm-зависимости проекта, используемые в процессе сборки.
8. `src` — исходные файлы расширения

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Начало работы

1. Переименуйте расширение. Например, новое имя будет `MyWebExtension` (замените на уникальное имя своего расширения).

   Для этого:

   - Переименуйте саму папку `TemplateWebExtension` на `MyWebExtension`.
   - Откройте файл `copy-path.js` и замените значение `EXTENSION_NAME` на `MyWebExtension`.
   - Проверьте, что путь `SITE_ROOT` указывает на то место, куда должны копироваться файлы расширения после сборки. Может быть указан только относительный путь (относительно самого файла).
2. Откройте командную строку в корне проекта, выполнить `npm install`. В результате появится папка `node_modules` со всеми необходимыми

зависимостями.

3. В командной строке выполните `npm run build`.

   После выполнения команды по пути `SITE_ROOT` (из файла `copy-path.js`) будут подготовлены файлы и папки, которые необходимо скопировать в Каталог установки Web-клиента. В частности:

   - Файл скриптов (`Content/Modules/MyWebExtension/extension.js`).
   - Файл стилей (`Content/Modules/MyWebExtension/extension.css`).

     Файлы в данной папке подключаются автоматически в алфавитном порядке.

Сборка и установка

1. Откройте консоль в папке `DocumentSignBatchOperationWebExtension` и выполните последовательно команды:

   ```
   npm install
   npm update
   npm run build:prod
   ```

2. Скопируйте каталог `SamplesOutput\Content\Modules\DocumentSignBatchOperationWebExtension` в каталог `Каталог-установки-Web-клиента\Content\Modules`.
3. Скопируйте файл расширения программы Конструктор Web-разметок из папки `DocumentSignBatchOperationDesignerExtension` в папку `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins` в каталоге Web-клиента.
4. Перезапустите службу **dvappserver** для Linux и **Docsvision Platform** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Импортируйте решение из файла `DocumentSignBatchOperation.xml`.
3. Сделайте решение `DocumentSignBatchOperation` активным для локации *Папка*. Проверить, что в ЭУ `FolderGridBatchOperationsNode` разметки присутствует дочерний ЭУ `DocumentSignBatchOperation`.
4. Перезапустите службу **dvappserver** для Linux и **Docsvision Platform** для

Windows.

5. Откройте Справочник сотрудников в Web-клиенте и назначьте пользователю, которым планируется подписание, сертификат по умолчанию.
6. Авторизуйтесь в Web-клиенте под именем пользователя из предыдущего пункта.

   Убедитесь на любой тестовой карточке документа, что подписание сертификатом из предыдущего пункта проходит корректно.

7. Откройте папку с документами в представлении *Дайджест*. Выделите при помощи флагов документы для подписания. На панели групповых операций нажмите кнопку **Подписать документ**.
8. При необходимости просмотрите подписываемые документы, нажатием на ссылки карточек в открывшемся окне подтверждения. Нажмите **OK**.

   В случае успешного подписания всех документов будет выведено сообщение об успешном подписании.

9. Проверьте на нескольких документах, что подписание прошло корректно.

Чтобы ссылки на карточки отображались корректно в представлении, отличном от *Дайджест*, требуется настроить свойство *Столбцы представления для презентации* ЭУ `DocumentSignBatchOperation` в программе Конструктор Web-разметок. Через запятую укажите `columnName` столбцов представления, выбранных для отображения.



Если в выбор документов в таблице попадет документ, недоступный для подписания по состоянию или настройкам, появится сообщение об ошибке, документ не будет подписан. Для остальных документов процесс подписания будет продолжен.

Дальнейшая разработка

Новые исходные файлы следует добавлять в папку `src`. Структура подчинённых в этой папке не имеет значения. Стили и скрипты могут находиться рядом или быть разнесены — это также неважно.

Единственное обязательное условие — файл `src/Index.ts` должен прямо или косвенно импортировать все остальные файлы расширения. Под косвенным импортированием подразумевается что файл `Index.ts` импортирует файл `A.ts`, а файл `A.ts` в свою очередь импортирует `B.ts` (в этом случае импортировать `B.ts` в

файле `Index.ts` не обязательно). Также рекомендуется в файле `Index.ts` располагать код по регистрации обработчиков событий в глобальной области видимости (в объекте window), регистрации ЭУ через `controlFactory`, сервисов и т.д. В противном случае в процессе сборки директивы импорта будут проигнорированы как неиспользуемые.

Для редактирования файлов удобно использовать Visual Studio Code 2022. Достаточно открыть саму папку в редакторе, после чего все файлы будут доступны в дереве проекта. Можно также использовать Visual Studio версии 2022 аналогичным образом.

При использовании сущностей Web-клиента можно использовать функцию авто-импорта. Например, мы можем начать писать вызов функции `publishAsGlobal`, и Visual Studio Code 2022 автоматически предложит добавить импорт во всплывающем меню.

В процессе разработки можно однажды запустить команду (для этого удобно воспользоваться встроенным терминалом VSCode):

```
npm run watch
```

Данная команда начнет отслеживание изменений в исходных файлах, и при каждом изменении будет повторно собирать расширение и копировать файлы в `SITE_ROOT`. В режиме отслеживания сборка осуществляется значительно быстрее.

```
npm run build
```

Команда `npm run build` собирает скрипты в один бандл `extension.js` без сжатия и с включенным маппингом исходных файлов. То есть, в браузере исходные файлы будут отображаться в том же виде, в каком они есть в папке `src`, что удобно для отладки. При некорректной работе маппинга, можно отключить в браузере маппинг, и использовать в отладке сам файл `extension.js` (т.к. он не минифицирован).

В процессе сборки используется `rollup и gulp`, однако можно использовать любую другую систему сборки (`webpack`, `parcel` и т.д.). Единственное требование — необходимо использовать какую-либо систему `javascript` модулей, поддерживаемую `SystemJS` (`amd`, `commonjs`, `system`).



Обратите внимание, что файлы модулей устанавливаются в папку

> `Content/Modules/<НазваниеРешения>`

Для сборки расширения в режиме `production` необходимо запустить команду:

```
npm run build:prod
```

Единственное её отличие от `npm run build` в том, что результирующие файлы минифицируются.

Панель отправки на ознакомление

Данный раздел содержит описание примера с элементом управления `Панель отправки на ознакомление`, который демонстрирует запуск бизнес-процесса из карточки документа и создание собственной боковой панели.

Ссылка на пример на GitHub: AcquaintancePanel.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Проект "AcquaintancePanelDesignerExtension"

1. Проект-расширение для конструктора разметок. Содержит описание элемента управления `Пример боковой панели` для программы Конструктор Web-разметок.
2. Демонстрирует использование описания ЭУ в формате `.xml`, задание ему стандартных свойств, реализованных в программе Конструктор Web-разметок и добавление собственного свойства, используя класс `ProprietyDescription`.

Проект "AcquaintancePanelServerExtension"

- Проект-расширение для Web-клиент. Содержит бизнес-логику и реализацию элемента управления на клиенте.
- Демонстрирует расширение функционала с помощью добавления новых сервисов, моделей, контроллеров.
- Реализован контроллер `LayoutBusinessProcessController` с методом `SendToAcquaintance`, который вызывает сервис `ILayoutBPService` для отправки документа на ознакомление. В контроллере принудительно записан идентификатор бизнес-процесса.

- Для старта этого бизнес-процесса необходимо передать в качестве параметров "Ознакомителей" (`Performers`), "Документ" (`Document`) и дату "Ознакомиться до" (`EndDate`).

Проект "AcquaintancePanelWebExtension"

- Клиентский ЭУ представляет собой кнопку, при нажатии на которую появляется боковая панель.
- Боковая панель включает в себя стандартные ЭУ Web-клиент:
  - `Сотрудники` для задания списка сотрудников, которым придет задание на ознакомление,
  - `Дата/время`, дата "Ознакомиться до".
- При нажатии на кнопку **Отправить на ознакомление** происходит проверка параметров, и с помощью объекта `requestManager` отправляется запрос на сервер.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проект `Controls > AcquaintancePanel > AcquaintancePanelDesignerExtension`.
3. Соберите проект `Controls > AcquaintancePanel > AcquaintancePanelServerExtension`.
4. Откройте консоль в папке `Controls > AcquaintancePanel > AcquaintancePanelWebExtension` и выполните команду `npm install` и `npm run build:prod`.
5. Скопируйте каталог `SamplesOutput\Plugins\AcquaintancePanelDesignerExtension` в каталог `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins`.
6. Скопируйте каталог `SamplesOutput\Extensions\AcquaintancePanelServerExtension` в каталог `Путь к сайту Web-клиента\Extensions`.
7. Скопируйте файл `SamplesOutput\Extensions\ru\AcquaintancePanelServerExtension.resources.dll` в каталог `Путь к сайту Web-клиента\Extensions\ru`.
8. Скопируйте файл `SamplesOutput\Extensions\uk\AcquaintancePanelServerExtension.resources.dll` в каталог `Путь к сайту Web-клиента\Extensions\uk`.
9. Скопируйте каталог `SamplesOutput\Content\Modules\AcquaintancePanel` в каталог

`Путь к сайту Web-клиента\Content\Modules`.

10. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте любую разметку просмотра.
3. Выберите условия использования этой разметки.
4. Откройте разметку и добавьте в неё новый элемент управления `Пример боковой панели`.
5. Задайте желаемые параметры элемента управления. Для корректной работы задайте `editOperation = "Send for acquaintance"` или "Отправка на ознакомление".
6. Сохраните разметку.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
8. Откройте карточку с этой разметкой.
9. Убедитесь, что появился новый элемент управления, кнопку, при нажатии на которую появляется боковая панель.
10. Заполните необходимые поля и нажмите **Отправить на ознакомление**.
11. Должен запуститься бизнес-процесс, и указанным сотрудникам придет задание на ознакомление со ссылкой на документ.

Пример флага

Данный раздел содержит описание примера реализации элемента управления `Пример флага`.

Ссылка на пример на GitHub: CheckBox.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Проект "CheckBoxDesignerExtension"

1. Проект-расширение для конструктора разметок. Содержит описание элемента

управления Пример флага для программы Конструктор Web-разметок.

2. Демонстрирует описание и подключение нового ЭУ, используя класс ControlTypeDescription, задание ему стандартных свойств, реализованных в программе Конструктор Web-разметок. См. PropertyFactory.GetNameProperty() и добавление собственного свойства, используя класс PropertyDescription. См. свойство DefaultValue.
3. Для свойства **Поле секции**, источник данных, реализован редактор, фильтрующий поля по типу boolean, класс BooleanMetadataEditor.

Проект "CheckBoxWebExtension"

Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт и стили для CheckBox.

Сборка и установка

1. Откройте /Samples.sln.
2. Соберите проект Controls > CheckBox > CheckBoxDesignerExtension.
3. Откройте консоль в папке Controls > CheckBox > CheckBoxWebExtension и выполните команду npm install и npm run build:prod.
4. Скопируйте каталог SamplesOutput\Plugins\CheckBoxDesignerExtension в каталог Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins.
5. Скопируйте каталог SamplesOutput\Content\Modules\CheckBoxWebExtension в каталог Путь к сайту Web-клиента\Content\Modules.
6. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте любую разметку для карточки вида *Служебная записка*.
3. Выберите условия использования этой разметки.
4. Откройте разметку и добавить в нее новый элемент управления Пример флага.
5. Задайте желаемые параметры элемента управления.
6. Сохраните разметку.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
8. Откройте карточку с этой разметкой.

9. Убедитесь, что появился новый элемент управления.

Групповая операция / Скачивание файлов

Данный раздел содержит описание примера реализации элемента управления групповой операции *Групповая операция / Скачивание файлов*.

Ссылка на пример на GitHub: DownloadFilesGroupOperation.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Проект "DownloadFilesGroupOperationDesignerExtension"

Проект-расширение для конструктора разметок. Содержит описание элемента управления групповой операции *Групповая операция / Скачивание файлов* для программы Конструктор Web-разметок. Демонстрирует описание и подключение нового ЭУ, используя класс `ControlTypeDescription`,

Проект "DownloadFilesGroupOperationServerExtension"

Содержит контроллер, в котором формируется список файлов документа, запрашиваемый элементом управления *Групповая операция / Скачивание файлов*.

Проект "DownloadFilesGroupOperationWebExtension"

Проект-расширение клиентской части Web-клиент. Содержит реализацию элемента управления *Групповая операция / Скачивание файлов*.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проект `Controls > DownloadFilesGroupOperation > DownloadFilesGroupOperationDesignerExtension`.
3. Соберите проект `Controls > DownloadFilesGroupOperation > DownloadFilesGroupOperationServerExtension`.
4. Откройте консоль в папке `Controls > DownloadFilesGroupOperation > DownloadFilesGroupOperationWebExtension` и выполните команду `npm install` и `npm run build:prod`.

5. Скопируйте каталог `SamplesOutput\Plugins\DownloadFilesGroupOperationDesignerExtension` в каталог `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins`.
6. Скопируйте каталог `SamplesOutput\Extensions\DownloadFilesGroupOperationServerExtension` в каталог `Путь к сайту Web-клиента\Extensions`.
7. Скопируйте каталог `SamplesOutput\Content\Modules\DownloadFilesGroupOperation` в каталог `Путь к сайту Web-клиента\Content\Modules`.
8. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. В разметку **WebFrame** › **Папка** добавьте ЭУ *Групповая операция / Скачивание файлов*.
3. Сохраните разметку.
4. Откройте каталог с документами в Web-клиенте.
5. Включить режим групповых операций и убедиться, что появился новый элемент управления.

Элемент управления "Обменный курс"

Данный раздел содержит описание примера реализации элемента управления `ExchangeRates`. Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт и стили для `SampleExchangeRates`.

Ссылка на пример на GitHub: ExchangeRates.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Откройте консоль в папке `Controls > ExchangeRates > ExchangeRatesWebExtension` и выполните команду `npm install` и `npm run build:prod`.

3. Скопируйте каталог `SamplesOutput\Plugins\SampleExchangeRatesDesignerExtension` в каталог `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins`.
4. Скопируйте каталог `SamplesOutput\Content\Modules\ExchangeRatesWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Импортируйте решение из файла `SampleExchangeRatesControlDescriptionLayout.xml`.
3. Сделайте решение `SampleExchangeRatesControlDescriptionLayout` активным для карточки типа Документ вида *ДокументУД/Исходящий*.
4. Откройте разметку `SampleExchangeRatesControlDescriptionLayout`
5. Зарегистрируйте бесплатный аккаунт на сайте fixer.io.
6. Перейдите по ссылке https://fixer.io/quickstart и скопировать ключ доступа к API.
7. Задайте параметру **`Ключ доступа к API`** элемента управления `ExchangeRates` значение вашего ключа из шага выше.
8. Задайте остальные желаемые параметры элемента управления.
9. Сохраните разметку.
10. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
11. Откройте карточку с этой разметкой.
12. Убедитесь, что появился новый элемент управления.

Описание ЭУ "SampleExchangeRatesControlDescription.xml"

Демонстрирует добавление нового ЭУ без написания расширения для конструктора разметок. Для этого необходимо описать ЭУ в xml-файле, используя уже подключенные к конструктору свойства и ресурсы.

```
<?xml version="1.0" encoding="utf-8" ?>
<Controls>
  <Control Name="SampleExchangeRates" ControlGroupResourceKey="ControlGroup_Samples"
DisplayName="Sample exchange rates">
    <Properties> ①
      <Property Type="Name" />
```

```
      <Property Type="Visibility" />
      <Property Type="StandardCssClass" DefaultValue="sample-exchange-rates" /> ②
      <Property Type="CustomCssClasses" />
    </Properties>
  </Control>
</Controls>
```

① Содержит перечисление свойств ЭУ.

② `DefaultValue` задаёт значение поля по умолчанию.

Элемент управления "Пример изображения"

Данный раздел содержит описание примера элемента управления `Пример изображения`.

Ссылка на пример на GitHub: Image.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Проект "ImageDesignerExtension"

1. Проект-расширение для конструктора разметок. Содержит описание элемента управления `Пример изображения` для программы Конструктор Web-разметок.
2. Демонстрирует описание и подключение нового ЭУ, используя класс `ControlTypeDescription`, задание ему стандартных свойств, реализованных в программе Конструктор Web-разметок[10] и добавление кастомных свойств, используя класс `PropretyDescription` [11].
3. Для собственного свойства `Slider` реализован свой редактор, `SliderEditor.xaml`.

Проект "ImageServerExtension"

Проект-расширение для Web-клиента. Содержит бизнес-логику и реализацию элемента управления на сервере. Демонстрирует работу с собственным свойством `Slider` — преобразование его значения `string` в рабочую модель, `List` с помощью класса `SliderConverter` и метода `ConvertForDisplay`. Эта модель дублируется на клиенте `ISliderItem[]`. Таким образом реализована связка `List` — `JSON` — `ISliderItem[]` между сервером и клиентом.

С помощью Slider можно пролистывать несколько изображений, вешая обработчики onClick на соответсвующее элементы разметки.

Проект "ImageWebExtension"

Содержит скрипты и стили ЭУ Пример изображения.

Сборка и установка

1. Откройте /Samples.sln.
2. Соберите проект Controls > Image > ImageDesignerExtension.
3. Соберите проект Controls > Image > ImageServerExtension.
4. Откройте консоль в папке Controls > Image > ImageWebExtension и выполните команду npm install, затем npm update и в конце npm run build:prod.
5. Соберите проект ControlProperties > Url > UrlPropertyDesignerExtension.
6. Скопируйте каталог SamplesOutput\Plugins\ImageDesignerExtension в каталог Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins.
7. Скопируйте каталог SamplesOutput\Extensions\ImageServerExtension в каталог Путь к сайту Web-клиента\Extensions.
8. Скопируйте каталог SamplesOutput\Content\Modules\ImageWebExtension в каталог Путь к сайту Web-клиента\Content\Modules.
9. Скопируйте каталог SamplesOutput\Plugins\UrlPropertyDesignerExtension в каталог Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins.
10. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте любую разметку.
3. Выберите условия использования этой разметки.
4. Откройте разметку и добавьте в нее новый элемент управления Пример изображения.
5. Задайте желаемые параметры элемента управления.



При указании URL адрес должен быть доступен согласно политике CORS. Большинство серверов не разрешают запросы из веб-приложений, расположенных на других доменах.

> Политике CORS гарантированно удовлетворяют адреса внутри Web-клиента, например `Content/App/Assets/Images/favorite-empty.svg`.

6. Сохраните разметку.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
8. Откройте карточку с этой разметкой.
9. Убедитесь, что появился новый элемент управления.

Элемент управления "Link"

Данный раздел содержит описание примера реализации элемента управления `Пример ссылки`. Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт и стили для ЭУ `SampleLink`.

Ссылка на пример на GitHub: Link.

Пример требует подключенного расширения программы Конструктор Web-разметок `UrlPropertyDesignerExtension`.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Откройте консоль в папке `Controls > Link > LinkWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
3. Скопируйте каталог `SamplesOutput\Plugins\LinkDesignerExtension` в каталог `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins`.
4. Скопируйте каталог `SamplesOutput\Content\Modules\LinkWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Импортируйте решение из файла `SampleLinkControlDescriptionLayout.xml`.
3. Сделайте решение `LinkControlDescriptionLayout` активным для карточки типа Документ вида ДокументУД/Исходящий.
4. Откройте разметку `LinkControlDescriptionLayout`.
5. Задайте желаемые параметры элемента управления `Пример ссылки`.
6. Сохраните разметку.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
8. Откройте карточку с этой разметкой.
9. Убедитесь, что появился новый элемент управления.

Описание ЭУ "SampleLinkControlDescription.xml"

Демонстрирует добавление нового ЭУ без написания расширения для конструктора разметок. Для этого необходимо описать ЭУ в xml-файле, используя уже подключенные к конструктору свойства и ресурсы.

```
<?xml version="1.0" encoding="utf-16" standalone="no"?>
<Data>
  <SolutionsDirectory CardID="3200C1F1-6CAA-4C73-A169-78F3A12D155C" CardTypeID="3200C1F1-
6CAA-4C73-A169-78F3A12D155C" Description="SolutionsDirectory" CreationDateTime="2018-12-
06T12:18:50" ChangeDateTime="2019-02-21T11:44:59" Template="0" Topic="" Barcode="">
    <Solutions>
      <SolutionsRow RowID="958039AB-72EC-4E16-B25A-D00C97A160DE" OwnServerID="00000000-
0000-0000-0000-000000000000" ChangeServerID="00000000-0000-0000-0000-000000000000"
SolutionID="A925F725-1D89-4FD5-900F-9A097869D479" Name="LinkClientExtensionLayout"
Version="1.0" Mode="0">
        <Layouts>
          <LayoutsRow RowID="56BCF095-8303-4DAB-955C-CAFADBB7CDEB" OwnServerID="00000000-
0000-0000-0000-000000000000" ChangeServerID="00000000-0000-0000-0000-000000000000"
LayoutID="5F515439-7950-4246-8EEF-808B3D894124" Name="LinkClientExtensionLayout"
LayoutFileID="DD1B7EFE-2E29-43F6-BFC9-32144861DADE" LocationID="8e40f327-9517-4a43-998d-
bf2bd619588d" LayoutFileHash="F58E7A92D0A915B0CF90C4AB182A7CEE" LayoutType="1" />
        </Layouts>
      </SolutionsRow>
    </Solutions>
  </SolutionsDirectory>
  <File FileID="DD1B7EFE-2E29-43F6-BFC9-32144861DADE" OwnerCardID="3200C1F1-6CAA-4C73-
A169-78F3A12D155C" Name="LinkClientExtensionLayout" Size="2056" CreationDateTime="2019-
02-21T10:10:01" ChangeDateTime="2019-02-21T11:44:59" StandardAttribs="0"
ExtendedAttribs="0">![CDATA[77u/PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0idXRmLTE2Ij8+PEVsZ
```

```
W1lbnQgSWQ9IjZjODA4M2NmLTljY2EtNDczZi05MTAxLWUwMWQyNzQ1YWYzZiIgVHlwZT0iTGF5b3V0Ij48UHJvcG
VydGllcz48UHJvcGVydHkgTmFtZT0iTmFtZSIgVmFsdWU9InJvb3QiIFR5cGU9IlN5c3RlbS5TdHJpbmcsIG1zY29
ybGliIiBJc0F2YWlsYWJsZT0idHJ1ZSIgLz48UHJvcGVydHkgTmFtZT0iQ2FyZE9wZW5pbmciIFZhbHVlPSIiIFR5
cGU9IlN5c3RlbS5TdHJpbmcsIG1zY29ybGliIiBJc0F2YWlsYWJsZT0idHJ1ZSIgLz48UHJvcGVydHkgTmFtZT0iQ
2FyZE9wZW5lZCIgVmFsdWU9IiIgVHlwZT0iU3lzdGVtLlN0cmluZywgbXNjb3JsaWIiIElzQXZhaWxhYmxlPSJ0cn
VlIiAvPjxQcm9wZXJ0eSBOYW1lPSJDYXJkU2F2aW5nIiBWYWx1ZT0iIiBUeXBlPSJTeXN0ZW0uU3RyaW5nLCBtc2N
vcmxpYiIgSXNBdmFpbGFibGU9InRydWUiIC8+PFByb3BlcnR5IE5hbWU9IkNhcmRTYXZlZCIgVmFsdWU9IiIgVHlw
ZT0iU3lzdGVtLlN0cmluZywgbXNjb3JsaWIiIElzQXZhaWxhYmxlPSJ0cnVlIiAvPjxQcm9wZXJ0eSBOYW1lPSJDY
XJkU3RhdGVDaGFuZ2luZyIgVmFsdWU9IiIgVHlwZT0iU3lzdGVtLlN0cmluZywgbXNjb3JsaWIiIElzQXZhaWxhYm
xlPSJ0cnVlIiAvPjxQcm9wZXJ0eSBOYW1lPSJDYXJkTGF5b3V0U3dpdGNoaW5nIiBWYWx1ZT0iIiBUeXBlPSJTeXN
0ZW0uU3RyaW5nLCBtc2NvcmxpYiIgSXNBdmFpbGFibGU9InRydWUiIC8+PC9Qcm9wZXJ0aWVzPjxFbGVtZW50cz48
RWxlbWVudCBJZD0iZDQ0ODg2NDUtZjM1Yi00OGE0LWIxZTYtMzU4NzZiZjY3MGNjIiBUeXBlPSJTYW1wbGVMaW5rI
j48UHJvcGVydGllcz48UHJvcGVydHkgTmFtZT0iTmFtZSIgVmFsdWU9InNhbXBsZUxpbmsxIiBUeXBlPSJTeXN0ZW
0uU3RyaW5nLCBtc2NvcmxpYiIgSXNBdmFpbGFibGU9InRydWUiIC8+PFByb3BlcnR5IE5hbWU9IlZpc2liaWxpdHk
iIFZhbHVlPSImbHQ7P3htbCB2ZXJzaW9uPSIxdW90OzEuMCIgxdW90OyBlbmNvZGluZz0mcXVvdDt1dGYtMTYmcXVv
dDs/Jmd0OyYjeEQ7JiN4QTsmbHQ7RGV2aWNlRGVwZW5kZW50UHJvcGVydHlPZkJvb2xlYW4geG1sbnM6eHNpPSIxd
W90O2h0dHA6Ly93d3cudzMub3JnLzIwMDEvWE1MU2NoZW1hLWluc3RhbmNlJnF1b3Q7IHhtbG5zOnhzZD0mcXVvdD
todHRwOi8vd3d3LnczLm9yZy8yMDAxL1hNTFNjaGVtYSIgxdW90OyZndDsmI3hEOyYjeEE7ICAmbHQ7RGVza3RvcCZ
ndDt0cnVlJmx0Oy9EZXNrdG9wJmd0OyYjeEQ7JiN4QTsgICZsdDtUYWJsZXQmZ3Q7dHJ1ZSZsdDsvVGFibGV0Jmd0
OyYjeEQ7JiN4QTsgICZsdDtTbWFydHBob25lJmd0O3RydWUmbHQ7L1NtYXJ0cGhvbmUmZ3Q7JiN4RDsmI3hBOyAgJ
mx0O0NhbkNoYW5nZSZndDt0cnVlJmx0Oy9DYW5DaGFuZ2UmZ3Q7JiN4RDsmI3hBOyZsdDsvRGV2aWNlRGVwZW5kZW
50UHJvcGVydHlPZkJvb2xlYW4mZ3Q7IiBUeXBlPSJTeXN0ZW0uU3RyaW5nLCBtc2NvcmxpYiIgUmVzb2x2ZXJUeXB
lPSJEZXZpY2VEZXBlbmRlbnQiIElzQXZhaWxhYmxlPSJ0cnVlIiAvPjxQcm9wZXJ0eSBOYW1lPSJTdGFuZGFyZENz
c0NsYXNzIiBWYWx1ZT0ic2FtcGxlLWxpbmsiIFR5cGU9IlN5c3RlbS5TdHJpbmcsIG1zY29ybGliIiBJc0F2YWlsY
WJsZT0idHJ1ZSIgLz48UHJvcGVydHkgTmFtZT0iVGFiU3RvcCIgVmFsdWU9ImZhbHNlIiBUeXBlPSJTeXN0ZW0uQm
9vbGVhbiwgbXNjb3JsaWIiIElzQXZhaWxhYmxlPSJ0cnVlIiAvPjxQcm9wZXJ0eSBOYW1lPSJDbGljayIgVmFsdWU
9IiIgVHlwZT0iU3lzdGVtLlN0cmluZywgbXNjb3JsaWIiIElzQXZhaWxhYmxlPSJ0cnVlIiAvPjxQcm9wZXJ0eSBO
YW1lPSJNb3VzZU92ZXIiIFZhbHVlPSIiIFR5cGU9IlN5c3RlbS5TdHJpbmcsIG1zY29ybGliIiBJc0F2YWlsYWJsZ
T0idHJ1ZSIgLz48UHJvcGVydHkgTmFtZT0iTW91c2VPdXQiIFZhbHVlPSIiIFR5cGU9IlN5c3RlbS5TdHJpbmcsIG
1zY29ybGliIiBJc0F2YWlsYWJsZT0idHJ1ZSIgLz48L1Byb3BlcnRpZXM+PC9FbGVtZW50PjwvRWxlbWVudHM+PC9
FbGVtZW50Pg==]]</File>
</Data>
```


Групповая операция подписания

Пример текста

Данный раздел содержит описание примера реализации элемента управления `Пример текста`.

Ссылка на пример на GitHub: TextBox.

Пример требует подключенного расширения программы Конструктор Web-разметок `UrlProprtyDesignerExtension`.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Проект "TextBoxDesignerExtension"

1. Проект-расширение для конструктора разметок. Содержит описание элемента управления `Пример текста` для программы Конструктор Web-разметок.
2. Демонстрирует описание и подключение нового ЭУ, используя класс `ControlTypeDescription`, задание ему стандартных свойств, реализованных в программе Конструктор Web-разметок[12] и добавление собственного свойства, используя класс `PropretyDescription` [13].

Проект "CheckBoxWebExtension"

Проект-расширение клиентской части Web-клиента. Содержит клиентский скрипт, формирующий поле ввода текста и картинку (свойство `Url`), и стили для ЭУ `Пример текста`.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проект `Controls > TextBox > TextBoxDesignerExtension`.
3. Откройте консоль в папке `Controls > TextBox > TextBoxWebExtension` и выполните команду `npm install`, затем `npm update` и в конце `npm run build:prod`.
4. Скопируйте каталог SamplesOutput\Plugins\TextBoxDesignerExtension в каталог `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins`.
5. Скопируйте каталог `SamplesOutput\Content\Modules\TextBoxWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
6. Выполните инструкции по сборке и установке примера `ControlProperties > Url`.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте любую разметку редактирования.



Пример рассчитан только на работу в разметках редактирования.

3. Выберите условия использования этой разметки.
4. Откройте разметку и добавить в нее новый элемент управления `Пример текста`.
5. Задайте желаемые параметры элемента управления.



При указании URL адрес должен быть доступен согласно политике CORS. Большинство серверов не разрешают запросы из веб-приложений, расположенных на других доменах. Политике CORS гарантированно удовлетворяют адреса внутри Web-клиента, например `Content/App/Assets/Images/favorite-empty.svg`.

6. Сохраните разметку.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
8. Откройте карточку с этой разметкой.
9. Убедитесь, что появился новый элемент управления.

Дополнительная информация

Руководство по разработке решений Web-клиента

Дополнительная информация

Дополнительная информация по разработке примеров для конструктора разметок.

- [WebClient:new-controls-get-client-component-service]
- [WebClient:new-controls-nested-controls]
- [WebClient:new-controls-stop-cancellable-operation]
- Больше примеров категории в навигационном меню слева…

Получение сервисов в клиентском компоненте

В клиентском компоненте элемента управления можно получить доступ к стандартным (список в справочнике JS API, раздел Services) и разработанным клиентским сервисам.

Ссылка на сервисы автоматически передается при загрузке клиентского компонента. Чтобы получить ссылку, нужно добавить в класс параметров поле

services с указанием типов требуемых сервисов:

```
@rw services?: $EditOperationStore & $UrlStore & $RequestManager;
```

В данном примере в services сохраняются сервисы: `$EditOperationStore`, `$UrlStore`, `$RequestManager`.

Обратиться к сервису можно следующим образом:

```
let editOperations = this.state.services.editOperations.getAll(); ①
```

① Получаем список всех операций редактирования с помощью метода `getAll` сервиса `$EditOperations`.

Использование вложенных элементов управления

Стандартные и разработанные элементы управления Web-клиента можно использовать для формирования виртуального DOM другого элемента управления.

Следующий код демонстрирует пример использования стандартного элемента управления `Дата/время` в функции отрисовки элемента управления.

```
renderControl() {
    return (
        <div>
            <DateTimePicker name="someControlNestedControl" editMode={EditMode.Edit}
                    required={true} placeHolder={resources
.AcquaintancePanel_ConsiderationDatePlaceholder} />
        </div>
    );
}
```

Остановка выполнения операции с помощью прерываемого события элемента управления

События, генерируемые элементом управления, могут быть стандартными (непрерываемые) и прерываемыми, с помощью которых обработчик может остановить выполнение операции (например, сохранение карточки).

Чтобы создать прерываемое событие:

1. Добавьте событие в описатель элемента управления:

```
<?xml version="1.0" encoding="utf-8" ?>
```

```
<Controls>
  <Control Name="SomeControl" DisplayName="Пример элемента управления">
    <Properties>
      <Property Type="Name" />
    </Properties>
    <Events>
      <Event Name="someEventing" DisplayName="Перед выполнение операции"/> ①
    </Events>
  </Control>
</Controls>
```

① Прерываемое событие.

2. В клиентском компоненте элемента управления в классе параметров для события укажите тип CancelableApiEvent:

```
export class SomeControlParams extends BaseControlParams {
    @apiEvent someEventing?: CancelableApiEvent<IEventArgs>;
}
```

3. В клиентском компоненте элемента управления в классе реализации:

   i. Проинициализируйте прерываемое событие с помощью метода `CancelableEvent.Create`:

```
construct() {
    super.construct();

    this.state.someEventing = CancelableEvent.Create<IEventArgs>(this.state.
wrapper);
}
```

   ii. Во внутреннем обработчике события, вызываемого перед выполнением операции, добавьте код, который будет выполняться при прерывании события внешними обработчиками, и код для продолжения обработки события:

```
protected internalEventHandler = async () => {
    try {
        await CancelableEvent.cast(this.state.someEventing).trigger().promise;

        console.log("Выполнение операции ..."); ①
    } catch {
```

```
        console.log("Выполнение операции прервано ..."); ②
    }
}
```

① Код, вызываемый при продолжении операции.

② Код, вызываемый при отмене операции.

Переопределение стиля элемента управления

Разработчик может изменить стандартный стиль элемента управления с помощью дополнительных CSS-правил.



Изменение внутренних стилей стандартных элементов управления не рекомендуется. Поддержка обратной совместимости верстки элементов управления не гарантируется Docsvision.

Изменение стандартного стиля

Чтобы изменить стандартный стиль элемента управления, выполните следующие действия:

1. Создайте CSS-файл с классом с правилами, переопределяющими стандартный стиль элемента управления.



CSS-правила, используемые стандартным стилем элемента управления, можно определить с помощью инструментов разработчика, предоставляемых веб-браузером. Пример приведён в конце пункта; также смотрите примеры в репозитории Docsvision на Github.

2. Создайте в папке `/lib/docsvision/webclient/Content/Modules` каталог для размещения файлов собственного Решения. В названии папки предпочтительно использовать только латиницу, не использовать пробелы и спецсимволы.
3. Скопируйте CSS-файл, полученный на шаге 1, в созданный каталог Решения.
4. Укажите название CSS-класса, реализованного на шаге 1, в программе Конструктор Web-разметок в свойстве Дополнительные css классы изменяемого элемента управления. Сохраните разметку.
5. Перезапустите службу Сервера Web-клиента или обновите страницу в браузере, отключив кэширование. Перезапуск необходим только при добавлении нового файла, при изменении — не требуется.

Пример переопределения стиля

В качестве примера далее приведена последовательность действий при изменении цвета фона элемента управления Строка.

1. Создадим файл style.css со следующим содержимым:

```
.new-textbox input
{
    background-color: yellow !important;
}
```

   Класс new-textbox определяет правило, которое применяется к элементу input, и устанавливает желтый цвет для фоновой заливки.

Класс был сформирован следующим образом:

1. Открыли разметку карточки с элементом управления Строка. В данном случае — разметка создания карточки исходящего документа.
2. Открыли код элемента управления Строка (в разметке карточки — поле Тема) с помощью инструментов разработчика.

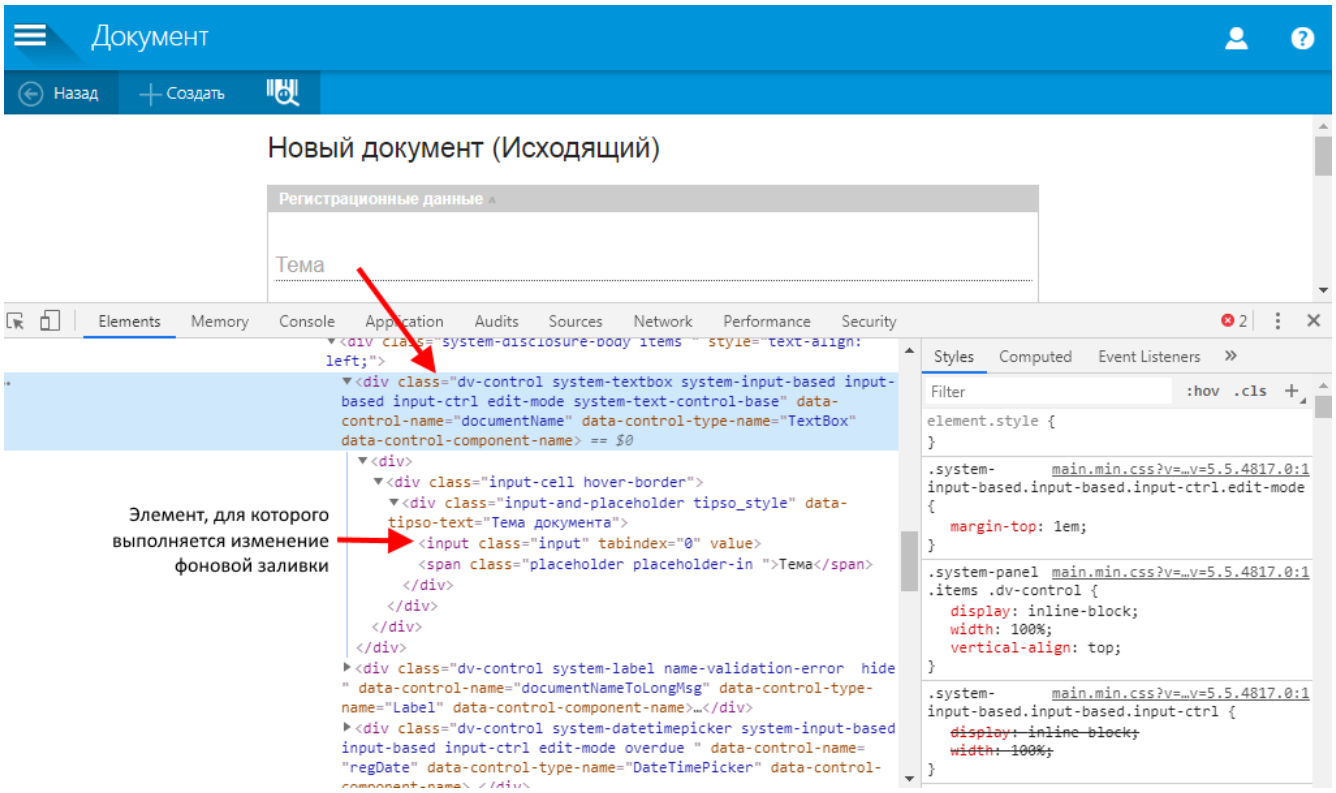


Рисунок 710. Код элемента управления в веб-браузере Chrome

3. Нашли в структуре элемента управления Строка элемент, требующий

изменения фоновой заливки. Данным элементом является `input` (поле для ввода).

Для создания селектора CSS-класса, который позволит установить стиль для элемента `input`, достаточно в селекторе указать название тега (`input`) и установить приоритет `important` для свойства `background-color` (фоновая заливка).

4. Создадим для Решения папку `/lib/docsvision/webclient/Content/Modules/SuperSolution`.
5. Скопируем файл style.css в папку `/lib/docsvision/webclient/Content/Modules/SuperSolution`.
6. Укажем название класса `new-textbox` в свойстве **Дополнительные css классы** элемента управления Строка в программе Конструктор Web-разметок и сохраним разметку.

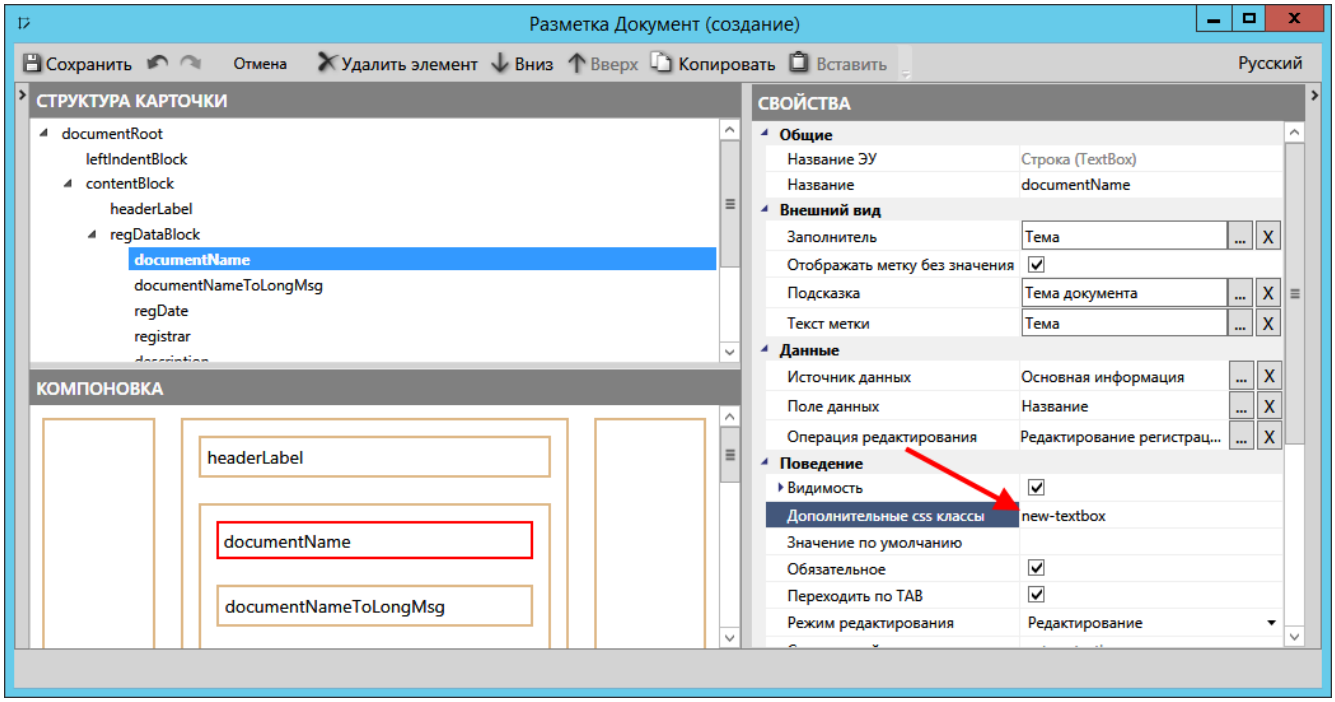


Рисунок 711. Дополнительные CSS-классы в настройках элемента управления TextBox

7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
8. Запустим карточку с настроенной разметкой для проверки применения стиля.

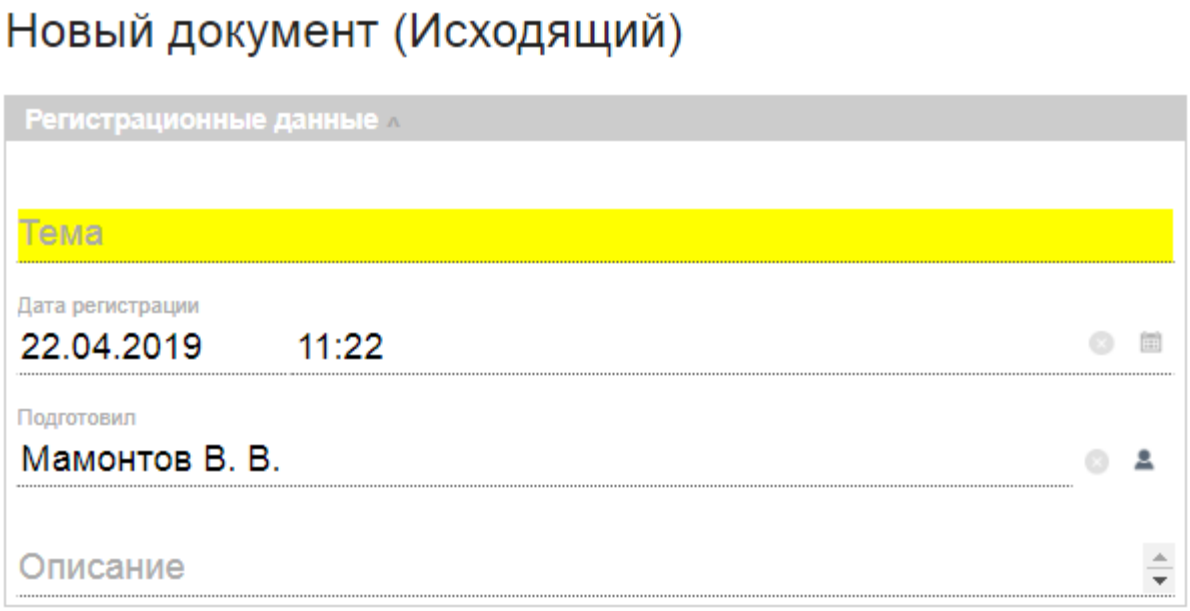


Рисунок 712. Разметка карточки с собственным стилем поля Тема

Переопределение реализации стандартного элемента управления собственной

Стандартная реализация элементов управления Web-клиента может не всегда отвечать требованиям пользователя. В качестве решения, программист может реализовать собственные типы элементов управления и использовать их в разметках, или переопределить стандартную реализацию существующих элементов управления. В последнем случае разметки остаются без изменений, а изменяется клиентская часть элементов управления.

Переопределение реализации стандартных элементов управления упрощает внедрение требуемой клиентской функциональности, но усложняет процесса обновления версии Web-клиента.



Данные способ разработки не позволяет переопределить серверную часть элемента управления и его описатель.

Чтобы переопределить стандартную реализацию элемента управления собственной:

1. Скопируйте файлы `.tsx` с реализацией клиентской части стандартного элемента управления и файлы зависимостей в отдельный каталог.
2. Внесите требуемые изменения в файлы.

   Изменять тип элемента управления не нужно.
3. Сохраните изменения и опубликуйте компоненты в каталоге собственного решения на сервере Web-клиента стандартным образом.

4. При обновлении версии Web-клиента выполните следующие действия:
   a. Проанализируйте исходный код стандартной реализации на наличие изменений[14] относительной версии, для которой создавалось переопределение.
   b. Если изменения затронули настройки или формат данных элемента управления, внесите соответствующие изменения в собственный компонент, переопределяющий стандартный.

Другие примеры

Руководство по разработке решений Web-клиента

Другие примеры

Описание прочих примеров разработки.

- [WebClient:other-custom-stage-service]
- [WebClient:other-dv-web-extensions]
- [WebClient:other-external-web-service]
- Больше примеров категории в навигационном меню слева…

Работа с машиночитаемой доверенностью

Данный пример, расположенный в репозитории Docsvision на GitHub, содержит методы работы с МЧД для Web-клиента и для Windows-клиента.

Ссылка на пример на GitHub: PowersOfAttorney.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Модуль Базовые объекты версии 6.1, сборка 176 и выше.
- Модуль Web-клиент версии 6.1, сборка 651 и выше.

Ограничения примера

1. Виды передоверия в примере предназначены только для ознакомления. Для их использования необходима доработка под ситуации, не требующие

нотариального удостоверения. По умолчанию создание карточек данных видов запрещено.

2. Формат 5.01 не реализован.
3. Для формата 002 не реализованы следующие сценарии:
   a. Основная доверенность

Доверитель	**Лицо, действующее без доверенности от имени доверителя**	**Представитель**
Российское юридическое лицо	Юридическое лицо	Организация / Индивидуальный предприниматель
Иностранное юридическое лицо	Все типы	Все типы
Индивидуальный предприниматель	Все типы	Все типы
Физическое лицо	Все типы	Все типы

   b. Передоверие

Лицо, передавшее полномочия	**Лицо, получившее полномочия**
Российское юридическое лицо	Все типы
Индивидуальный предприниматель	Все типы
Физическое лицо	Организация/ Индивидуальный предприниматель

   c. Нотариальное удостоверение доверенностей.
4. Для формата EMCHD_1 (003) не реализованы следующие сценарии:
   a. Основная доверенность

Доверитель	**Единоличный исполнительный орган**	**Представитель**
Российское юридическое лицо	Управляющая компания/ Индивидуальный предприниматель	Юридическое лицо/ Индивидуальный предприниматель/ Филиал (обособленное подразделение) российского юридического лица/ Филиал (аккредитованное представительство) иностранного юридического лица
Иностранное юридическое лицо	Все типы	Все типы
Индивидуальный предприниматель	Все типы	Все типы
Физическое лицо	Все типы	Все типы

b. Передоверие

Лицо, передавшее полномочия	**Лицо, получившее полномочия**
Российское юридическое лицо	Все типы
Индивидуальный предприниматель	Все типы
Физическое лицо	Юридическое лицо/ Индивидуальный предприниматель/ Филиал (обособленное подразделение) российского юридического лица/ Филиал (аккредитованное представительство) иностранного юридического лица

Лицо, передавшее полномочия	Лицо, получившее полномочия
Филиал (обособленное подразделение) российского юридического лица	Все типы
Филиал (аккредитованное представительство) иностранного юридического лица	Все типы

c. Нотариальное удостоверение доверенностей.

Описание примера

Пример включает компоненты:

- **PowersOfAttorney › PowersOfAttorneyServerExtension** — папка с серверным расширением Web-клиента, в котором реализованы функции создания СКД из демонстрационной карточки доверенностей.
- **PowersOfAttorney › PowersOfAttorneyWebExtension** — папка с клиентским расширением, в котором реализованы обработчики смены состояния ПКД и управления МЧД, видимостью и обязательностью полей.
- **PowersOfAttorney › PowersOfAttorneyDesignerExtension** — папка с расширением конструктора разметок, в котором реализовано описание элемента управления для группового подписания.
- **PowersOfAttorney › PowersOfAttorney.UserCard.Common** — папка с проектом `PowersOfAttorney.UserCard.Common`, который используется и для Web-клиента и для Windows-клиента.
- **PowersOfAttorney › PowersOfAttorney.Scripts** — содержит файл скрипта для Windows клиента.
- **PowersOfAttorney › Data › PowersOfAttorneySolution** — папка с решением, включающим разметки тестовой карточки доверенности.
- **PowersOfAttorney › Data › SolutionOfPOA.sol** — решение, добавляющее в систему Docsvision новые виды карточки *Документ* (*Доверенность* и *Передоверие*) и необходимые расширенные метаданные.

Сборка проекта

i. Сборка серверной части.

1. Откройте решение **PowersOfAttorney ›**

PowersOfAttorneyServerExtension.sln.

2. Соберите проект `PowersOfAttorneyServerExtension`.

ii. Сборка клиентской части.

1. Откройте в командной строке папку **PowersOfAttorney › PowersOfAttorneyWebExtension**.
2. Выполните команды:

```
$ npm install
$ npm update
$ npm run build:prod
```

iii. Публикация компонентов на сервере Web-клиент.

1. Скопируйте папку `PowersOfAttorney\SamplesOutput\Content\Modules\PowersOfAttorneyWebExtension` в `Каталог-установки-Web-клиента\Content\Modules`.
2. Скопируйте файл `PowersOfAttorney\SamplesOutput\Extensions\PowersOfAttorneyServerExtension\PowersOfAttorneyServerExtension.dll` в `Каталог-установки-Web-клиента\Extensions`.
3. Скопируйте файл `PowersOfAttorney\SamplesOutput\Extensions\PowersOfAttorneyServerExtension\PowersOfAttorneyServerExtension.resources.dll` в `Каталог-установки-Web-клиента\Extensions\ru`.
4. Скопируйте файл `PowersOfAttorney\SamplesOutput\Extensions\PowersOfAttorney.UserCard.Common\PowersOfAttorney.UserCard.Common.dll` в `Каталог-установки-Web-клиента\Extensions`.
5. Скопируйте файл `PowersOfAttorney\SamplesOutput\Extensions\PowersOfAttorney.UserCard.Common\PowersOfAttorney.UserCard.Common.resources.dll` в `Каталог-установки-Web-клиента\Extensions\ru`.

iv. Перезапустите службу **dvappserver** для Linux и **Docsvision Platform** для Windows.

Проверка примера



Для проверки примера требуется установить версию Backoffice и Web-клиент с поддержкой СКД.

1. В Менеджере решений импортируйте решение *Машиночитаемая доверенность* — `PowersOfAttorney\Data\SolutionOfPOA.sol`. Рекомендуется использовать версию Менеджера решений версии 6.1 и выше.

   Компоненты данного решения:

   - Справочник видов:
     - *Документ — МЧД Доверенность (версия 002)*, *Передоверие (версия 002)*, *Доверенность (версия EMHCD_1)*, *Передоверие (версия EMHCD_1)*.
     - *Задание — Задание КС — На подписание МЧД*, *На согласование МЧД*.
   - Метаданные: Полномочия, МЧД формат единой формы, Текстовые полномочия (а также все подчиненные).
   - Конструктор правил нумерации: МЧД.
   - Поиск: МЧД (МЧД все, МЧД действующие, МЧД — я автор, МЧД — я подписант, МЧД отозванные, МЧД мои, Поиск МЧД)
   - Представление: МЧД → МЧД единого формата — представление
   - Папка *Доверенность* для хранения маршрутов согласования и подчиненные папки для работы с доверенностями.
   - Согласование: маршрут "согласование и подписание МЧД единого формата" — согласование настроено на группу юридического отдела (справочник сотрудников), подписание настроено на группу генеральные директора (справочник сотрудников). Согласование параллельное, без ограничений по времени. Можно убрать участников при отправке на согласование.

2. В справочнике видов (тип: *Согласование*, вид: *Усовершенствованное согласование*) настройте согласование для видов карточек МЧД. В окне *Настройка способа создания карточки* выберите шаблон *Согласование и подписание МЧД единый формат*.

3. В программе Конструктор Web-разметок импортируйте решение *Машиночитаемая доверенность*. Для этого выберите файл `Solution.xml` в папке решения `PowersOfAttorney\Data\PowersOfAttorneySolution\`.

4. Проверьте, что в Справочнике сотрудников заполнены поля для участников МЧД как описано в пользовательской документации.

5. В Web-клиенте создайте карточку документ вида *Доверенность (версия 002)*. Заполните обязательные поля.
   Если заполнены не все обязательные поля, при сохранении будет выдано предупреждение.

Чтобы создать свою разметку, необходимо:

- Добавить скрипт для отображения в модальном окне незаполненных обязательных полей `showRequiredFields` в качестве обработчика события **Перед началом сохранения карточки** `root` для разметок создания и/или редактирования в конструкторе разметок.
- Добавить скрипт `customizePowerOfAttorneyCardForEditLayout` в качестве обработчика события **После загрузки всех ЭУ** для root во всех разметках создания и редактирования карточки доверенности.
- Добавить скрипт `customizePowerOfAttorneyCardForViewCard` в качестве обработчика события **После загрузки всех ЭУ** для root во всех разметках просмотра карточки доверенности.
- Добавить скрипт `customizeSubstitutionPowerOfAttorneyCardForEditLayout` **После загрузки всех ЭУ** для root во всех разметках просмотра карточки передоверия.
- Добавить скрипт `customizeSubstitutionPowerOfAttorneyCardForViewLayout` в качестве обработчика события **После загрузки всех ЭУ** для root во всех разметках просмотра карточки передоверия.

6. В Web-клиенте создайте карточку документ вида *Доверенность (версия EMCHD_1)*. Заполните обязательные поля. Если заполнены не все обязательные поля, при сохранении будет выдано предупреждение.

   Чтобы создать свою разметку, необходимо:

   - Добавить скрипт для отображения в модальном окне незаполненных обязательных полей `showRequiredFields` в качестве обработчика события **Перед началом сохранения карточки** `root` для разметок создания и/или редактирования в конструкторе разметок.
   - Добавить скрипт `customizeSingleFormatPowerOfAttorneyForEditLayout` в качестве обработчика события **После загрузки всех ЭУ** `root` для разметок создания и редактирования карточки доверенности в конструкторе разметок.
   - Добавить скрипт `customizeSingleFormatPowerOfAttorneyForViewLayout` в качестве обработчика события **После загрузки всех ЭУ** `root` для разметок просмотра, описания и справки карточки доверенности в конструкторе разметок.
   - Добавить скрипт `customizeSingleFormatSPOACardForEditLayout` в качестве обработчика события **После загрузки всех ЭУ** `root` для разметок создания и редактирования карточки передоверия в конструкторе разметок.
   - Добавить скрипт `customizeSingleFormatSPOACardForViewLayout` в качестве

обработчика события **После загрузки всех ЭУ** `root` для разметок просмотра, описания и справки карточки передоверия в конструкторе разметок.

- Добавить скрипт `customizeSingleFormatPowerOfAttorneyForLocationLayout` в качестве обработчика события **После загрузки всех ЭУ** `root` для разметки локация карточки доверенности в конструкторе разметок.
- Добавить скрипт `customizeSingleFormatPowerOfAttorneyForLocationLayout` в качестве обработчика события **После загрузки всех ЭУ** `root` для разметки локация карточки передоверия в конструкторе разметок.

7. Нажмите кнопку создания МЧД. В результате будет создана МЧД, связанная с текущей карточкой документа.

   Чтобы создать свою разметку, необходимо:

   - Добавить скрипт `createPowerOfAttorney` в качестве обработчика события **При щелчке** для кнопки в разметке просмотра карточки доверенности (версия 002) в конструкторе разметок.
   - Добавить скрипт `createRetrustPowerOfAttorney` в качестве обработчика события **При щелчке** для кнопки в разметках просмотра карточки передоверия (версия 002) в конструкторе разметок.
   - Добавить скрипт `createEMCHDPowerOfAttorney` в качестве обработчика события **При щелчке** для кнопки для разметок просмотра карточки доверенности (версия EMCHD_1) в конструкторе разметок.
   - Добавить скрипт `createEMCHDRetrustPowerOfAttorney` в качестве обработчика события **При щелчке** на кнопку для разметок просмотра карточки передоверия (версия EMCHD_1) в конструкторе разметок.

8. Нажмите кнопку экспорта МЧД. На компьютер будет сохранён архив, содержащий файл МЧД в формате XML.

   Чтобы создать свою разметку, необходимо:

   - Добавить скрипт `exportPowerOfAttorneyWithoutSignature` в качестве обработчика события **При щелчке** для кнопки в разметке просмотра карточки доверенности и передоверия в конструкторе разметок.

9. Если планируется отправлять доверенность на регистрацию в распределённый реестр ФНС через провайдеров внешнего ЭДО необходимо выполнить соответствующие настройки.

   - Для работы через Контур.Диадок выполните настройки описанные в документации модуля "Модуль интеграции с операторами ЭДО", раздел "Настройка работы с МЧД".

- Для этого случая, в примере решения есть преднастроенные разметки: *Доверенность версии 002 (просмотр)_Контур.Диадок* и *Доверенность версии EMCHD_1 (просмотр)_Контур.Диадок*.

◦ Для работы через Контур.Доверенность выполните настройки описанные в документации модуля "Интеграции с реестром МЧД", раздел "Настройка Модуля интеграции с реестром МЧД".

- Для этого случая, в примере решения есть преднастроенные разметки: *Доверенность версии 002 (просмотр)_Контур.Доверенность* и *Доверенность версии EMCHD_1 (просмотр)_Контур.Доверенность*.

10. Нажмите кнопку подписания МЧД. Будет предложено выбрать сертификат подписи, выполнится подписание МЧД.

◦ Для создания своей разметки нужно добавить скрипт `signPowerOfAttorney` в качестве обработчика события **При щелчке** на кнопку для разметок просмотра карточки доверенности и передоверия в конструкторе разметок.

◦ При работе через Контур.Диадок, для подписания и последующей регистрации доверенности по файлу нужно добавить скрипт `signAndSendPowerOfAttorneyToRegistrationAsFile` в качестве обработчика события **При щелчке** на кнопку для разметки просмотра карточки доверенности.

◦ Если требуется разделить события подписания и отправки, можно воспользоваться скриптом `sendPowerOfAttorneyToRegistrationAsFile` (он выполняет только отправку уже подписанной доверенности в реестр через Контур.Диадок).

◦ При работе через Контур.Доверенность, для подписания и последующей регистрации доверенности по файлу нужно добавить скрипт `signAndSendForRegistrationToRegistry` в качестве обработчика события **При щелчке** на кнопку для разметки просмотра карточки доверенности.

◦ Если требуется разделить события подписания и отправки, можно воспользоваться скриптом `sendForRegistrationToRegistry` (он выполняет только отправку уже подписанной доверенности в реестр через Контур.Доверенность).

11. Нажмите кнопку экспорта МЧД. На компьютер будет сохранён архив, содержащий файл МЧД в формате XML и его подпись.

Чтобы создать свою разметку, необходимо:

◦ Добавить скрипт `exportPowerOfAttorneyWithSignature` в качестве обработчика события **При щелчке** для кнопки в разметки просмотра карточки

доверенности и передоверия.

12. Нажмите кнопку отзыва МЧД.

    Чтобы создать свою разметку, необходимо:

    - Добавить скрипт `revokePowerOfAttorney` в качестве обработчика события **При щелчке** на кнопку для разметки просмотра карточки доверенности и передоверия (версия 002) в конструкторе разметок.

    Чтобы создать свою разметку с отзывом без заявления, необходимо:

    - Добавить скрипт `revokePowerOfAttorneyWithoutApplication` в качестве обработчика события **При щелчке** на кнопку для разметки просмотра карточки доверенности и передоверия в конструкторе разметок.

    При работе через Контур.Доверенность

    - Для формирования, подписания и последующей отправки заявления на отзыв в ЦПРР ФНС нужно добавить скрипт `revokeAndRecallPowerOfAttorney` в качестве обработчика **При щелчке** на кнопку отзыва для разметки просмотра карточки доверенности.

    Если требуется разделить события формирования заявления и его отправки:

    - Можно воспользоваться скриптом `recallPowerOfAttorney`. Скрипт выполняет только отправку уже сформированного и подписанного заявления на отзыв через Контур.Доверенность.

13. Нажмите кнопку экспорта заявления на отзыв для доверенности и передоверия (версия 002). На компьютер будет сохранён архив, содержащий файл заявления на отзыв в формате XML и его подпись.

    Чтобы создать свою разметку, необходимо:

    - Добавить скрипт `exportApplicationForRevocation` в качестве обработчика события **При щелчке** на кнопку для разметки просмотра карточки доверенности (версия 002) и передоверия (версия 002) в конструкторе разметок.

14. Нажмите кнопку удаления пользовательской карточки доверенности.

    Из пользовательской карточки доверенности также удаляется системная карточка доверенности.

    Чтобы создать свою разметку, необходимо:

    - Добавить скрипт `deletePowerOfAttorney` в качестве обработчика события **Перед удалением карточки** для `root`.

15. Доверенность можно подписать в ходе согласования.

    Чтобы подписать доверенность в ходе согласования, необходимо:

    - Отправить Доверенность на согласование как описано в пользовательской документации.
    - Нажать кнопку **Подписать** в *Задании*. В результате сформируется СКД, затем подпишется доверенность, ПКД перейдет в статус `Подписана`, а задание перейдет в статус `Завершено`.
    - Для создания своей разметки нужно добавить скрипт `signAndSendPowerOfAttorneyToRegistrationAsFileFromTask` в качестве обработчика события `Перед выполнением операции` на ЭУ `Автомат состояния` для подписания и последующей регистрации доверенности по файлу.
      Для подписания без регистрации используйте скрипт `signPowerOfAttorneyFromTask`.



Права доступа к СКД определяются дискреционной моделью и наследуются от ПКД. Права переходят по *Сильной ссылке*, в которую прописывается `InstanceID` СКД благодаря наличию поля `POASysCardId` типа `RefCardID`.

Пример скриптов для Windows клиента

1. Скрипт находится в файле `CardDocumentДоверенность__версия_EMHCD_1_Script.cs` в проекте `PowersOfAttorney.Scripts`
2. Проект нужен только для проверки компилируемости скрипта. Ссылки на сборку `PowersOfAttorney.Scripts.dll` добавлять не надо.
3. Скрипт необходимо скопировать в справочник скриптов для двух видов (для доверенности EMHCD и передоверия EMHCD). Если у родительского вида для этих видов нет скрипта, надо открыть его и сгенерировать для него скрипт по умолчанию.
4. Необходимо закомментировать в файле скрипта для обоих видов строчку

   ```
   using CardDocumentМЧДScript = DocsVision.BackOffice.WinForms.ScriptClassBase;
   ```

   Строка нужна только для компиляции файла скрипта в составе проекта `PowersOfAttorney.Scripts`.

5. Скрипты для этих видов отличаются только названиями классов. Необходимо для каждого вида оставить только одно соответствующее ему название

класса (см. комментарий в скрипте).

6. В скриптах необходимо добавить ссылку на сборку `PowersOfAttorney.UserCard.Common.dll`, которую также необходимо распространить на все клиентские рабочие места.
7. В конструкторе разметок необходимо добавить кнопки (например в ленту). Названия кнопок должны соответствовать обработчикам в скрипте.

   В скрипте обработчики выглядят как `имяКнопки__ItemClick();` Если названия кнопки будут соответствовать обработчикам, то обработчики к кнопкам привяжутся автоматически, вручную их создавать не надо.

8. Можно привязать к кнопкам соответствующие операции, чтобы кнопки были доступны только в тех состояниях, когда их нажатие имеет смысл.

API для работы с МЧД

Для работы с МЧД можно использовать перечисленные ниже классы API.

- В `DocsVision.BackOffice.Cards.Web.Model` добавлены новые для работы с машиночитаемыми доверенностями:
  - `CreatePowerOfAttorneyFnsDovBbRequest` с полями:
    - `PowerOfAttorneyFnsDovBbData powerOfAttorneyData` — данные создаваемой доверенности.
    - `Guid representative` — представитель.
    - `Guid signer` — подписант.
    - `Guid parentPowerOfAttorney` — идентификатор родительской доверенности.
  - `SignPowerOfAttorneyRequest` с полями:
    - `Guid PowerOfAttorneyId` — идентификатор доверенности.
    - `byte[] Signature` — данные подписи.
- В контроллер `PowerOfAttorneyApiController` добавлены новые методы:
  - `POST CreatePowerOfAttorney(CreatePowerOfAttorneyFNSDOVBBRequest request)` — вызывает `IPowerOfAttorneyService.CreatePowerOfAttorney` (создание доверенности) с передачей полей из `CreatePowerOfAttorneyFNSDOVBBRequest`.
  - `GET GetMachineReadablePowerOfAttorney(Guid powerOfAttorneyId)` — возвращает МЧД доверенности для подписания.

- `POST AttachSignatureToPowerOfAttorney(AttachSignatureToPowerOfAttorneyRequest)` — вызывает `IPowerOfAttorneyService.AttachSignature`, загружающий подпись в существующую доверенность и изменяющий статус доверенности.

Подробнее про использование API для работы с МЧД можно узнать из описания веб-API, см. раздел "[WebClient:how-to-use-swagger]".

Специальный сервис логики этапа согласования

Данный раздел содержит описание примера реализации специального сервиса логики этапа согласования.

Ссылка на пример на GitHub: CustomStageService.

Сервис выполняет следующие функции:

- Подменяет вычисление согласующих.
- Показывает вариант настройки действий после создания задания на примере дополнения содержания задания.

Пример рассчитан на версии модулей Служба фоновых операций и Windows-клиент версии 6.1 или выше.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проект `Others > CustomStageService`.

Пример переопределения сервиса этапа

```
namespace CustomStageService
{
    public class CustomStageService : ApprovalStageService
    {
        public override IEnumerable<ApprovalStageApprover> GetStageApprovers
(ApprovalStage approvalStage, Document document) ①
        {
            List<ApprovalStageApprover> approvers = new List<ApprovalStageApprover>();
```

```
            IStaffService staffService = Context.GetService<IStaffService>();
            StaffEmployee employee = staffService.FindEmpoyeeByAccountName
(@"Test\TestEmployee");

            approvers.Add(CreateApprover(employee)); ②

            return approvers;
        }
        protected override void OnTaskCreated(ApprovalStage approvalStage, ApprovalPath
approvalPath, Reconcile reconcileCard, Task task) ③
        {
            task.MainInfo.Content += Environment.NewLine + "Тест расширения"; ④
        }
    }
}
```

① Переопределяем вычисления согласующих для этапа.

Параметры:

- `approvalStage` — текущий этап,
- `document` — документ.

② `CreateApprover` — штатный метод `ApprovalStageService` создающий запись согласующего из сотрудника.

③ Дополняем создание задания согласования.

Параметры:

- `approvalStage` — текущий этап.
- `approvalPath` — маршрут.
- `reconcileCard` — согласование.
- `task` — созданное задание.

④ Дополним содержание задания.

Проверка примера

1. В файле `WebCSamples\Others\CustomStageService\CustomStageService.cs` заменить `Test\TestEmployee` на одну из своих учётных записей, присутствующих в Docsvision.
2. Пересобрать расширение.
3. Добавить на сервере библиотеку `CustomStageService.dll` из каталога `bin\Debug\net6.0`, если в VisualStudio выбрана конфигурация сборки "Debug"

или `bin\Release\net6.0`, если выбрана конфигурация "Release".

- в папку common на сервер, где установлен WorkerService (`/lib/docsvision/common` на Linux)
- в папку, где установлен Windows-клиент, с которого производится настройка (например `C:\Program Files (x86)\Docsvision\Client`).

4. Открыть шаблон карточки любого этапа (например, "Согласование с рецензированием" из стандартного согласования модуля "Управление документами"), перейти на вкладку "Специальная логика этапа", в поле "Имя сервиса" выбрать библиотеку.
5. Настроить и запустить согласование с данным этапом в маршруте. В результате на данном этапе в качестве согласующего будет выбран пользователь, указанный на шаге 1 (например, если используется этап согласования, то этому пользователю придет задание на согласование). Также в созданном задании в описании появится текст "Тест расширения".

Разработка расширения для DVWebTool и DVSupService



Программа *DVWebTool* устанавливается на клиентскую машину только в ОС Microsoft Windows. Для ОС Linux используется *DVSupService*, см. подробнее Установка и запуск службы DVSupService.

Ссылка на пример на GitHub: Watermark.

Расширения для *DVSupService* и *DVWebTool* можно распространять с помощью пакетов Debian, с подробной инструкцией и необходимыми материалами можно ознакомиться в репозитории на GitHub.

Дополнительным компонентом модуля Web-клиент является программа *DVWebTool*, которая предоставляет функциям Web-клиента доступ к устройствам и программному обеспечению, установленному на компьютере пользователя Web-клиента. Программа *DVWebTool* также может быть использована для обработки произвольных задач на пользовательском компьютере.

Программа *DVWebTool* является приложением для рабочего стола с возможностью расширения функциональности. Расширение *DVWebTool* представляет собой DLL-файл содержащий ожидаемый класс с веб-методами. Данные методы могут вызываться клиентскими компонентами Web-клиента (есть стандартный метод JSAPI для вызова) или другими способами.

В качестве примера будет разработано расширение, реализующее функцию добавления водяных знаков в PDF-файлы карточки *Документ*.

Алгоритм работы расширения:

1. При вызове функции расширения Клиент передаёт идентификаторы PDF файлов карточки *Документ* (идентификаторы карточек *Файл* с версиями, а не идентификаторы собственно файлов) и идентификатор самой карточки *Документ*.
2. Расширение *DVWebTool* загружает PDF-файлы на компьютер пользователя с помощью функций, предоставляемых Web-клиентом.
3. Расширение *DVWebTool* добавляет в полученные PDF-файлы водяные знаки используя функции сторонней библиотеки для работы с PDF.
4. Расширение *DVWebTool* прикрепляет полученные файлы с водяными знаками с помощью функций, предоставляемых Web-клиентом.

   Расширение не должно изменять существующие файлы карточки: полученные PDF-файлы с водяными знаками должны быть добавлены в карточку в качестве дополнительных.

Для реализации функций, указанных в пп. 2 и 4, потребуется разработать расширение Web-клиента с соответствующей функциональностью.

Также для примера необходимо разработать клиентское расширение, которое будет вызывать функции расширения *DVWebTool* с передачей им идентификаторов PDF-файлов *Документа*, в которые нужно добавить водяные знаки.



Полный код примера доступен в репозитории на GitHub.

Расширение Web-клиента

Для работы расширения *DVWebTool* необходимо, чтобы Web-клиент по запросу возвращал файл из карточки *Файл* с версиями с переданным идентификатором, а также мог прикреплять переданные файлы к карточке *Документ*.

Требуемая функциональность может быть реализована с помощью расширения Web-клиента.

Ниже приведена часть исходного кода контроллера, реализованного в серверном расширении.

Файл Others/Watermark/WatermarkServerExtension/Controllers/FileOperationsController.cs

```
        public HttpResponseMessage GetFile(Guid fileCardID)
        {
            FileReader fileReader = fileService.GetFileReader(fileCardID);

            if (fileReader.FileID == Guid.Empty)
                return new HttpResponseMessage(HttpStatusCode.NotFound);

            HttpResponseMessage response = CreateResponseForFile(fileReader);
            return response; ①
        }

        [HttpPost]
        public async Task<ActionResult> AddFile([FromForm(Name = "file")]
FormFileCollection formFiles, Guid cardId) ②
        {

            string rootPath = CreateAndGetTempFolder();
            try
            {
                List<string> files = await SaveFilesFromResponse(formFiles); ③

                await fileService.AddFilesToCard(cardId, files);
                return Ok();

            }
            catch (Exception e)
            {
                return Problem(statusCode: (int)HttpStatusCode.InternalServerError,
detail: e.Message);
            }
        }
```

① Создаёт ответ с файлом.

② Добавляет в карточку файлы из запроса.

③ Добавляемые файлы.

В контроллере используется сервис IFileService, реализация которого приведена ниже. Код дополнительных методов смотрите в полном примере на GitHub.

Файл Others/Watermark/WatermarkServerExtension/Services/FileService.cs

```
    public class FileService : IFileService
    {
        private readonly ICurrentObjectContextProvider currentObjectContextProvider;
```

```
        private Guid DOCUMENT_CARD_ID = new Guid("B9F7BFD7-7429-455E-A3F1-94FFB569C794");

        public FileService(ICurrentObjectContextProvider currentObjectContextProvider) {
            this.currentObjectContextProvider = currentObjectContextProvider;
        }

        public async System.Threading.Tasks.Task<IEnumerable<Guid>> AddFilesToCard(Guid
cardID, List<string> files) ①
        {
            var document = GetDocumentCard(cardID); ②
            ObjectContext objectContext = GetObjectContext();

            ILockService lockService = GetLockService(objectContext);

            if (lockService.IsObjectLockedByAnotherUser(document)) {
                throw new Exception($"Карточка {cardID} заблокирована другим
пользователем");
            }

            if (lockService.LockObjectBase(document) == false) {
                throw new Exception($"Не удалось заблокировать карточку {cardID}");
            }

            IDocumentService documentService = GetDocumentService(objectContext);

            return await System.Threading.Tasks.Task.Run(() =>
            {
                IEnumerable<Guid> documentFileIds;
                try
                {
                    IEnumerable<DocumentFile> documentsFiles = documentService
.AddAdditionalFiles(document, files);
                    objectContext.AcceptChanges();
                    documentFileIds = documentsFiles.Select(t => t.FileId);
                }
                catch (Exception ex)
                {
                    throw new Exception($"Ошибка при добавлении файлов в карточку {
cardID}\n {ex.Message}");
                }
                finally {
                    lockService.UnlockObject(document);
                }

                return documentFileIds; ③
            });
        }
```

```
        public FileReader GetFileReader(Guid fileCardID) ④
        {
            ObjectContext objectContext = GetObjectContext();
            IVersionedFileCardService versionedFileCardService =
GetVersionedFileCardService(objectContext);

            VersionedFileCard fileCard = versionedFileCardService.OpenCard(fileCardID);
⑤

            Guid fileID = fileCard.CurrentVersion.Id; ⑥

            UserSession userSession = GetUserSession();

            if (userSession.FileManager.FileExists(fileID) == false)  ⑦
                return new FileReader();

            var file = userSession.FileManager.GetFile(fileID); ⑧

            return new FileReader() { ⑨
                FileID = fileID,
                FileName = file.Name,
                Stream = file.OpenReadStream()
            };
        }
```

① Добавляет файлы с файловой системы в карточку `cardID`.

② Получает карточку, к которой прикрепляются файлы.

③ Возвращает идентификаторы добавленных файлов с версиями.

④ Получает указатель для чтения файла из карточки файла с версиями `fileCardID`.

⑤ Получение файла карточки с версиями.

⑥ Получение идентификатора файла из последней версии.

⑦ Если файла нет, возвращается пустой указатель.

⑧ Запрашивается файл текущей версии.

⑨ Возвращается указатель для чтения файла.

Расширение программы DVWebTool

Когда реализовано расширение Web-клиента, предоставляющее и записывающее файлы карточек, может быть реализовано расширение *DVWebTool*, использующее данные функции.

Расширение *DVWebTool* представляет собой сборку, в которой реализован интерфейс `Docsvision.DVWebTool.WebServices.IServiceManager`. Данный интерфейс определяет метод `Register`, который регистрирует контроллеры с необходимыми функциями во внутреннем веб-сервере *DVWebTool*, и поле `DisplayName` с названием расширения.

Ниже приведён код класса, реализующего интерфейс `IServiceManager` в данном примере.

Файл `Others/Watermark/WatermarkWebToolExtension/WatermarkManager.cs`

```
public class WatermarkManager : IServiceManager
{
    public string DisplayName => "Watermark to PDF"; ①

    public void Register(WebSocketSharper.Server.WebSocketServer server) ②
    {
        server.AddWebSocketService<WatermarkController>("/Watermark"); ③
    }
}
```

① Название расширения для информации в окне "О программе".

② Регистрация контроллера расширения.

③ Регистрация контроллера `PDFWatermarkController` для маршрута `Watermark`.

В данном примере выполняется регистрация контроллера `WatermarkController` для обработки запросов, поступающих по пути `/Watermark`.

Контроллер, передаваемый в `AddWebSocketService`, должен быть производным типа `Docsvision.DVWebTool.WebServices.BaseService`. При его реализации необходимо зарегистрировать в `BaseService.actions` веб-методы, с помощью которых будут вызываться функции *DVWebTool*, предоставляемые расширением. Данные методы будут доступны для вызова по протоколу `WebSocket` по адресу `ws://localhost:/%Адрес контроллера%/%Название метода%`.

Ниже приведена часть реализации контроллера `WatermarkController`, содержащего методы обработки входящих запросов на добавление водяного знака.

Файл `Others/Watermark/WatermarkWebToolExtension/WatermarkController.cs`

```
public class WatermarkController : BaseService
{
    private readonly ServiceProvider serviceProvider;
```

```
        private static readonly Logger Logger = LogManager.GetCurrentClassLogger();

        private string WATERMARK = "Секретно"; ①

        public WatermarkController()
        {
            serviceProvider = new ServiceProvider();
            Init();
        }

        private void Init() ②
        {
            actions.Add(nameof(AddWatermarkToFiles), AddWatermarkToFiles);
        }

        private WebServiceResponse AddWatermarkToFiles(WebServiceRequest
webServiceRequest, JObject data) ③
        {
            Logger.Info("Получено задание на добавление водяного знака");

            if (data == null)
            {
                return CreateBadResponse("С клиента не переданы данные для работы");
            }

            AddWatermarkRequest request; ④
            try
            {
                request = data.ToObject<AddWatermarkRequest>();
            }
            catch
            {
                Logger.Error($"Ошибка преобразования полученного сообщения: {data}");
                return CreateBadResponse("Поступивший запрос не соответствует ожидаемому
формату");
            }

            string doneInfo;

            try
            {
                doneInfo = HandleRequest(request).Result;
            }
            catch (Exception ex)
            {
                return CreateBadResponse(ex.Message);
            }
```

```
            return CreateEndProcessResponse(request.CardID, $"Водяные знаки добавлены в
файлы:<p/>{doneInfo}");
        }


        private async Task<string> HandleRequest(AddWatermarkRequest request) ⑤
        {
            var connectionService = new ConnectionToWebClient(request.ServerAddress,
request.AccessToken);

            List<string> files = new List<string>();

            foreach (var fileId in request.FileIDs) ⑥
            {
                try
                {
                    string pathToFile = await connectionService.PullFile(fileId);
                    files.Add(pathToFile);
                }
                catch (Exception ex)
                {
                    Logger.Error(ex.Message);
                    throw new Exception($"Не удалось получить из карточки файл с
идентификатором {fileId}");
                }
            }

            var watermarkService = new WatermarkService(); ⑦

            List<Task<string>> processes = new List<Task<string>>(); ⑧
            string doneInfo = "";

            foreach (var file in files)
            {
                try
                {
                    processes.Add(watermarkService.AddWatermark(file, WATERMARK));
                    doneInfo += $" {Path.GetFileName(file)}<p/>";
                }
                catch (Exception ex)
                {
                    Logger.Error(ex.Message);
                    throw new Exception($"Не удалось добавить водяной знак в файл {Path
.GetFileName(file)}");
                }
            }

            var filesWithWatermark = await Task.WhenAll(processes);
```

```
            try
            {
                await connectionService.PushFiles(request.CardID, filesWithWatermark); ⑨
            }
            catch (Exception ex)
            {
                Logger.Error(ex.Message);
                throw new Exception($"Не удалось сохранить файлы в карточке");
            }
            return doneInfo;
        }
```

① Текст водяного знака.

② Регистрация методов контроллера `PDFWatermarkController`. Название метода регистрозависимое.

③ Веб-метод добавления водяного знака. Метод должен принимать два параметра: `WebServiceRequest` и `JObject`. Данные передаются в `data`.

④ Загружаем данные из полученного запроса в модель `AddWatermarkRequest`.

⑤ Обработчик запроса на добавление водяного знака. Получает данные запроса. Возвращает строку с названиями файлов, в которые добавлены запросы.

⑥ Загружаем с Web-клиента файлы, идентификаторы которых переданы в запросе.

⑦ Сервис для работы с водяными знаками.

⑧ Добавление водяных знаков в файлы из списка `files`.

⑨ Отправка запроса на прикрепление файлов `filesWithWatermark` к карточке `request.CardID`.

Контроллер `WatermarkController` использует функции двух сервисов:

- `ConnectionToWebClient` для получения файлов из карточек Файл с версиями и загрузки файлов в карточки Документ. Данный сервис использует функции, реализованного серверного расширения Web-клиента.
- `WatermarkService` для добавления водяных знаков в PDF файлы.



Реализации данных сервисов смотрите в исходных кодах примера на GitHub.

После публикации расширения *DVWebTool*, зарегистрированный в расширении метод `AddWatermarkToFiles` контроллера `Watermark` может быть вызван из клиентского расширения Web-клиента с помощью сервиса `DVWebToolConnection`.

Клиентское расширение

В качестве примера использования функций расширения *Watermark to PDF* программы *DVWebTool* было разработано клиентское расширение, которое выполняет две задачи:

- Получает из текущей открытой карточки Документа идентификаторы основных файлов формата PDF.
- Отправляет идентификаторы файлов расширению *Watermark to PDF*.

Общие требования к реализации клиентских расширений приведены в пункте "Расширение возможностей клиентской части Web-клиента".

Прежде всего реализуем сервис, получающий список идентификаторов файлов, в которые нужно добавить водяной знак и передавать его в метод `AddWatermarkToFiles` расширения *Watermark to PDF*. Ниже приведён исходный код данного сервиса.

Файл `Others/Watermark/WatermarkWebExtension/src/WatermarkService.tsx`

```
export class WatermarkService { ①

    constructor(private services: $RequestManager & $WebServices & $ApplicationSettings
        & $MessageBox & $SiteUrl & $CurrentEmployeeId & $JwtTokenController) { ②
    }

    async AddWatermarkToFiles(cardID: string, fileIDs: string[]): Promise
<IWebServicesResponse<any>> { ③

        const accessToken = await this.services.jwtTokenController.getAccessToken(); ④

        const data: any = { ⑤
            data: {
                cardID: cardID,
                fileIDs: fileIDs,
                userID: this.services.currentEmployeeId,
                serverAddress: this.services.siteUrl,
                accessToken
            },
            action: 'AddWatermarkToFiles', ⑥
            locale: this.services.applicationSettings.culture.twoLetterISOLanguageName ⑦
```

```
        };

        return DVWebToolConnection.trySendData("Watermark", data, this.services); ⑧
    }
}

export type $WatermarkService = { watermarkService: WatermarkService }; ⑨
export const $WatermarkService = serviceName((s: $WatermarkService) => s.
watermarkService);
```

① Клиентский сервис, предоставляющий доступ к методу добавления водяных знаков, предоставляемому расширением DVWebTool.

② Метод принимает идентификатор карточки и идентификатор её конвертируемых файлов.

③ Токен доступа, с помощью которого DVWebTool будет выполнять запросы к серверу Web-клиента.



Обратите внимание, время жизни токена ограничено! Подробнее см. ниже.

④ В данных нужно также передать:

- Адрес сервера Web-клиента. DVWebTool должен подключиться к Web-клиенту для получения и сохранения файлов карточки.
- ID пользователя для отправки оповещения о завершении процесса

⑤ Название метода, вызываемого из расширения *DVWebTool*.

⑥ Обязательное для передачи название локали.

⑦ Вызываем метод `AddWatermarkToFiles` из контроллера `Watermark` расширения DVWebTool. Тип `DVWebToolConnection` предоставляет методы для работы с DVWebTool.

⑧ Регистрируем сервис `WatermarkService`.

Сервис `$WatermarkService` предоставляет единственный метод `AddWatermarkToFiles`, который вызывает функцию программы *DVWebTool* с помощью метода `DVWebToolConnection.trySendData`. При вызове данного метода нужно передать название контроллера и данные, в которых должно быть название вызываемого метода контроллера — в поле `action` передаваемых данных.

В качестве метода, использующего сервис `$WatermarkService` реализуем обработчик нажатия кнопки разметки карточки Документ, который будет

получать идентификаторы основных PDF-файлов карточки, вызывать метод `$WatermarkService.AddWatermarkToFiles` и после завершения его работы обновлять содержимое элемента управления Список файлов, или отображать ошибку.

Файл `Others/Watermark/WatermarkWebExtension/src/EventHandlers.tsx`

```
export async function addWatermark(sender: LayoutControl) { ①
   showNotify("Запущен процесс добавления водяных знаков");

   let cardId = sender.layout.getService($CardId); ②
   let files = sender.layout.getService($FileService).getFiles();

   let fileIDs = new Array();

   files.forEach((item) => { ③
      if (item.data.isMain && item.data.name.toLowerCase().endsWith(".pdf")) { ④
         fileIDs.push(item.data.fileId);
      }
   })

   let watermarkService = sender.layout.getService($WatermarkService); ⑤

   let response = await watermarkService.AddWatermarkToFiles(cardId, fileIDs); ⑥

   if (response.success == false) {
      showError(response.errorMessage);
   } else {
      showNotify(response.data.message);

      if (layoutManager.cardLayout == null)
         return;

      let currentCardId = sender.layout.getService($CardId);
      if (currentCardId == cardId) {
         let fileList = layoutManager.cardLayout.controls.get<FileListControl>("fileList
"); ⑦
         await fileList.reloadFromServer();
      }
   }
}
```

① Обработчик для события нажатия иконки добавления водяных знаков.

② Получаем ID текущей карточки и список файлов из элемента `FileList`.

③ Получаем из модели `files` только идентификаторы файлов.

④ Обрабатываем только основные файлы с расширением PDF.

⑤ Получаем реализованный сервис для работы с водяными знаками.

⑥ Вызываем функцию добавления водяных знаков для файлов с ИД из списка `fileIDs`.

⑦ Обновляем список файлов, если есть. Возможно уже открыта другая карточка, но в данном случае это не существенно.

Настройка времени действия токена

Время жизни токена доступа зависит от значений параметров `JwtAccessTokenLifetime` и `JwtClockSkew` в конфигурационном файле модуля.

```
{
"Docsvision": {
    "WebClient": {
      "SettingGroups": {
        "System": {
          "JwtAccessTokenLifetime": 300, ①
          "JwtClockSkew": 300 ②
        }
      }
    }
  }
}
```

① Время жизни токена. По умолчанию `5` минут (`300` секунд).

② Допустимая погрешность часов при проверке времени жизни токена. Значение прибавляется к `JwtAccessTokenLifetime`. По умолчанию `5` минут (`300` секунд).

Сборка проекта

Сборка серверного расширения Web-клиента и расширения DVWebTool.

1. Откройте решение `Samples.sln`.
2. Соберите проект **Other › Watermark › WatermarkServerExtension**.
3. Соберите проект **Other › WatermarkWebToolExtension**.

Сборка клиентской части.

4. Откройте в командной строке папку **Others › Watermark › WatermarkWebExtension**.

5. Выполните команды:

```
npm install
npm update
npm run build:prod
```

Публикация компонентов на сервере Web-клиент.

6. Остановите службу Сервера Web-клиента.
7. Скопируйте папку `SamplesOutput/Content/Modules/WatermarkWebExtension/` в `/lib/docsvision/webclient/Content/Modules`.
8. Скопируйте папку `SamplesOutput/Extensions/WatermarkServerExtension` в `/lib/docsvision/webclient/Extensions`.
9. Для работы пользователей на Windows скопируйте файлы из папки `SamplesOutput/Site/Content/Tools/DVWebTool/Application Files/` в `C:\Program Files (x86)\Docsvision\WebClientToolkit\WebToolBuilder\DVWebTool`.
10. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Регистрация расширения DVWebTool на сервере Web-клиента.

11. Запустите программу Панель управления Web-клиентом.
12. Нажмите кнопку **Собрать DVWebTool**.
13. В диалоге введите адрес публикации Web-клиента, при необходимости установите другие опции, и нажмите кнопку **OK**.
14. Установите собранную утилиту DvWebTool на сервер Web-клиента, см. подробнее "[webclient:admin:webservices-install]" в документации Web-клиента.
15. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Сборка и установка пакета для DvSupService

Для работы пользователей на Linux нужно обеспечить копирование файлов расширения в папку установки DvSupService — `/lib/docsvision/dvsupservice`. Рекомендуемый способ это сделать — собрать deb-пакет[15] и распространить его любым удобным способом:

1. Убедитесь, что собран проект `Other > Watermark > WatermarkWebToolExtension`. В результате в папке `Others/Watermark/DebPackage/Build/DvSupServiceExtension` должны оказаться файлы расширения в папках `Assemblies` и `Services`. Параметр `ExtensionsPath` в конфигурационном файле DVSupService позволяет изменять

папку для расширений, по-умолчанию это `Assemblies`.

2. Скопировать папку `Others\Watermark\DebPackage` на Linux в каталог, где планируется выполнять сборку пакета (расположение папки значения не имеет).
3. Установите необходимые для сборки компоненты следующей командой:

   ```
   sudo apt-get install make debhelper dpkg-dev
   ```

4. Откройте консоль в папке `Build` и соберите пакет с помощью следующей команды:

   ```
   cd DebPackage/Build make
   ```

   В результате в папке `DebPackage` появится файл `dvsupservice-extension_1.0_amd64.deb`.

5. Остановите сервис **dvsupservice** командой:

   ```
   sudo systemctl stop dvsupservice
   ```

6. Распространите и установите на машины пользователей собранный пакет. Для этого можно создать репозиторий, чтобы можно было установить расширение с помощью команды `sudo apt-get install dvsupservice-extension`. См. "Создание простого репозитория". Либо просто скопируйте файл на клиентскую машину и выполните команду:

   ```
   sudo dpkg -i dvsupservice-extension_1.0_amd64.deb
   ```

7. Запустите сервис **dvsupservice** командой:

   ```
   sudo systemctl start dvsupservice
   ```

Проверка примера

Настройте разметку:

1. В программе Конструктор Web-разметок добавьте элемент **Кнопка** в любую разметку просмотра карточки *Документ*.

2. Укажите для события **При щелчке** обработчик addWatermark.
3. Сохраните разметку.
4. Установите или обновите программу *DVWebTool*. См. пункт Установка DVWebTool при помощи WebToolBuilder в руководстве пользователя Web-клиент.
5. Запустите программу *DVWebTool*. Убедитесь, что программа *DVWebTool* и Web-клиент будут запущены от имени одного пользователя.

Убедитесь, что все расширения установлены:

8. В Web-клиенте перейдите в раздел *О программе*.

   В разделе Подключенные расширения должны быть указаны расширения:

   - *WatermarkServerExtension* (Сборка <номер сборки>).
   - *Watermark to PDF <номер сборки>* — web-расширение.

9. Откройте панель *About* из меню Docsvision *DVWebTool*.

   В списке установленных расширений должно быть расширение:

   - *Watermark to PDF <версия>*.

10. Откройте для просмотра любую карточку с разметкой, настроенной в п. 1.
11. Добавьте один или несколько основных файлов с расширением .pdf.
12. Нажмите на добавленную кнопку с обработчиком addWatermark.

Появится сообщение Запущен процесс добавления водяных знаков — начнётся процесс добавления водяных знаков.

После завершения процедуры появится сообщение Водяные знаки добавлены в файлы: "список PDF-файлов, в которые добавлены водяные знаки".

В карточку будут добавлены дополнительные файлы с постфиксом _marked, являющиеся копиями оригинальных файлов с добавленным водяным знаком: слово Секретно.

Особенности обновления

При обновлении версии Web-клиента регистрация расширения в программе *DVWebTool* будет отменена (конфигурационные файлы программы перезаписываются).

После установки новой версии Web-клиента необходимо повторно зарегистрировать расширения *DVWebTool* на сервере Web-клиент.

Web-сервис для взаимодействия с платформой

Данный раздел содержит демонстрационный вариант Web-сервиса, содержащего пример взаимодействия с платформой Docsvision.

Ссылка на пример на GitHub: ExternalWebService.

Web-сервис позволяет выполнять следующие действия с платформой Docsvision:

1. Получите информацию о карточке документа по её идентификатору.
2. Создайте карточку документа по переданной модели.
3. Обновите данные карточки документа.
4. Удалите карточку документа по её идентификатору.
5. Измените состояние карточки документа.
6. Прикрепите файл к карточке документа.
7. Получите файл по его идентификатору.
8. Получите результат выполнения расширенного отчета.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Откройте `/Samples.sln`
2. Соберите проект `Others > ExternalWebService > WebService`
3. Соберите проект `Others > ExternalWebService > WebServiceClient`
4. На сервере Docsvision установите Docsvision Resource Kit. С помощью утилиты DVCardManager загрузите в БД Docsvision библиотеку `ReportsLibrary` из каталога `ExternalWebService\CardDefs\ReportsLibrary`.

   Подробное описание процедуры загрузки пользовательской библиотеки карточек содержится в руководстве разработчика Docsvision.

5. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Проверка примера

1. Отредактируйте в файле SamplesOutput\ExternalWebService\WebService\appsettings.json параметры ServerAddress и BaseName так, чтобы они указывали на рабочий сервер Docsvision и базу.
2. Запустите службу Сервера Web-клиента из папки SamplesOutput\ExternalWebService\WebService.
   a. На Windows запустите WebService.exe из под учётной записи Docsvision Web-клиента.
   b. На Linux запустите командой dotnet WebService.dll
   c. Для запуска под нужной учётной записью можно задать параметры SystemUserAccount и SystemUserPassword в appsettings.json.
3. Запустите приложение SamplesOutput\ExternalWebService\WebServiceClient\WebServiceClient.exe. В выводе консоли отобразится результат выполнения операций.

Каталог "CardDefs"

Содержит описание библиотеки ReportsLibrary с одним расширенным отчетом, который возвращает описание карточки (поле description) по её идентификатору.

```
<?xml version="1.0" encoding="utf-8"?>
<CardLibrary xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" ID="D030376C-1F99-445B-9455-
01538CB11461" Alias="ReportsLibrary" ControlInfo="" IconFile="" Version="1"
MsiProductCode="" MsiPackageName="">
  <Name>
    <LocalizedString Language="en">ReportsLibrary</LocalizedString>
    <LocalizedString Language="ru-RU">ReportsLibrary</LocalizedString>
  </Name>
  <Reports>
    <Report Alias="TestReport" ID="85701F4B-61FF-4A33-ADD1-959B471E7558" SQLFile="">
      <Name>
        <LocalizedString Language="en">TestReport</LocalizedString>
      </Name>
      <Parameter Type="uniqueidentifier" Name="CardId" />
      <SqlFiles>
        <SqlFile ServerType="mssql">Sql\TestReport.sql</SqlFile>
        <SqlFile ServerType="pgsql">PgSql\TestReport.sql</SqlFile>
      </SqlFiles>
      <Result>
        <ResultSet Name="dvtable_report_result_{85701f4b-61ff-4a33-add1-959b471e7558}">
```

```
          <Column Name="Description" DataType="String" Size="255" Nullable="true" />
        </ResultSet>
      </Result>
    </Report>
  </Reports>
  <Log>
    <Name />
  </Log>
</CardLibrary>
```

Проект "WebService.Interfaces"

Проект содержит общие модели для веб-сервиса и клиентского приложения.

Проект "WebService"

Предоставляет API для взаимодействия с сервером Docsvision. Содержит сервис `DocumentService` для работы с карточками типа `Document`.

Также включает в себя `ReportService`, содержащий метод для выполнения расширенного отчета.

Проект "WebServiceClient"

Содержит пример консольной утилиты для взаимодействия с веб-сервисом посредством WebApi.

Интеграция с сервисом Контур.Фокус

Данный раздел содержит описание примера интеграции с сервисом Контур.Фокус.

Ссылка на пример на GitHub: KonturIntegration.

Пример включает два компонента:

- `KonturServerExtension` — папка с серверным расширением Web-клиента, в котором реализовано проксирование запросов к Контур.Фокус (для обхода ограничений CORS в браузере).
- `KonturWebExtension` — папка с клиентским расширением, в котором реализовано открытие отчета о контрагенте и загрузка информации.
- `KonturSolution.xml` — решение с разметкой для организации контрагента, содержащее разметку с необходимыми обработчиками и элементами управления.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Сборка и установка

1. Сборка серверной части.
   a. Откройте решение `Samples.sln`.
   b. Соберите проект `Other > Kontur > Kontur`.
2. Сборка клиентской части.
   a. Откройте в командной строке папку `Others > KonturIntergration > KonturWebExtension`.
   b. Выполните команды:

```
npm install
npm update
npm run build:prod
```

3. Публикация компонентов на сервере Web-клиента.
   a. Скопируйте папку `SamplesOutput\Content\Modules\KonturWebExtension\` в `Путь к сайту Web-клиента\Content\Modules`.
   b. Скопируйте папку `SamplesOutput\Extensions\KonturServerExtension` в `Путь к сайту Web-клиента\Extensions`.
   c. Перезапустите Web-сервис.

Проверка примера

При создании контрагента

1. В программе Конструктор Web-разметок импортируйте решение `KonturSolution.xml`.
2. В `appsettings.json` Web-клиента добавьте в корневой элемент настройку с ключом доступа к API Контур.Фокус:

```
{
"KonturSettings": {
```

```
  "ApiKey": "..."
}
}
```

3. Откройте разметку с ЭУ `PartnerDepartment`, например, создание исходящего документа, откройте окно справочника и нажмите кнопку добавления организации.
4. В открывшейся разметке введите ИНН организации, например, `6663003127`.
5. Нажмите кнопку **Просмотреть отчет о контрагенте** — откроется окно с информацией о контрагенте.
6. Нажмите на кнопку **Загрузить информацию** — значения элементов управления на разметке заполнятся данными из Контур.Фокус.

В карточке документа

1. Добавьте разметку карточки типа документ кнопку, назначьте ей обработчик `openBriefReportAndAttachFile`.
2. В ту же разметку добавьте ЭУ с именем `INN` типа `TextBox`. Настройте его таким образом, что бы он содержал актуальный ИНН контрагента, например, через расширенные источники данных.
3. Откройте документ и нажмите на кнопку, откроется отчет в модальном окне и к документу будет прикреплен файл отчета в качестве дополнительного файла.

Компонент "KonturServerExtension"

```
namespace Kontur
{
    public class KonturServerExtension : WebClientExtension ①
    {

        public KonturServerExtension(IServiceProvider serviceProvider) ②
            : base()
        {
        }

        public override string ExtensionName ③
        {
            get { return Assembly.GetAssembly(typeof(KonturServerExtension)).GetName
().Name; }
        }
```

```
        public override Version ExtensionVersion ④
        {
            get { return new Version(FileVersionInfo.GetVersionInfo(Assembly
.GetExecutingAssembly().Location).FileVersion); }
        }

        #region WebClientExtension Overrides

        public override void InitializeServiceCollection(IServiceCollection services)
        {
            services.AddSingleton<IKonturRequestService, KonturRequestService>();
            services.AddOptions<KonturSettings>().BindConfiguration(KonturSettings.Key);

        }

        protected override List<ResourceManager> GetLayoutExtensionResourceManagers() ⑤
        {
            return new List<ResourceManager>
            {

            };
        }

        #endregion
    }
}
```

① Задаёт описание расширения для Web-клиента, которое задано в текущей сборке.

② Создаёт новый экземпляр, см. `KonturServerExtension`.

- `serviceProvider` — сервис-провайдер.

③ Получить название расширения.

④ Получить версию расширения.

⑤ Получает менеджеры ресурсов для расширения конструктора разметок.

Отправка оповещений пользователям

Для отправки оповещений пользователям Web-клиента могут быть использованы функции SignalR, реализованные в сервисе `IRealtimeCommunicationService`. Данный сервис позволяет отправлять оповещения всем пользователям или определённому пользователю.

Ссылка на пример на GitHub: SignalForUsers.

Не следует путать оповещение с функцией отображения диалоговых окон (пункт "Отображение сообщений, запросов, предупреждений").

Текстовые оповещения отображаются в правом нижнем углу окна Web-клиента.

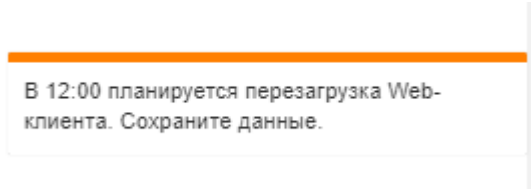


Рисунок 713. Пример оповещения



На стороне клиента также предоставляется доступ к механизму оповещения, позволяющий передавать сообщения между вкладками браузера и на сервер Web-клиента. Данная функциональность реализована в клиентском сервисе `$RealtimeCommunicationService` (см. описание в JsDocApi).

В качестве примера работы с сервисом IRealtimeCommunicationService реализуем функцию рассылки сообщений всем пользователям, доступную на стороне клиента.



Полный код примера доступен в репозитории на GitHub.

Реализация серверного метода для рассылки оповещений

Во-первых, нужно реализовать серверный метод, использующий функции сервиса `IRealtimeCommunicationService` для рассылки оповещений.

Для этого разработаем серверное расширение с web-функцией. Ниже приведён код такой функции, реализованной в контроллере.

Файл `Controllers\ServiceController.cs`

```
using DocsVision.Platform.WebClient.Models.RealTimeCommunication;
using DocsVision.Platform.WebClient.Models.RealTimeCommunication.NotificationMessage;
using DocsVision.Platform.WebClient.Services;
using System;
using Microsoft.AspNetCore.Mvc;

namespace SignalForUsersExtension
{
    public class ServiceController : ControllerBase
```

```
    {
        readonly IRealtimeCommunicationService communicationService;

        public ServiceController(IRealtimeCommunicationService communicationService) ①
        {
            this.communicationService = communicationService;
        }

②
        [HttpPost]
        public void SendAll([FromBody]SendMessageRequest request)
        {
            var commMessage = new RealtimeCommunicationMessage
<NotificationRealtimeMessage>();
            var messageData = new NotificationRealtimeMessage()
            {
                NotificationType = request.MessageType,
                Message = request.Message
            };
            commMessage.Initialize(NotificationRealtimeMessage.MessageType, messageData);
            communicationService.SendAll(commMessage);
        }
    }
}
```

① Сервис IRealtimeCommunicationService получаем из конструктора.

② Отправляет оповещение всем пользователям
<param name="request">Сообщение</param>

Для удобства использования методов серверного расширения разработаем клиентское расширение с сервисом, который будет вызывать web-методы.

Файл `src\MessageService.tsx`

```
import { serviceName } from "@docsvision/webclient/System/ServiceUtils";
import { $RequestManager } from "@docsvision/webclient/System/$RequestManager";
import { urlStore } from "@docsvision/webclient/System/UrlStore";

export class MessageService { ①
    constructor(private services: $RequestManager) {
    }

    sendAll(message: string, messageType: MessageType) { ②
        let url = urlStore.urlResolver.resolveApiUrl("SendAll", "Service");

        this.services.requestManager.post(url, JSON.stringify({ message, messageType }));
```

```
    }
}

export type $MessageService = { messageService: MessageService };
export const $MessageService = serviceName((s: $MessageService) => s.messageService);
```

① Клиентский сервис, предоставляющий доступ к серверным методам работы с подсистемой оповещений.

② Отправляет сообщение message всем пользователям. Тип сообщения `messageType` влияет на оформление сообщения.

Здесь `messageType` — ограниченный список значений, которые может принимать тип сообщения:

```
export enum MessageType {
    Alert = 0,
    Success = 1,
    Warning = 2,
    Error = 3,
    Info = 4
}
```

Осталось добавить функцию, которая будет обрабатывать, к примеру, нажатие кнопки разметки и вызывать метод `MessageService.sendAll`:

Файл `src\EventHandlers.tsx`

```
import { IEventArgs } from "@docsvision/webclient/System/IEventArgs";
import { Layout } from "@docsvision/webclient/System/Layout";
import { $MessageService, MessageType, MessageService } from "./MessageService";
import { layoutManager } from "@docsvision/webclient/System/LayoutManager";

export function sendMessage(sender: Layout, e: IEventArgs) { ①
   let messageService = layoutManager.cardLayout.getService<MessageService>
($MessageService);
   return messageService.sendAll("В 12:00 планируется перезагрузка {wc}а. Сохраните
данные.", MessageType.Warning);
}
```

① Функция использует сервис `$MessageService` (проект) для отправки оповещения всем пользователям.

Функция `sendMessage` может быть указана в качестве обработчика нажатия кнопки или другого события.

Реализация собственного штампа электронной подписи

Реализация не поддерживается в текущей версии системы.

Устаревшие примеры

Руководство по разработке решений Web-клиента

Устаревшие примеры

Данные примеры помечены как устаревшие и неактуальны.

- [WebClient:obsolete-hyper-comments]
- [WebClient:obsolete-sample-office-work]
- Больше примеров категории в навигационном меню слева…

Пример текста

Данный раздел содержит описание примера реализации элемента управления `HyperComments`.



Платформа `HyperComments` перестала поддерживать бесплатный тариф. Необходимо приобретение лицензии.

Пример требует подключенного расширения программы Конструктор Web-разметок `UrlProprtyDesignerExtension`.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Проект "HyperCommentsWebExtension"

Проект-расширение клиентской части Web-клиента. Содержит пример одного из способов интеграции со сторонними веб-приложениями.

Проект "HyperCommentsDesignerExtension"

Проект-расширение для конструктора разметок. Содержит описание элемента управления `HyperComments` для программы Конструктор Web-разметок.

Демонстрирует описание и подключение нового ЭУ, используя класс `ControlTypeDescription`, задание ему стандартных свойств, реализованных в программе Конструктор Web-разметок[16] и добавление собственного свойства, используя `PropretyDescription`, класс[17].

Сборка и установка

1. Откройте `/Samples.sln`.
2. Соберите проект `Controls > HyperComments > HyperCommentsDesignerExtension`.
3. Откройте консоль в папке `Controls > HyperComments > ExchangeRatesWebExtension` и выполните команду `npm install` и `npm run build:prod`.
4. Скопируйте каталог `SamplesOutput\Plugins\HyperCommentsDesignerExtension` в каталог `Путь к директории с исполняемым файлом конструктора Web-разметок\Plugins`.
5. Скопируйте каталог `SamplesOutput\Content\Modules\HyperCommentsWebExtension` в каталог `Путь к сайту Web-клиента\Content\Modules`.
6. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.

Регистрация виджета "HyperComments"

1. Перейдите на сайт https://www.hypercomments.com.
2. Нажмите на кнопку **Войти**.
3. Авторизоваться под аккаунтом Google.
4. Выбрать лицензию и следовать инструкциям.
5. Нажмите на кнопку **Установить**.
6. Ввести адрес сайта (Допустимо имя сайта в локальной сети).
7. Выбрать тип виджета - HTML Custom code.
8. Нажмите **Далее**.
9. Нажмите **Готово**.
10. Перейдите на страницу *Виджет — Общее*.
11. Под заголовком указан идентификатор виджета ID: `######`, его следует запомнить, он потребуется далее.
12. Установите флаг напротив настройки `Get-параметры`. Учитывайте GET параметры и сгенерируйте новый виджет комментирования для отдельной страницы.

13. Нажмите на кнопку **Сохранить**.

Проверка примера

1. Запустите конструктор разметок.
2. Скопируйте любую разметку для карточки вида *Служебная записка*.
3. Выберите условия использования этой разметки.
4. Откройте разметку и добавить в нее новый элемент управления `Hyper comments`.
5. Укажите в свойстве `Widget id` идентификатор из пункта выше.
6. Сохраните разметку.
7. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.
8. Откройте карточку с этой разметкой.
9. Убедитесь, что появился новый элемент управления.

Пример. Элемент управления для выбора из Справочника номенклатуры дел

В данном пункте рассмотрено создание элемента управления для выбора дела из Справочника номенклатуры дел.

В примере рассмотрены следующие вопросы разработки элементов управления:

- Создание расширения программы Конструктор Web-разметок, решающего следующие задачи:
  - Ограничение выбора поля карточки со значением элемента управления — поле должно содержать ссылку на строку *Справочника номенклатуры дел*.
  - Возможность ограничить для пользователя область выбора дел — настройки элемента управления должны содержать параметр для выбора *Раздела*, из которого можно выбирать дела.
- Создание серверного расширения Web-клиент, который должен предоставлять:
  - Сервис для работы с данными Справочника номенклатуры дел (с использованием базового API Docsvision).
  - Конвертер значений, возвращающий отображаемое значение элемента управления.
- Создание клиентского расширения с несколькими функциями:

- Реализация пользовательского интерфейса, одним из элементов которого должно быть дерево разделов Справочника номенклатуры дел.
- Реализация функции ограничения выбора дел из раздела Справочника номенклатуры дел 5, указанного в программе Конструктор Web-разметок.
- Предоставление возможности быстрого поиска дел.



Полный исходный код примера приведён в репозитории Docsvision Samples на GitHub, проект `Controls/RefCases`.

Создание описателя элемента управления и расширения программы Конструктор Web-разметок

Прежде всего определим перечень настроек разрабатываемого элемента управления, которые должны быть доступны пользователю программы Конструктор Web-разметок:

- Стандартный набор свойств: текст метки, подсказка, обязательные и пр..
- Свойства для выбора источника данных.
- Специальное свойство для ограничения области выбора дел.
- Стандартный набор событий (при наведении/отведении курсора и при смене данных).

Описатель элемента управления

Для описания данного элемента управления сформируем текстовый описатель со следующим содержимым:

Файл `RefCategoriesDesignerExtension\xml\RefCasesControlDescription.xml`

```
<?xml version="1.0" encoding="utf-8" ?>
<Controls>
  <Control Name="RefCases" ControlGroupResourceKey="ControlGroup_Directories"
ResourceKey="Control_RefCases">
    <Properties>
①
      <Property Type="Name" /> ②
      <Property Type="PlaceHolder" /> ③
      <Property Type="Tip" /> ④
      <Property Type="LabelText" />
      <Property Type="ShowEmptyLabel" />
      <Property Type="Visibility" />
      <Property Type="StandardCssClass" DefaultValue="ref-cases" />
      <Property Type="CustomCssClasses" />
      <Property Type="TabStop" DefaultValue="True" />
```

```
      <Property Type="Required" />

⑤
      <Property Type="ExtendedDataSource" /> ⑥
      <Property Type="DataSource" /> ⑦
      <Property Type="DataField" Editor="RefCasesFieldMetadataEditor" />
      <Property Type="Binding" BindingConverter="RefCasesConverter" /> ⑧
      <Property Type="EditOperation" /> ⑨

      <Property Name="RootSection" ResourceKey="Control_RefCases_Section"
                Category="BehaviorCategory" DataType="System.Guid"
                Editor="RefCasesSectionEditor" /> ⑩
    </Properties>
    <Events> ⑪
      <Event Name="Click" ResourceKey="ControlTypes_ClickEventProperty" />
      <Event Name="MouseOver" ResourceKey="ControlTypes_MouseOverEventProperty" /> ⑫
      <Event Name="MouseOut" ResourceKey="ControlTypes_MouseOutEventProperty" /> ⑬
      <Event Name="DataChanged" ResourceKey="ControlTypes_DataChangedEventProperty" /> ⑭
    </Events>
  </Control>
</Controls>
```

① Стандартный набор свойств:

② **Название**;

③ **Заполнитель**;

④ **Подсказка**.

⑤ Свойства для настройки источника данных и операции редактирования:

⑥ `ExtendedDataSource` — выбор расширенного источника данных;

⑦ `DataSource`, `DataField` — выбор секции и поля с данными элемента управления. Редактор `RefCasesFieldMetadataEditor` (будет разработан далее) реализует заявленную функциональность ограничения полей карточки, доступных для выбора;

⑧ Binding — системное свойство, с которым на клиент передаётся связь с источником данных.

Значение элемента управления — идентификатор строки дела, выбранного в *Справочнике номенклатуры дел*. Пользователю предпочтительно показывать название дела из справочника, а не идентификатор.

Такое преобразование данных осуществляется с помощью конвертера, название которого указывается в атрибуте `BindingConverter`. В данном примере

название конвертера `RefCasesConverter` (будет разработан далее);

⑨ `EditOperation` — выбор операции редактирования.

⑩ Новое свойство `RootSection` обеспечивает заявленную возможность выбора разделе *Справочника номенклатуры дел*, из которого пользователю будет разрешено выбирать дела.

Значение свойства будет устанавливаться с помощью специального редактора `RefCasesSectionEditor` (будет разработан далее в серверном расширении Web-клиента).

⑪ Также для элемента управления определены события:

⑫ При наведении курсора;

⑬ При отведении курсора;

⑭ При смене данных.

Расширение программы Конструктор Web-разметок

В описателе элемента управления определено несколько нестандартных редакторов, а также ресурсы с текстовыми значениями. Данные сущности нужно добавить в Конструктор Web-разметок с помощью расширения, код которого приведён ниже.

Файл `RefCategoriesDesignerExtension\Extension\RefCasesDesignerExtension.cs`

```
using DocsVision.Platform.Tools.LayoutEditor.Extensibility;
using RefCasesDesignerExtension.Editors;
using System;
using System.Collections.Generic;
using System.Resources;

①
namespace RefCasesDesignerExtension.Extension
{
    class RefCasesDesignerExtension : WebLayoutsDesignerExtension
    {
        public RefCasesDesignerExtension(IServiceProvider provider)
            : base(provider)
        {
        }

        protected override Dictionary<string, Type> GetEditors()
        {
            return new Dictionary<string, Type>
```

```
            {
                { "RefCasesFieldMetadataEditor", typeof(RefCasesFieldMetadataEditor)}, ②
                { "RefCasesSectionEditor", typeof(RefCasesSectionEditor)} ③
            };
        }

        protected override List<ResourceManager> GetResourceManagers()
        {
            return new List<ResourceManager>
            {
                Resources.ResourceManager
            };
        }
    }
}
```

① В данном расширении регистрируется два редактора, которые указаны в текстовом описателе:

② `RefCasesFieldMetadataEditor` — редактор для выбора поля карточки, содержащего значение элемента управления;

③ `RefCasesSectionEditor` — редактор для выбора раздела *Справочника номенклатуры дел*, из которого пользователям будет разрешено выбирать дела.

Редактор "RefCasesFieldMetadataEditor"

Редактор `RefCasesFieldMetadataEditor` является стандартным редактором редактором для выбора поля карточки, реализованном в классе `Docsvision.BackOffice.WebLayoutsDesigner.Editors.FieldMetadataEditor`.

Файл `RefCategoriesDesignerExtension\Editors\RefCasesFieldMetadataEditor.cs`

```
using System.Windows;
using DocsVision.Platform.Data.Metadata.CardModel;
using System;
using Xceed.Wpf.Toolkit.PropertyGrid;
using Xceed.Wpf.Toolkit.PropertyGrid.Editors;
using DocsVision.BackOffice.WebLayoutsDesigner.Editors;
using DocsVision.Platform.WebLayoutsDesigner.NewEditors;

namespace RefCasesDesignerExtension.Editors
{
    public class RefCasesFieldMetadataEditor : ITypeEditor ①
    {
        public FrameworkElement ResolveEditor(PropertyItem propertyItem)
```

```
        {
            var refCasesID = new Guid("246197EA-846A-44DA-9EA3-0BCAE5500388");
            var sectionCasesID = new Guid("56AF8231-B918-42D4-AC15-90EC2E9A0725");

            var editor = new FieldMetadataEditor();

            editor.FieldFilter = (field) => { ②
                return (field.FieldType == FieldType.RefId
                        && field.LinkedCardTypeId == refCasesID
                        && field.LinkedSectionId == sectionCasesID);
            };

            return editor.ResolveEditor(propertyItem);
        }

    }
}
```

① Редактор для выбора поля карточки, ссылающегося на Дело в Справочнике номенклатуры дел.

② Устанавливаем фильтр для выбора полей только из справочника.

Ограничение возможности выбора полей карточки включено с помощью фильтра `FieldFilter`, в котором проверяется тип поля (`field.FieldType`), которое должно быть ссылочным полем (`FieldType.RefId`), ссылающимся на секцию *Дела* (`field.LinkedSectionId == sectionCasesID`) *Справочника номенклатуры дел* (`field.LinkedCardTypeId == refCasesID`).

В стандартной реализации приложения *Делопроизводство 5* поле карточки, используемое для хранения ссылки на *Дело*, не является ссылочным, поэтому для него не подходит фильтр, использованный в данном примере, при настройке разметки данное поле будет недоступно для выбора. Если в Web-клиенте нужно повторить настройки разметки Windows-клиента, фильтр нужно изменить следующим образом:

```
var editor = new FieldMetadataEditor();

editor.FieldFilter = (field) => {
    return (field.FieldType == FieldType.RefId
        && field.LinkedCardTypeId == refCasesID
        && field.LinkedSectionId == sectionCasesID);
};
```

Редактор "RefCasesSectionEditor"

Редактор `RefCasesSectionEditor` имеет более сложную реализацию (по сравнению с `RefCasesFieldMetadataEditor`), из-за необходимости отображения дерева *Разделов Справочника номенклатуры дел 5*.

У данного редактора есть две составляющие:

- Графическая — предоставляет форму для выбора Разделов.
- Функциональная — предоставляет функции, загружающие дерево Разделов и выполняющие сопутствующие операции.



В простейшем случае можно обойтись без сложного редактора, и предоставить возможность непосредственно вводить идентификатор требуемого *Раздела* справочника в значение настройки элемента управления, или отказаться от данной настройки.

Графическая и функциональные составляющие также распределяются между двумя компонентами:

- Основной компонент редактора с реализацией интерфейса `ITypeEditor`.
- Форма с деревом Разделов.

Далее приведён код основного компонента редактора без графической составляющей (см. полный исходный код примера на GitHub).

```
using DocsVision.Platform.Tools.LayoutEditor;
using DocsVision.Platform.Tools.LayoutEditor.PropertiesEditor;
using DocsVision.Platform.WebClient;
using System;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using Xceed.Wpf.Toolkit.PropertyGrid;
using Xceed.Wpf.Toolkit.PropertyGrid.Editors;

namespace RefCasesDesignerExtension.Editors
{
    public partial class RefCasesSectionEditor : UserControl, ITypeEditor
    {
        public static readonly DependencyProperty ValueProperty = DependencyProperty
.Register("Value", typeof(Guid), typeof(RefCasesSectionEditor),①
        new FrameworkPropertyMetadata(Guid.Empty, FrameworkPropertyMetadataOptions
```

```
.BindsTwoWayByDefault));
        public static readonly DependencyProperty TextProperty = DependencyProperty
.Register("Text", typeof(string), typeof(RefCasesSectionEditor), new
FrameworkPropertyMetadata(string.Empty));

        public Guid Value ②
        {
            get { return (Guid)GetValue(ValueProperty); }
            set { SetValue(ValueProperty, value); }
        }

        public string Text ③
        {
            get { return (string)GetValue(TextProperty); }
            set {
                SetValue(TextProperty, value);
                Clear.Visibility = value != ""? Visibility.Visible: Visibility.Collapsed;
// Кнопка очистки значения ④
            }
        }

        private IServiceProvider serviceProvider;
        private SessionContext sessionContext;

        public RefCasesSectionEditor()
        {
            InitializeComponent();
        }

        public FrameworkElement ResolveEditor(PropertyItem propertyItem) ⑤
        {
            var bindingObject = (IControlPropertiesObject)propertyItem.Instance;

            this.serviceProvider = bindingObject.ServiceProvider; ⑥
            var currentObjectContextProvider = this.serviceProvider.GetService(typeof
(ICurrentObjectContextProvider)) as ICurrentObjectContextProvider;
            this.sessionContext = currentObjectContextProvider
.GetOrCreateCurrentSessionContext();

            Binding binding = new Binding("Value"); ⑦
            binding.Source = propertyItem;
            binding.Mode = propertyItem.IsReadOnly ? BindingMode.OneWay : BindingMode
.TwoWay;
            BindingOperations.SetBinding(this, RefCasesSectionEditor.ValueProperty,
binding);

            if (this.Value != Guid.Empty) ⑧
                this.Text = new RefCasesUtils(sessionContext).GetSectionTitle(this.
```

```
Value);

            return this;
        }

        private void ShowSections_Click(object sender, RoutedEventArgs e) ⑨
        {
            var sectionTree = new SectionsTree(sessionContext);
            if (sectionTree.ShowDialog() == true)
            {
                this.Value = sectionTree.SelectedNodeID;
                this.Text = sectionTree.SelectedNodeText;
            }
        }

        private void Clear_Click(object sender, RoutedEventArgs e) ⑩
        {
            this.Value = Guid.Empty;
            this.Text = "";
        }
    }
}
```

① Объявляем свойства зависимости для связывания со значением настройки (идентификатор *Раздела*) и отображаемым значением.

② Идентификатор выбранного *Раздела* справочника — является значение настройки.

③ Название выбранного *Раздела* справочника, отображаемое в строке настройки.

④ Кнопка очистки значения.

⑤ Реализация метода `ITypeEditor.ResolveEditor`.

⑥ Получаем поставщика сервисов из элемента управления.

⑦ Связываем значение компонента с `ValueProperty`.

⑧ Получаем отображаемое значение выбранного *Раздела* при загрузке элемента.

⑨ Открываем форму для выбора *Раздела*.

⑩ Очищаем значение настройки.

Основные функции прокомментированы в коде. При реализации нового редактора (в данном случае не используются готовые реализации, как в

редакторе `RefCasesFieldMetadataEditor`) особое внимание следует обратить на необходимость связывания значение настройки — `Value` — со свойством зависимости (в данном примере — `ValueProperty`).

Метод `ShowSections_Click` вызывается при нажатии кнопки выбора значения свойства. Данный метод открывает форму `SectionsTree` с деревом Разделов *Справочника номенклатуры дел 5* (будет разработан далее).

Метод `Clear_Click` вызывается при нажатии кнопки очистки значения настройки.

Код вспомогательного метода `RefCasesUtils.GetSectionTitle`, возвращающего текстовое описание для Раздела справочника, идентификатор которого передан в метод:

Файл `RefCategoriesDesignerExtension\Editors\RefCasesUtils.cs`

```
public string GetSectionTitle(Guid sectionId)
{
    SectionData sectionSesction = cardManager.GetDictionaryData(refCasesId).Sections
[sectionsSectionID]; ① ② ③

    if (sectionSesction.RowExists(sectionId) == false) ④
        return "Ошибка!";

    RowData sectionRow = sectionSesction.GetRow(sectionId);
    RowData yearRow = sectionRow.SubSection.ParentRow;

    if (yearRow != null)
        return string.Format("{0}. {1}", yearRow["Year"], sectionRow["Name"].ToString());

    return "Ошибка!";
}
```

① `cardManager` — Менеджер карточек (базовое API Docsvision).

② `refCasesId` — идентификатор *Справочника номенклатуры дел 5*.

③ Возможно раздел был удалён.

④ `sectionsSectionID` — идентификатор секции "Разделы" справочника.

Далее приведён код компонента дерева *Разделов* без графической составляющей (см. полный исходный код примера на GitHub).

Файл RefCategoriesDesignerExtension\Editors\SectionsTree.xaml.cs

```
using DocsVision.Platform.WebClient;
using System;
using System.Windows;

namespace RefCasesDesignerExtension.Editors
{

    public partial class SectionsTree : Window
    {
        public string SelectedNodeText = "";
        public Guid SelectedNodeID = Guid.Empty;
        private RefCasesUtils refCasesUtils;

        public SectionsTree(SessionContext sessionContext)
        { ①
            InitializeComponent();
            refCasesUtils = new RefCasesUtils(sessionContext);

            Years.ItemsSource = refCasesUtils.GetYears(); ②
        }

        private void Years_SelectionChanged(object sender, System.Windows.Controls
.SelectionChangedEventArgs e) ③
        {
            if (Years.SelectedIndex == -1 || !(Years.SelectedItem is Year))
                return;

            Year selectedYear = Years.SelectedItem as Year;

            Sections.ItemsSource = refCasesUtils.GetSections(selectedYear.ID); ④
        }

        private void Accept_Click(object sender, RoutedEventArgs e) ⑤
        {
            if (Sections.SelectedItem == null)
                return;
            var selectedNode = Sections.SelectedItem as Node;
            var selectedYear = Years.SelectedItem as Year;

            this.SelectedNodeText = string.Format("{0}. {1}", selectedYear.Value,
selectedNode.Name);
            this.SelectedNodeID = selectedNode.ID;

            this.DialogResult = true;
            this.Close();
        }
```

```
        private void Cancel_Click(object sender, RoutedEventArgs e) ⑥
        {
            this.Close();
        }
    }
}
```

① Для формирования дерева *Разделов* нужно получить данные из *Справочника номенклатуры дел 5*. Для этого объявляем необходимость передачи контекста сессии в конструкторе класса.

② Получаем список лет из Справочника номенклатуры дел 5.

③ При выборе Года формируем дерево Разделов для данного года.

④ Получаем список Разделов из Справочника номенклатуры дел 5.

⑤ Обработка нажатия кнопки сохранения выбора.

⑥ Обработка нажатия отмены.

Основная "работа" здесь выполняется методами `RefCasesUtils.GetYears`, `RefCasesUtils.GetSections`.

Метод `RefCasesUtils.GetYears` получает все строки из секции Года *Справочника номенклатуры дел 5*:

Файл `RefCategoriesDesignerExtension\Editors\RefCasesUtils.cs`

```
public IEnumerable<Year> GetYears()
{
    refCasesData = cardManager.GetDictionaryData(refCasesId);
    SectionData yearSection = refCasesData.Sections[yearsSectionId];

    return yearSection.Rows.Select<RowData, Year>(row => new Year() { ID = row.Id, Value
= row["Year"].ToString() });
}
```

Метод `RefCasesUtils.GetSections` получает дерево строк для секции Разделы *Справочника номенклатуры дел 5*.

Файл `RefCategoriesDesignerExtension\Editors\RefCasesUtils.cs`

```
public List<Node> GetSections(Guid yearID)
{
    var yearSection = refCasesData.Sections[yearsSectionId];
```

```
    if (yearSection.RowExists(yearID)) {
        RowDataCollection sectionRows = yearSection.GetRow(yearID).ChildSections
[sectionsSectionID].Rows;

        return GetNodesFromRows(sectionRows);
    }

    return new List<Node>();
}

List<Node> GetNodesFromRows(RowDataCollection rows) ①
{
    var nodes = new List<Node>();

    foreach (var row in rows)
    {
        var node = new Node() { ID = row.Id, Name = row["Name"].ToString() };
        if (row.HasChildRows)
            node.Nodes = GetNodesFromRows(row.ChildRows);

        nodes.Add(node);
    }
    return nodes;
}
```

1. Возвращает список Разделов для строк секции справочника

Создание серверного расширения Web-клиента

В описателе элемента управления был объявлен конвертер для значения элемента — `RefCasesConverter` (свойство `Binding`), который должен приводить настоящее значение (идентификатор строки *Дела* в *Справочнике номенклатуры дел 5*) к виду, удобному для пользователя — в данном примере к строке вида: `Год. Название разделов. Название дела`.

Также для работы клиентской части понадобятся методы, предоставляющие список лет, дерево разделов и список дел из *Справочника номенклатуры дел 5*, а также функцию для поиска дела по названию (если нужно реализовать функцию поиска на клиенте).

Сервис для работы со Справочником номенклатуры дел 5

Прежде всего необходимо реализовать сервис для работы со *Справочником номенклатуры дел 5*, методы которого должны:

- Возвращать список лет.
- Возвращать год для определённой секции — нужно, если в программе Конструктор Web-разметок был выбран раздел, из которого выбираются дела.
- Возвращать все разделы для определённого года, а если требуется, только разделы из корневого раздела, выбранного в программе Конструктор Web-разметок.
- Возвращать все дела из определённого раздела.
- Возвращать отображаемое название дела — для конвертера значений `RefCasesConverter`.
- Возвращать список дел с искомым названием, а если требуется, только дела из корневого раздела, выбранного в программе Конструктор Web-разметок.

Ниже представлен интерфейс сервиса, описывающего установленные требования. Первые два условия для сервиса объединены в один метод `GetYears`.

Файл `RefCasesServerExtension\Services\IRefCasesService.cs`

```
using RefCasesServerExtension.Models;
using System;
using System.Collections.Generic;

namespace RefCasesServerExtension.Services
{
    public interface IRefCasesService
    {
        List<Year> GetYears(Guid? rootSection = null);
        List<Section> GetSections(Guid yearID, Guid? rootSection = null);
        List<Case> GetCases(Guid sectionID);
        string GetCaseTitle(Guid caseID);
        List<CaseClientModel> SearchCases(string caseName, Guid? rootSection = null);
    }
}
```

И реализация сервиса `IRefCasesService`:

Файл `RefCasesServerExtension\Services\RefCasesService.cs`

```
using DocsVision.Platform.ObjectManager;
using DocsVision.Platform.WebClient;
using DocsVision.Platform.WebClient.Managers;
using RefCasesServerExtension.Models;
using System;
```

```
using System.Collections.Generic;
using System.Linq;

namespace RefCasesServerExtension.Services
{
    public class RefCasesService : IRefCasesService
    {

        private Guid refCasesId = new Guid("246197EA-846A-44DA-9EA3-0BCAE5500388"); ①
        private Guid yearsSectionId = new Guid("BD44C786-6E41-450E-BD60-66919657E51B");
②
        private Guid sectionsSectionID = new Guid("319E425D-543C-45DB-BD51-
955B58476EDB"); ③
        private Guid caseSectionID = new Guid("56af8231-b918-42d4-ac15-90ec2e9a0725"); ④

⑤
        private const string Year = "Year";
        private const string DisplayIndex = "DisplayIndex";
        private const string Case_Name = "Case_Name";
        private const string Case_Index = "Case_Index";
        private const string Case_SectionDisplayIndex = "Case_SectionDisplayIndex";
        private const string Name = "Name";

        ICurrentObjectContextProvider objectContextProvider;
        private AdvancedCardManager cardManager => objectContextProvider
.GetOrCreateCurrentSessionContext().AdvancedCardManager;

        public RefCasesService(ICurrentObjectContextProvider objectContextProvider) =>
this.objectContextProvider = objectContextProvider;

        public List<Year> GetYears(Guid? rootSection) ⑥
        {
            CardData refCasesData = cardManager.GetDictionaryData(refCasesId);
            SectionData yearSection = refCasesData.Sections[yearsSectionId];

            if (rootSection.HasValue && refCasesData.Sections[sectionsSectionID
].RowExists(rootSection.Value)) ⑦
            {
                var sectionRow = refCasesData.Sections[sectionsSectionID].GetRow
(rootSection.Value);
                var yearRow = sectionRow.SubSection.ParentRow;

                return new List<Year>() {
                    new Year() { ID = yearRow.Id, DisplayValue = yearRow[Year].ToString()
}
                };
            }
```

```
            return yearSection.Rows.Select<RowData, Year>(row => new Year() ⑧
            {
                ID = row.Id,
                DisplayValue = row[Year].ToString()
            }).ToList();
        }

        public List<Section> GetSections(Guid yearID, Guid? rootSection) ⑨
        {
            CardData refCasesData = cardManager.GetDictionaryData(refCasesId);
            SectionData sectionsSection = refCasesData.Sections[sectionsSectionID];

            if (rootSection.HasValue && sectionsSection.RowExists(rootSection.Value)) ⑩
            {
                RowData row = sectionsSection.GetRow(rootSection.Value);
                return GetSectionsFromRows(new List<RowData>() { row });
            }

            SectionData yearSection = refCasesData.Sections[yearsSectionId];

            if (yearSection.RowExists(yearID)) ⑪
            {
                RowDataCollection sectionRows = yearSection.GetRow(yearID).ChildSections
[sectionsSectionID].Rows;
                return GetSectionsFromRows(sectionRows);
            }

            return new List<Section>();
        }

        public List<Case> GetCases(Guid sectionID) ⑫
        {
            CardData refCasesData = cardManager.GetDictionaryData(refCasesId);
            SectionData sectionsSection = refCasesData.Sections[sectionsSectionID];

            if (sectionsSection.RowExists(sectionID))
            {
                RowDataCollection sectionRows = sectionsSection.GetRow(sectionID
).ChildSections[caseSectionID].Rows;
                return GetCasesFromRows(sectionRows);
            }

            return new List<Case>();
        }

        public string GetCaseTitle(Guid caseID) ⑬
        {
            CardData refCasesData = cardManager.GetDictionaryData(refCasesId);
```

```
            SectionData section = refCasesData.Sections[caseSectionID];

            if (section.RowExists(caseID))
            {
                RowData caseRow = section.GetRow(caseID); ⑭
                var sectionRow = caseRow.SubSection.ParentRow;
                var yearRow = sectionRow.SubSection.ParentRow;

                return string.Format("{0}, {1}, {2}", yearRow[Year], sectionRow
[DisplayIndex], caseRow[Case_Name]);
            }

            return "Ошибка!";
        }

        public List<CaseClientModel> SearchCases(string caseName, Guid? rootSection =
null) ⑮
        {
            CardData refCasesData = cardManager.GetDictionaryData(refCasesId);

            RowDataCollection allRows;

            if (rootSection.HasValue && refCasesData.Sections[sectionsSectionID
].RowExists(rootSection.Value)) ⑯
                allRows = refCasesData.Sections[caseSectionID].GetAllRows(rootSection
.Value, true);
            else
                allRows = refCasesData.Sections[caseSectionID].GetAllRows();

            var results = new List<CaseClientModel>();

            foreach (var caseRow in allRows)
            {
                if (caseRow[Case_Name].ToString().IndexOf(caseName, StringComparison
.InvariantCultureIgnoreCase) > -1 ||
                    (caseRow[Case_SectionDisplayIndex].ToString() + "-" + caseRow
[Case_Index].ToString()).IndexOf(caseName, StringComparison.InvariantCultureIgnoreCase) >
-1) ⑰
                {
                    var sectionRow = caseRow.SubSection.ParentRow;
                    var yearRow = sectionRow.SubSection.ParentRow;

                    results.Add(new CaseClientModel()
                    {
                        Id = caseRow.Id,

                        Name = string.Format("{0}, {1}, {2}", yearRow[Year], sectionRow
[DisplayIndex], caseRow[Case_Name]) ⑱
```

```
                    });
                }
            }

            return results;
        }


        List<Section> GetSectionsFromRows(IEnumerable<RowData> rows) ⑲
        {
            var nodes = new List<Section>();

            foreach (var row in rows)
            {
                var node = new Section() { ID = row.Id, DisplayValue = row[Name].
ToString() };
                if (row.HasChildRows)
                    node.Sections = GetSectionsFromRows(row.ChildRows);

                nodes.Add(node);
            }
            return nodes;
        }

        List<Case> GetCasesFromRows(RowDataCollection rows) ⑳
        {
            var nodes = new List<Case>();

            foreach (var row in rows)
            {
                var node = new Case() { ID = row.Id, DisplayValue = row[Case_Name
].ToString() };
                if (row.HasChildRows)
                    node.Cases = GetCasesFromRows(row.ChildRows);

                nodes.Add(node);
            }
            return nodes;
        }
    }
}
```

① Справочник номенклатуры дел 5.

② Секция *Года* справочника.

③ Секция *Разделы* справочника.

④ Секция *Дела* справочника.

⑤ Названия полей.

⑥ Возвращает список лет из *Справочника номенклатуры дел 5*.

⑦ Если выбран корневой раздел, нужно вернуть только его *Год*.

⑧ Иначе возвращаем все *Года*.

⑨ Возвращает дерево разделов указанного *Года* из *Справочника номенклатуры дел 5*.

⑩ Если выбран корневой раздел, вернуть только его подразделы.

⑪ Иначе вернуть все разделы указанного *Года*.

⑫ Вернуть все дела указанного *Раздела*.

⑬ Вернуть отображаемое название *Дела*.

⑭ Получаем для дела родительские строки раздела и *Года*.

⑮ Поиск дела по названию и по идентификатору *Дела*.

⑯ Если указан коневой *Раздел*, поиск только в нём, иначе — во всех *Разделах*.

⑰ Проверяем название *Раздела* — поле `Case_Name`.

⑱ Возвращаем сразу отображаемое название.

⑲ Возвращаем список *Разделов* для строк секции справочника.

⑳ Возвращаем список *Дел* для строк справочника.

Конвертер "RefCasesConverter"

Как уже было сказано ранее, конвертер нужен для формирования отображаемого значения элемента управления, показываемого при инициализации элемента. Иначе в элементе будет показан идентификатор выбранного дела.

Для разрабатываемого элемента управления название конвертера (`RefCasesConverter`) было объявлено в описателе элемента управления. Теперь его нужно реализовать и зарегистрировать в серверном расширении Web-клиента.

В данном пример конвертер использует метод `IRefCasesService.GetCaseTitle` для получения отображаемого названия Дела.

Файл `RefCasesServerExtension\BindingConverters\RefCasesConverter.cs`

```
using DocsVision.WebClientLibrary.Layout.IL;
```

```
using DocsVision.WebClientLibrary.ObjectModel;
using DocsVision.WebClientLibrary.ObjectModel.Services.BindingConverters;
using DocsVision.WebClientLibrary.ObjectModel.Services.LayoutModel;
using RefCasesServerExtension.Models;
using RefCasesServerExtension.Services;
using System;

namespace RefCasesServerExtension.BindingConverters
{
    public class RefCasesConverter : BaseBindingConverter ①
    {
        private IRefCasesService refCasesService;

        public RefCasesConverter(IServiceProvider serviceProvider, IRefCasesService
refCasesService) : base(serviceProvider, "RefCasesConverter") ②
        {
            this.refCasesService = refCasesService; ③
        }

        public override BindingResult ConvertForDisplay(ControlContext controlContext,
LayoutBinding binding, BindingResult bindingResult) ④
        {
            var itemId = bindingResult.Value != null ? (Guid)bindingResult.Value : Guid
.Empty;
            var name = itemId == Guid.Empty ? "" : refCasesService.GetCaseTitle(itemId);

            var model = new CaseClientModel() { Id = itemId, Name = name }; ⑤

            return bindingResult.Clone(model);
        }
    }
}
```

① Класс конвертера может быть производным от `BaseBindingConverter` или полностью реализовывать интерфейс `IBindingConverter`.

② В базовый класс нужно передать название конвертера, указанного в описателе `RefCasesConverter`.

③ Получаем реализованный ранее сервис для работы со *Справочником номенклатуры дел 5*.

④ Основной метод, возвращающий отображаемое значение (а точнее модель) для значения элемента управления — `bindingResult.Value`.

⑤ Клиент ожидает модель, включающую идентификатор и название *Дела*.

Веб-методы серверного расширения

Клиентской части элемента управления нужны данные из *Справочника номенклатуры дел 5*: для показа списка лет, разделов, а также дел. Предоставить доступ к данным можно с помощью веб-методов серверного расширения.

Далее приведён код контроллера, предоставляющего такие веб-методы.

Файл RefCasesServerExtension\Controllers\RefCasesOperationController.cs

```
using RefCasesServerExtension.Models;
using RefCasesServerExtension.Services;
using System;
using System.Linq;
using DocsVision.Platform.WebClient.Models;
using DocsVision.Platform.WebClient.Models.Generic;
using Microsoft.AspNetCore.Mvc;

namespace RefCasesServerExtension.Controllers
{
    public class RefCasesOperationController : ControllerBase
    {
        private readonly IRefCasesService refCasesService;

        public RefCasesOperationController(IRefCasesService refCasesService) ①
        {
            this.refCasesService = refCasesService;
        }

        [HttpPost]
        public Year[] GetYears([FromQuery]Guid? rootSectionID) ②
        {
            return refCasesService.GetYears(rootSectionID).ToArray();
        }

        [HttpPost]
        public Section[] GetSections([FromQuery]Guid yearID, [FromQuery]Guid?
rootSectionID) { ③
            return refCasesService.GetSections(yearID, rootSectionID).ToArray();
        }

        [HttpPost]
        public Case[] GetCases([FromQuery]Guid sectionID) ④
        {
            return refCasesService.GetCases(sectionID).ToArray();
        }
```

```
        [HttpPost]
        public string GetCaseDisplayName([FromQuery]Guid caseID) ⑤
        {
            return refCasesService.GetCaseTitle(caseID);
        }

        [HttpPost]
        public CaseSearchResult SearchCase([FromQuery]string caseName, [FromQuery]int
skipCount, [FromQuery]int maxCount, [FromQuery]Guid? rootSectionID) ⑥
        {
            var rows = refCasesService.SearchCases(caseName, rootSectionID); ⑦

            var result = new CaseSearchResult
            {
                Items = rows.Skip(skipCount).Take(maxCount).ToArray(), ⑧ ⑨

                HasMore = rows.Count > skipCount + maxCount
            }; ⑩

            return result;
        }
    }
}
```

① В конструкторе контроллера получаем ссылку на реализованный сервис для работы со Справочником номенклатуры дел 5.

② Возвращает список лет.

③ Возвращает список разделов.

④ Возвращает список дел.

⑤ Возвращает отображаемое название дела.

⑥ Поиск дела по имени.

⑦ Получаем все подходящие дела.

⑧ Оставляем только количество запрошенных клиентом.

⑨ Т. к. количество результатов поискового запроса `SearchCase` может быть достаточно большим, клиенту предоставлена возможность ограничивать его с помощью параметров `skipCount` (количество пропускаемых результатов) и `maxCount` (максимальное количество результатов, принимаемых клиентом за один раз).

⑩ Устанавливаем флаг наличия дополнительных результатов.

Ядро серверного расширения

Реализованные сущности необходимо зарегистрировать в серверном расширении.

Файл RefCasesServerExtension\RefCasesServerExtension.cs

```
using System;
using System.Collections.Generic;
using System.Diagnostics;
using System.Reflection;
using System.Resources;
using Autofac;
using Autofac.Extras.Ordering;
using DocsVision.WebClient.Extensibility;
using DocsVision.WebClient.Helpers;
using DocsVision.WebClientLibrary.ObjectModel.Services.BindingConverters;
using Microsoft.Extensions.DependencyInjection;
using RefCasesServerExtension.BindingConverters;
using RefCasesServerExtension.Services;

namespace RefCasesServerExtension
{
    public class RefCasesServerExtension : WebClientExtension
    {
        public RefCasesServerExtension(IServiceProvider serviceProvider)
            : base()
        {
        }

        public override string ExtensionName
        {
            get { return Assembly.GetAssembly(typeof(RefCasesServerExtension)).GetName
().Name; }
        }

        public override Version ExtensionVersion
        {
            get { return new Version(FileVersionInfo.GetVersionInfo(Assembly
.GetExecutingAssembly().Location).FileVersion); }
        }

        #region WebClientExtension Overrides

        public override void InitializeServiceCollection(IServiceCollection services) ①
        {
            services.AddSingleton<IBindingConverter, RefCasesConverter>(); ②
            services.AddSingleton<IRefCasesService, RefCasesService>(); ③
```

```
        }

        protected override List<ResourceManager> GetLayoutExtensionResourceManagers() ④
        {
            return new List<ResourceManager>
            {
                { Resources.ResourceManager}
            };
        }

        #endregion
    }
}
```

① Регистрация компонентов, реализованных в расширении.

② Регистрируем конвертер.

③ Регистрируем сервисы

④ Также для примера добавлено несколько локализованных текстовых ресурсов, которые будут использоваться в клиентском расширении.



Контроллеры регистрировать не требуется.

Создание клиентского расширения с реализацией элемента управления

Непосредственно элемент управления для Web-клиента реализуется в клиентском расширении, и состоит из нескольких частей (см. пункт Разработка и публикация клиентского компонента элемента управления): класс параметров, интерфейс состояния, интерфейсный класс и класс реализации.

С точки зрения пользовательского интерфейса реализуемый элемент управления будет представлять собой поле для ввода текста, в котором по умолчанию будет отображаться значение из поля карточки, связанного со строкой *Дела Справочника номенклатуры дел*. Особенностью элемента управления является необходимость реализации дополнительного графического компонента, в котором будут отображаться данные для выбора значения из *Справочника номенклатуры дел 5*, а также возможность быстрого поиска дел по справочнику.

Клиентский сервис для работы со "Справочником номенклатуры дел"

Реализуемому элементу управления потребуется доступ к *Справочнику номенклатуры дел*, который предоставляют веб-методы, разработанные в серверном расширении. Взаимодействовать с веб-методами можно вызывая их

напрямую в основном коде элемента или используя отдельный клиентский сервис.

Ниже представлен код такого сервиса, который будет использоваться далее.

Файл RefCasesWebExtension\src\Services\RefCasesServices.ts

```
import { $RequestManager } from "@docsvision/webclient/System/$RequestManager";
import { urlStore } from "@docsvision/webclient/System/UrlStore";
import { Models } from "../Controls/RefCases/Data/CaseModel";
import { serviceName } from "@docsvision/webclient/System/ServiceUtils";

export class RefCasesService {
    constructor(private services: $RequestManager) {
    }

    getYears(rootSectionID?: string): Promise<Models.RefCasesYearModel[]> { ①
        let url = urlStore.urlResolver.resolveApiUrl("GetYears", "RefCasesOperation");
        url = url + "?rootSectionID=" + rootSectionID;

        return this.services.requestManager.post<Models.RefCasesYearModel[]>(url, "");
    }

    getSections(yearID?: string, rootSectionID?: string): Promise<Models
.RefCasesSectionModel[]> { ②
        let url = urlStore.urlResolver.resolveApiUrl("GetSections", "RefCasesOperation");
        url = url + "?yearID=" + yearID + "&rootSectionID=" + rootSectionID;

        return this.services.requestManager.post<Models.RefCasesSectionModel[]>(url, "");
    }

    getCases(sectionID?: string): Promise<Models.RefCasesCaseModel[]> { ③
        let url = urlStore.urlResolver.resolveApiUrl("GetCases", "RefCasesOperation");
        url = url + "?sectionID=" + sectionID;

        return this.services.requestManager.post<Models.RefCasesCaseModel[]>(url, "");
    }

    getCaseTitleName(caseID?: string): Promise<string> { ④
        let url = urlStore.urlResolver.resolveApiUrl("GetCaseDisplayName",
"RefCasesOperation");
        url = url + "?caseID=" + caseID;

        return this.services.requestManager.post<string>(url, "");
    }

    searchCase(caseName: string, skipCount: number, maxCount: number, rootSectionID?:
```

```
string): Promise<Models.CaseSearchResult> { ⑤
        let url = urlStore.urlResolver.resolveApiUrl("SearchCase", "RefCasesOperation");
        url = url + "?caseName=" + caseName + "&skipCount=" + skipCount + "&maxCount=" +
maxCount + "&rootSectionID=" + rootSectionID;

        return this.services.requestManager.post<Models.CaseSearchResult>(url, "");
    }
}

export type $RefCasesService = { refCasesService: RefCasesService };
export const $RefCasesService = serviceName((s: $RefCasesService) => s.refCasesService);
```

① Возвращает модель списка лет.

② Возвращает модель списка (дерева) разделов.

③ Возвращает модель списка дел.

④ Возвращает отображаемое название дела.

⑤ Возвращает результат поиска дела по названию. В `skipCount` и `maxCount` будут передаваться ограничения для количества результатов, возвращаемых сервером.

Класс параметров

В классе параметров объявляются свойства и события элемента управления, указанные в его описателе.

Данный элемент управления представляет собой поле для ввода текста, основанный на элементе `InputBasedControl`, в котором уже объявлены стандартные свойства подобного элемента. Единственным новым свойством является свойство `RootSection`, содержащее идентификатор корневого раздела справочника.

Ниже приведён код класса параметров, наследующий основной набор свойств от `InputBasedControlParams` — класса параметров элемента `InputBasedControl`. Значение элемента управления описывается моделью `Models.RefCasesCaseDisplayModel`.

Файл `RefCasesWebExtension\src\Controls\RefCases\RefCases.tsx`

```
export class RefCasesParams extends InputBasedControlParams<Models
.RefCasesCaseDisplayModel> {

    @r standardCssClass?: string = "ref-cases"; ①

    @rw rootSection?: string; ②
```

```
    @rw services?: $LayoutDirectoryDesignerController & $EditOperationStore & $LayoutInfo
& $RefCasesService; ③
}
```

① Стандартный CSS класс со стилями элемента управления Необходимо переопределить с собственным названием CSS класса.

② Секция справочника, из которой разрешено выбирать дела.

Настраивается в программе Конструктор Web-разметок. Может быть пустым (ограничение отсутствует).

③ Клиентские сервисы. `$RefCasesService` — собственный сервис для работы с данными *Справочника номенклатуры дел*.

Интерфейс состояния

В интерфейсе состояния нужно объявить свойства, описывающие внутреннее состояние элемента управления. В частности во внутреннем состоянии будет храниться "биндинг" элемента управления.

Файл `RefCasesWebExtension\src\Controls\RefCases\RefCasesImpl.tsx`

```
①
export interface RefCasesState extends RefCasesParams, InputBasedControlState<Models
.RefCasesCaseDisplayModel> {
    binding: IBindingResult<Models.RefCasesCaseDisplayModel>;  ②

    dialog: ModalWindow; ③

    requestHelper: RequestHelper; ④

    inputKeyDown: SimpleEvent<React.KeyboardEvent<any>>; ⑤
}
```

① @internal.

② Используется для хранения биндинга.

③ Диалоговое окно выбора *Дела* из справочника.

④ Вспомогательный компонент, предоставляющий метод для обработки длительных операций (будет использоваться при получении данных из справочника).

⑤ Событие ввода значения в поле элемента управления.

Интерфейсный класс

В интерфейсном классе необходимо создать экземпляр реализации элемента управления, загрузить "биндинг", а также реализовать метод getBindings, возвращающий "биндинг" с текущим значением.

Файл RefCasesWebExtension\src\Controls\RefCases\RefCases.tsx

```
export class RefCases extends InputBasedControl<Models.RefCasesCaseDisplayModel,
RefCasesParams, RefCasesState>
{
    protected getServices(): $LayoutInfo {
        return this.state.services;
    }

    protected createParams(): RefCasesParams { ①
        return new RefCasesParams();
    }

    @handler("binding") ②
    protected set binding(binding: IBindingResult<Models.RefCasesCaseDisplayModel>) {
        this.value = binding && binding.value;
        this.state.canEdit = editOperationAvailable(this.state.services, binding);
        this.state.binding = binding;
    }

    protected getBindings() { ③
        let binding = cloneObject(this.state.binding);
        return [getBindingResult(binding, this.params.value && this.params.value.id ||
null, () => at(RefCasesParams).labelText)];
    }

    protected createImpl() { ④
        return new RefCasesImpl(this.props, this.state);
    }
}
```

① Инициализация параметров элемента управления.

② Загружаем биндинг при инициализации элемента управления.

③ Возвращаем биндинги.

④ Предоставляем экземпляр реализации элемента управления Справочник номенклатуры дел.

Класс реализации

В данном примере элемент управления представляет собой поле для ввода текста, дополненное функцией быстрого поиска, реализуемой с использованием компонента Typeahead.

Реализация класса в данном примере является достаточно большой: здесь код будет приведён частично.

Файл RefCasesWebExtension\src\Controls\RefCases\RefCasesImpl.tsx

```
①
export class RefCasesImpl extends InputBasedControlImpl<Models.RefCasesCaseDisplayModel,
RefCasesParams, RefCasesState>
{
    private typeahead: Typeahead;

    constructor(props: RefCasesParams, state: RefCasesState) {
        super(props, state);

        this.state.requestHelper = new RequestHelper(() => this.forceUpdate()); ②

        this.state.inputKeyDown = new SimpleEvent<React.KeyboardEvent<any>>(this);

        this.findItems = this.findItems.bind(this);
        this.showDictionary = this.showDictionary.bind(this);
        this.onSelected = this.onSelected.bind(this);
        this.attachTypeahead = this.attachTypeahead.bind(this); ③
    }

    async showDictionary() { ④
        if (this.state.dialog && this.state.dialog.IsOpened) {
            return;
        }

        let controlInModal: RefCasesSelectDialog; ⑤

        let params = new ModalWindowParams(); ⑥
        params.headerText = resources.RefCases_SelectFromDirectory;
        params.content = "";
        params.buttonOkShow = true;
        params.buttonOkText = resources.Navigator_ButtonSelect;

        let okFunction = () => { ⑦
            let selectedCase = cloneObject(controlInModal.selectedCase);

            if (selectedCase) { ⑧
```

```
                this.state.services.refCasesService.getCaseTitleName(selectedCase
.uniqueId).then((title) => {

                    let displayValue = {
                        id: selectedCase.uniqueId,
                        name: title
                    } as Models.RefCasesCaseDisplayModel;

                    this.setValue(displayValue, true);
                });

                if (this.state.dialog) {
                    this.state.dialog.Hide();
                    this.state.dialog = null;
                }
            }
        };

        params.buttonOkFunction = okFunction; ⑨

        const value = this.getValue(); ⑩
        this.state.dialog = new ModalWindow(params);

        renderModalContent(this.state.dialog, ( ⑪
            <RefCasesSelectDialog key={this.state.name + "_Modal"} ref={(el) =>
controlInModal = el}
                rootSectionId={this.state.rootSection}
                services={this.state.services}

                nodeSelected={(node) => { ⑫
                    if (this.state.dialog) {
                        if (node) this.state.dialog.OkButtonElement.classList.remove(
"disabled");
                        else this.state.dialog.OkButtonElement.classList.add("disabled");
                    }
                }}

                nodeAccepted={okFunction} /> ⑬
        ));

        this.state.dialog.Show();
        this.state.dialog.OkButtonElement.classList.add("disabled");
    }


    protected renderInputWithPlaceholder(): React.ReactNode { ⑭

        let buttons: IBoxWithButtonsButtonInfo[] = [ ⑮
```

```
            {
                onClick: this.showDictionary,
                name: "open-dictionary",
                iconClassName: "open-dictionary-button-icon dv-ico dv-ico-dictionary",
                visible: this.editAvailable,
                title: resources.RefCases_SelectFromDirectory,
                disabled: !this.editAvailable,
                tabIndex: this.getTabIndex(),
            }
        ];

        return ( ⑯
            <Typeahead className={"universal-directory-box"} extraButtons={buttons}
                findItems={this.findItems}
                clearButton={this.hasValue()}
                searchText={this.state.inputText}
                afterOpenCallback={() => this.afterOpenCallback()}
                popoverClassName={this.state.standardCssClass}
                popoverAttributes={{ "data-control-name": this.state.name }}
                inputKeyDown={this.state.inputKeyDown}
                onSelected={this.onSelected}
                disabled={!this.editAvailable}
                ref={this.attachTypeahead}>
                {super.renderInputWithPlaceholder()}
            </Typeahead>
        );
    }



    protected findItems(typeaheadQuery: ITypeaheadSearchQuery): Promise
<ITypeaheadSearchResult> { ⑰
        return new Promise<ITypeaheadSearchResult>((resolve, reject) => {
            this.state.services.refCasesService.searchCase(typeaheadQuery.searchText,
typeaheadQuery.skipCount, typeaheadQuery.maxCount, this.state.rootSection).then(response
=> {
                let result = {
                    items: response.items.map(item => new CaseTypeaheadVariant(item)), ⑱

                    hasMore: response.hasMore ⑲
                } as ITypeaheadSearchResult;
                resolve(result);
            }).catch(reject);
        });
    }
}
```

① @internal.

② Инициализация компонента для выполнения длительных операций.

③ Связывание обработчиков с контекстом.

④ Отрисовка основного элемента управления: поле для ввода текста, к которому добавляется стандартная кнопка выбора значения из справочника.

⑤ Кнопка открытия справочника. Отключается, если нет прав на операцию редактирования.

⑥ Формируем элемент с быстрым поиском.

⑦ Метод, отображающий диалоговое окно выбора *Дела* из справочника (реализовано в отдельном компоненте).

⑧ Компонент диалогового окна выбора из справочника.

⑨ Устанавливаем параметры диалогового окна.

⑩ Обработчик нажатия кнопки **ОК** в диалоговом окне.

⑪ Если выбрано *Дело*, его модель (`RefCasesCaseDisplayModel`) устанавливается в значение элемента управления.

⑫ Устанавливается обработчик нажатия кнопки **ОК**.

⑬ Получаем текущее значение элемента управления.

⑭ Формируем диалоговое окно.

⑮ Кнопка **ОК** включается при выборе *Дела*.

⑯ Обработчик для двойного щелчка по делу — аналогично нажатию кнопки **ОК**.

⑰ Обработчик события быстрого поиска по справочнику. Результаты поиска отображаются в списке, выводимом под окном поля ввода.

⑱ Результаты поиска, должны быть приведены к `ITypeaheadSearchResult`.

⑲ Флаг, сообщающий о наличии результатов, не включенных в данный ответ.

В данном компоненте три ключевых функции:

- `renderInputWithPlaceholder` — формирует поле для ввода текста с возможностью быстрого поиска, которая реализована в компоненте `Typeahead`.
- `showDictionary` — открывает диалоговое окно с элементами для выбора дела из справочника.
- `findItems` — реализует функцию получения результатов поиска дел по справочнику.

Компонент `Typeahead` умеет ограничивать кол-во результатов, получаемых за один раз. Данные ограничения регулируются с помощью его параметров:

- `firstPageSize` — максимальное кол-во результатов при первом запросе, по умолчанию — 8.
- `nextPageSize` — максимальное кол-во результатов при следующих запросах, по умолчанию — 15.

При выполнении первого запроса в функцию `findItems` передаётся аргумент `ITypeaheadSearchQuery` со значениями `skipCount = 0` и `maxCount = firstPageSize`. При следующем запросе — при нажатии пользователем кнопки **Показать ещё**, в `findItems` передаются значения `skipCount = firstPageSize` и `maxCount = nextPageSize`. Таким образом клиент может запросить у сервера недостающие данные, если после первого запроса сервер вернул флаг `hasMore = true`.

Компонент диалогового окна выбора дела из справочника

Диалоговое окно выбора дела из *Справочника номенклатуры дел* представляет собой панель с тремя элементами управления:

- Раскрывающийся список для выбора года,
- Дерево разделов,
- Дерево дел.

> В модуле *Делопроизводство 5* секция дел является древовидной, функция вложенных дел не используется. Тем не менее в данном примере для совместимости используется элемент, поддерживающий древовидную структуру данных.
>
> С целью уменьшения количества исходного кода в данном примере умышленно не используется функция выбора текущего значения элемента управления в окне выбора Дела. Данная возможность может быть реализована самостоятельно.

В приведённом далее коде пропущены участки с классом параметров и интерфейсом состояния диалогового окна. Полный исходный код доступен на GitHub.

Файл RefCasesWebExtension\src\Controls\RefCases\RefCasesSelectDialog\RefCasesSelectDialog.tsx

```
export class RefCasesSelectDialog extends React.Component<IRefCasesSelectDialogProps,
IRefCasesSelectDialogState> {

①
    state: IRefCasesSelectDialogState;

    constructor(props: IRefCasesSelectDialogProps) {
        super(props);

        this.state = {} as IRefCasesSelectDialogState;
        this.state.requestHelper = new RequestHelper(() => this.forceUpdate());


        this.collectYearsList = this.collectYearsList.bind(this);
        this.loadSectionsTree = this.loadSectionsTree.bind(this);
        this.loadCasesTree = this.loadCasesTree.bind(this);

        this.onSectionNodeSelected = this.onSectionNodeSelected.bind(this);
        this.onCaseNodeSelected = this.onCaseNodeSelected.bind(this);
        this.onCaseNodeAccepted = this.onCaseNodeAccepted.bind(this); ②

        this.collectYearsList(); ③
    }

    public get selectedCase() {
        return this.state.selectedCaseNode;
    }

    protected collectYearsList() { ④
        this.props.services.refCasesService.getYears(this.props.rootSectionId).then
((items) => { ⑤
            this.state.years = items.map(x => ({
                id: x.id,
                title: x.displayValue
            } as IComboBoxElement));

            this.setState({ showYearsList: true }); ⑥
        });
    }

    protected loadSectionsTree(): Promise<IDynamicTreeNodeData[]> { ⑦
        return new Promise<IDynamicTreeNodeData[]>((resolve, reject) => {
            this.state.requestHelper.send(
                () => this.props.services.refCasesService.getSections(this.state
.selectedYearID, this.props.rootSectionId),
                items => {
```

```
                    let nodes = RefCasesSectionTreeNode.Create(items);
                    resolve(nodes);
                },
                reject);
        });
    }

    protected loadCasesTree(): Promise<IDynamicTreeNodeData[]> { ⑧
        return new Promise<IDynamicTreeNodeData[]>((resolve, reject) => {
            this.state.requestHelper.send(
                () => this.props.services.refCasesService.getCases(this.state
.selectedSectionID),
                items => {
                    let nodes = RefCasesCaseTreeNode.Create(items);
                    resolve(nodes);
                },
                reject);
        });
    }

    protected onSectionNodeSelected(node: TreeNode) { ⑨
        this.state.selectedSectionID = node.uniqueId;

        this.state.selectedCaseNode = null;
        this.props.nodeSelected && this.props.nodeSelected(null);

        this.setState({ showCasesTree: false }, () => this.setState({ showCasesTree: true
}));
    }

    protected onCaseNodeSelected(node: TreeNode) { ⑩
        this.state.selectedCaseNode = node.data as RefCasesCaseTreeNode;
        this.props.nodeSelected && this.props.nodeSelected(node.data as
RefCasesCaseTreeNode);
    }

    protected onCaseNodeAccepted(node: TreeNode) { ⑪
        this.state.selectedCaseNode = node.data as RefCasesCaseTreeNode;
        this.props.nodeAccepted && this.props.nodeAccepted(node.data as
RefCasesCaseTreeNode);
    }

    render(): React.ReactNode { ⑫

        let yearsList = <div>{resources.RefCases_Years} ⑬
            <CommonComboBox elements={this.state.years} selectedID={this.state
.selectedYearID}
            onChange={(selectedElement: IComboBoxElement) => { ⑭
```

```
                this.setState({ selectedYearID: selectedElement.id });
                this.setState({ showSectionsTree: false }, () => this.setState({
showSectionsTree: true })); ⑮
                this.setState({ showCasesTree: false });
                this.forceUpdate();
            }}

            renderElementList={(elements, expanded) =>
                <PopoverComboBoxBodyContent mode={PopoverMode.BottomDropdown} isOpen=
{expanded} className="combobox-helper">
                    {elements}
                </PopoverComboBoxBodyContent>
            } />
            </div>;

        let sectionsTree = <div className="ref-cases-dialog__tree"> ⑯
            <div className="tree-block">{resources.RefCases_Sections}
                <DynamicTree loadNodes={this.loadSectionsTree} treeHeight={300}
                nodeSelected={this.onSectionNodeSelected} > ⑰
                </DynamicTree>
            </div>
        </div>;

        let casesTree = <div className="ref-cases-dialog__tree"> ⑱
            <div className="tree-block">{resources.RefCases_Cases}
                <DynamicTree loadNodes={this.loadCasesTree} treeHeight={300}
                nodeSelected={this.onCaseNodeSelected} nodeAccepted={this
.onCaseNodeAccepted} > ⑲
                </DynamicTree>
            </div>
        </div>;

        return (
            <div>
                {this.state.showYearsList && yearsList}
                {this.state.showSectionsTree && sectionsTree}
                {this.state.showCasesTree && casesTree}
            </div>
        );
    }
}
```

① @internal

② Связываем функции и обработчиков событий с контекстом.

③ Загружаем список лет из справочника.

④ Загружаем из *Справочника номенклатуры дел* список лет в `state.years`, который является источником данных для элемента управления `CommonComboBox`.

⑤ Если установлен раздел, из которого возможен выбор *Дел* — `rootSectionId`, будет возвращен только год с данным разделом.

⑥ Показываем элемент со списком лет.

⑦ Возвращает список разделов из Справочника номенклатуры дел.

⑧ Загружает список дел после выбора раздела.

⑨ Сохраняет *Дело* в `selectedNode` после его выбора в списке дел.

⑩ Сохраняет *Дело* в `selectedNode` после его выбора в списке дел и нажатия кнопки **Выбрать**.

⑪ Сохраняет *Дело* в `selectedNode` после его выбора в списке дел.

⑫ Инициализация интерфейса.

⑬ Список лет.

⑭ При выборе года из списка инициализируем дерево разделов.

⑮ Элемент для выбора *Раздела* перемонтируется, элемент для выбора *Дела* отмонтируется.

⑯ Дерево *Разделов*.

⑰ При выборе *Раздела* из дерева разделов вызываем метод, загружающий дерево *Дел*.

⑱ Список дел.

⑲ При выборе *Дела*, сохраняем значение узла, а при двойном щелчке вызываем обрабочик `onCaseNodeAccepted`.

Пример стиля для дерева дел и дерева разделов

Файл `RefCasesWebExtension/src/Controls/RefCases/RefCasesSelectDialog/RefCasesSelectDialog.scss`

```
.ref-cases-dialog__tree {
    border-bottom: 1px solid lightgray;
    padding-top: 10px;
    display: inline-block;
    width: 50%;
}
```

Дополнительно

Руководство по разработке решений Web-клиента

> **Дополнительно**
>
> Дополнительная информация по разработке решений Web-клиента.
>
> - [WebClient:standartControlsPropertiesAndEvents]
> - [WebClient:non-standard-property-editors]
> - [WebClient:standartStyles]
> - Больше информации в навигационном меню слева…

Стандартные свойства и события элементов управления

Далее приведено описание стандартных свойств и событий элементов управления, которое включает: тип значения свойства, метод получения описателя свойства, а также ключ ресурса с локализованным названием — для событий.

Свойства

Name / Название

Тип: `String`

Метод получения описателя: `PropertyFactory.GetNameProperty`

Width / Ширина в процентах

Тип: `Double`

Метод получения описателя: `PropertyFactory.GetWidthProperty`

Header / Заголовок

Тип: `String`

Метод получения описателя: `PropertyFactory.GetHeaderProperty`

Binding / Свойство, содержащее механизм получения данных

Тип: `String`

Метод получения описателя свойства: PropertyFactory.GetBindingProperty

DataSource / Источник данных

Тип: Guid

Метод получения описателя: PropertyFactory.GetDataSourceProperty

EditOperation / Операция редактирования

Тип: Guid

Метод получения описателя: PropertyFactory.GetEditOperationProperty

ExtendedDataSource / Расширенный источник данных

Тип: Guid

Метод получения описателя: PropertyFactory.GetExtendedDataSourceProperty

StandardCssClass / Стандартный css класс

Тип: String

Метод получения описателя: PropertyFactory.GetStandardCssClassProperty

CustomCssClasses / Дополнительные css классы

Тип: String

Метод получения описателя: PropertyFactory.GetCustomCssClassesProperty

TabStop / Переходить по TAB

Тип: Boolean

Метод получения описателя: PropertyFactory.GetTabStopProperty

Visibility / Видимость

Тип: Boolean

Метод получения описателя: PropertyFactory.GetVisibilityProperty

Default / Значение по умолчанию

Тип: String

Метод получения описателя: PropertyFactory.GetDefaultProperty

Required / Обязательное

Тип: Boolean

Метод получения описателя: PropertyFactory.GetRequiredProperty

EditMode / Режим редактирования

Тип: Int32

Метод получения описателя: PropertyFactory.GetEditModeProperty

PlaceHolder / Заполнитель

Тип: String

Метод получения описателя: PropertyFactory.GetPlaceHolderProperty

Tip / Подсказка

Тип: String

Метод получения описателя: PropertyFactory.GetTipProperty

LabelText / Текст метки

Тип: String

Метод получения описателя: PropertyFactory.GetLabelTextProperty

ShowEmptyLabel / Отображать метку без значения

Тип значения: Boolean

Метод получения описателя: PropertyFactory.GetShowEmptyLabelProperty

MinWidth / Минимальная ширина

Тип: Double

Метод получения описателя: PropertyFactory.GetMinWidthProperty

ButtonText / Текст кнопки

Тип: String

Метод получения описателя: PropertyFactory.GetButtonTextProperty

События

Все события имеют тип String.

Click / При щелчке

Метод получения описателя: `PropertyFactory.GetClickEvent`

Ключ ресурса: `ControlTypes_ClickEventProperty`

MouseOver / При наведении курсора

Метод получения описателя: `PropertyFactory.GetMouseOverEvent`

Ключ ресурса: `ControlTypes_MouseOverEventProperty`

MouseOut / При отведении курсора

Метод получения описателя: `PropertyFactory.GetMouseOutEvent`

Ключ ресурса: `ControlTypes_MouseOutEventProperty`

Focus / При получении фокуса

Метод получения описателя: `PropertyFactory.GetFocusEvent`

Ключ ресурса: `ControlTypes_FocusEventProperty`

Blur / При потере фокуса

Метод получения описателя: `PropertyFactory.GetBlurEvent`

Ключ ресурса: `ControlTypes_BlurEventProperty`

Collapsed / При сворачивании

Метод получения описателя: `PropertyFactory.GetCollapsedEvent`

Ключ ресурса: `ControlTypes_CollapsedEventProperty`

Collapsing / Перед сворачиванием

Метод получения описателя свойства: PropertyFactory.GetCollapsingEvent`

Ключ ресурса: `ControlTypes_CollapsingEventProperty`

Expanded / При разворачивании

Метод получения описателя: `PropertyFactory.GetExpandedEvent`

Ключ ресурса: `ControlTypes_ExpandedEventProperty`

Expanding / Перед разворачиванием

Метод получения описателя: `PropertyFactory.GetExpandingEvent`

Ключ ресурса: `ControlTypes_ExpandingEventProperty`

DataChanged / При смене данных

Метод получения описателя: `PropertyFactory.GetDataChangedEvent`

Ключ ресурса: `ControlTypes_DataChangedEventProperty`

Нестандартные встроенные редакторы свойств

1. Редактор локализуемых текстовых свойств.



Рисунок 714. Редактор локализуемых текстовых свойств

Пример использования в коде:

```
private PropertyDescription GetLocalizableMessagePropertyDescription () {
    var propertyDescription = new PropertyDescription {
        Type = typeof (string),
        Name = "LocalizableMessage",
        Category = DocsVision.Platform.Tools.LayoutEditor.Infrostructure.
PropertyCategoryConstants.DataCategory,
        DisplayName = "Localizable message",
        Localized = true,
        Editor = typeof (DocsVision.Platform.Tools.LayoutEditor.Editors
.LocalizedPropertyEditor)
    };
    return propertyDescription;
}
```

Для использования редактора нужно:

a. Установить свойство `Type` в `string`.

b. Установить свойство `Localized` в `true`.

c. Установить свойство `Editor` в `Docsvision.Platform.Tools.LayoutEditor.Editors.LocalizedPropertyEditor`.

2. Редактор перечислений.

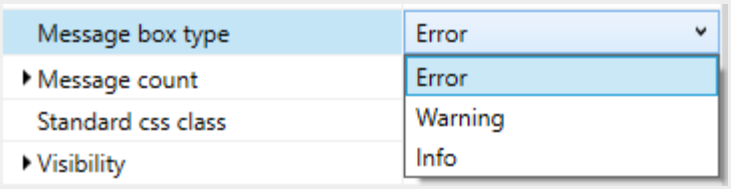


Рисунок 715. Редактор перечислений

Пример использования в коде:

```
enum MessageBoxType {
    Error = 0,
    Warning = 1,
    Info = 2
}

class MessageBoxTypeSource : Xceed.Wpf.Toolkit.PropertyGrid.Attributes
.IItemsSource { ①
    private readonly System.IServiceProvider serviceProvider;

    public MessageBoxTypeSource (System.IServiceProvider serviceProvider) {
        this.serviceProvider = serviceProvider;
    }

    public Xceed.Wpf.Toolkit.PropertyGrid.Attributes.ItemCollection GetValues ()
{ ②
        var itemCollection = new Xceed.Wpf.Toolkit.PropertyGrid.Attributes
.ItemCollection ();

        itemCollection.Add ((int) MessageBoxType.Error, MessageBoxType.Error
.ToString ());
        itemCollection.Add ((int) MessageBoxType.Warning, MessageBoxType.Warning
.ToString ());
        itemCollection.Add ((int) MessageBoxType.Info, MessageBoxType.Info
.ToString ());

        return itemCollection;
    }
}

class MessageBoxTypeEditor : Xceed.Wpf.Toolkit.PropertyGrid.Editors.ITypeEditor {
③
    public FrameworkElement ResolveEditor (PropertyItem propertyItem) {

        var editor = new ItemsSourceEditor (typeof (MessageBoxTypeSource)); ④
        return editor.ResolveEditor (propertyItem);
    }
```

```
}

var propertyDescription = new PropertyDescription { ⑤
    Type = typeof (int),
    Name = "MessageBoxType",
    Category = PropertyCategoryConstants.BehaviorCategory,
    DisplayName = Resources.Property_MessageBoxType_DisplayName,
    DefaultValue = 0,
    Editor = typeof (MessageBoxTypeEditor) ⑥
};
```

① Класс, содержащий данные списка. Должен быть унаследован от IItemsSource.

② Метод `GetValues` должен вернуть значения. Тип значений `ItemCollection`.

③ Класс редактора значения свойства должен содержать значения.

④ Редактор значения свойства создаётся из класса `ItemsSourceEditor`, в который передается класс с данными списка.

⑤ Свойство для выбора значения из списка.

⑥ Разработанный редактор значения свойства.

3. Редактор устройствозависимых свойств.

Значение свойств данного вида зависят от типа устройства, на котором открыт Web-клиент.

Пример использования в коде:

```
private PropertyDescription GetDeviceDependentPropertyDescription () {
    var propertyDescription = new PropertyDescription {
        Type = typeof (int),
        Name = "DeviceDependent",
        Category = PropertyCategoryConstants.DataCategory,
        DisplayName = "Device Dependent",
        DeviceDependent = true,
        DefaultValue = DeviceDependentHelper.GetDeviceDependentDefaultValue(1)
    };
    return propertyDescription;
}
```

Стандартные стили

Модуль предоставляет возможность при разработке собственного Решения использовать стандартные стили, что позволяет сохранить общую цветовую схему.

Стандартные стили:

- `.card-type-background-color` — определяет фон элемента типа карточки;
- `.card-type-background-color-hover` — определяет фон элемента типа карточки при наведении курсора;
- `.card-type-background-color-light` — определяет светлый фон элемента типа карточки;
- `.card-type-background-color-light-hover` — определяет светлый фон элемента типа карточки при наведении мыши;
- `.card-type-background-color-disabled` — определяет заблокированный фон элемента типа карточки (например, когда кнопка недоступна для нажатия);
- `.card-type-foreground-color` — определяет цвет шрифта типа карточки.

Пример использования стилей:

```
<div class="card-type-background-color">
 <a href="/opencard" class="card-type-foreground-color">Открыть</a>
</div>
```

Описание проекта TemplateWebExtension

Проект `TemplateWebExtension` является частью репозитория WebClient-Samples. Данный проект предназначен для разработки и сборки клиентских расширений и скриптов.

Описание файлов проекта

В папке "TemplateWebExtension" расположены файлы:

- `copy-path.js` — содержит константы с названием расширения и параметрами сборки; в данном файле указывается каталог, в который выгружаются файлы при сборке.
- `gulpfile.js` — файл конфигурации Gulp (используется для сборки стилей);
- `package.json` — файл конфигурации NPM с командами сборки и списком зависимостей.

- `package-lock.json` — служебный файл NPM.
- `rollup.config.js` — файл конфигурации Rollup. Используется для сборки скриптов в один файл — "бандл".
- `tsconfig.json` — файл конфигурации TypeScript.
- `node_modules` — служебная папка NPM, в которую при выполнении команды npm `install` загружаются npm-зависимости.
- `src` — папка для размещения исходных файлы расширения.

Папка src содержит файлы:

- `EventHandlers.tsx` — пример файла с обработчиком события.
- `Index.ts`-- файл с обязательной функцией регистрации клиентского расширения.
- `Template.scss` — пример файла стилей.

Пример рассчитан на версию Web-клиента dv6 или выше.

Перечень необходимых инструментов:

- NodeJS {node-v}
- Microsoft Visual Studio 2022.

Описание файла Index.ts

Содержимое файла `Index.ts`:

```
import * as EventHandlers from "./EventHandlers";
import { extensionManager } from "@docsvision/webclient/System/ExtensionManager";

extensionManager.registerExtension({ ①
    name: "Template web extension",
    version: "1.0",
    globalEventHandlers: [ EventHandlers ]
})
```

① Регистрация расширения позволяет корректно установить все обработчики событий, сервисы и прочие сущности веб-приложения.

Команда `import * as EventHandlers from "./EventHandlers"` импортирует все публичные сущности (с `export`) из файла `EventHandlers` (`.tsx`).

Чтобы импортировать функции из другого файла, необходимо добавить новую команду импорта, указав относительный путь к файлу:

```
import * as EventHandlers from "./EventHandlers";
import * as NewEventHandlers from "./Scripts/NewEventHandlers"; ①
```

① Добавлена строчка импорта из файла /Scripts/NewEventHandlers.

Команда `import { extensionManager } from "@docsvision/webclient/System/ExtensionManager"` импортирует из модуля "@docsvision/webclient/System/ExtensionManager" сущность extensionManager (менеджер расширений).

Команда `extensionManager.registerExtension({ ⋯ })` регистрирует клиентское расширение с названием `Template web extension`, которое предоставляет глобальные обработчики (`globalEventHandlers`).

Чтобы добавить в расширение обработчики из нового файла, добавьте их в массив `globalEventHandlers`:

```
globalEventHandlers: [ EventHandlers, NewEventHandlers] ①
```

① Добавлены обработчики из `NewEventHandlers`.

Описание файла EventHandlers.tsx

Содержимое файла:

```
import { IEventArgs } from "@docsvision/webclient/System/IEventArgs";
import { Layout } from "@docsvision/webclient/System/Layout";

export function someHandler(sender: Layout, e: IEventArgs) { ①
    ...
}
```

① Все функции, классы и переменные, используемые за пределами модуля (т.е. файла), должны экспортироваться (содержать ключевое слово export в объявлении).

Команды `import` в начале файла выполняют загрузку из модулей Web-клиента сущностей, используемых в коде обработчика.

Функция `someHandler` является примером обработчика события разметки (тип `sender` — `Layout`). Ключевое слово `export` указывает, что данная функция доступна

для импорта. Чтобы экспортировать другую функцию, необходимо также указывать ключевое слово `export`:

```
export function someHandler(sender: Layout, e: IEventArgs) {

}

export function newHandler(sender: Employee, e: IEventArgs) { ①
}
```

① Добавлена новая функция для обработки события в ЭУ с типом `Employee`.

Для создания клиентского скрипта достаточно добавить функции-обработчики в файл `EventHandlers.tsx`. Например, следующий код, вызванный при изменении значения элемента управления `Сотрудник`, отобразит информационное окно с ФИО выбранного сотрудника:

```
import { Employee } from "@docsvision/webclient/BackOffice/Employee";
import { IEventArgs } from "@docsvision/webclient/System/IEventArgs";
import { MessageBox } from "@docsvision/webclient/Helpers/MessageBox/MessageBox"; ①

export function changeEmployeeHandler(sender: Employee, e: IEventArgs) { ②
  MessageBox.ShowInfo(sender.params.value.displayName);
}
```

① Импорт необходимых сущностей из модулей Web-клиента.

② Реализация функции-обработчика.



Функции-обработчики должны экспортироваться — сопровождаться ключевым словом `export`.

Сервисы получения моделей объектов

Следующие сервисы предназначены для получения на клиенте моделей объектов для программного изменения значений элементов управления.

- `$DirectoryDesignerRowController` — предоставляет методы для получения объектов *Конструктора справочников*:
  - `getDirectoryDesignerRow(rowId: string)` — получает объект *Строка справочника* для строки с идентификатором `rowId`.

    - `getDirectoryDesignerRows(rows: GenModels.CollectionRequestModelOf<string>)` — получает объекты *Строка справочника* для строк с идентификаторами rows.
- `$CardKindController` — предоставляет методы для получения объектов *Справочника видов карточек*:
    - `getCardKind(cardKindId: string)` — получает объект *Вид карточки* для вида с идентификатором `cardKindId`.
    - `getCardKinds(cardKinds: GenModels.CollectionRequestModelOf<string>)` — получает объекты *Вид карточки* для видов с идентификаторами `cardKinds`.
- `$DepartmentController` — предоставляет методы для получения объектов *Справочника сотрудников* и *Справочника контрагентов*:
    - `getStaffDepartment(departmentId: string)` — получает объект *Подразделение* для подразделения с идентификатором `departmentId`.
    - `getPartnerDepartment(departmentId: string)` — получает объект *Подразделение контрагента* для подразделения с идентификатором `departmentId`.
    - `getStaffDepartments(departments: GenModels.CollectionRequestModelOf<string>)` — получает объекты *Подразделение* для подразделений с идентификаторами `departments`.
    - `getPartnerDepartments(departments: GenModels.CollectionRequestModelOf<string>)` — получает объекты *Подразделение контрагента* для подразделений с идентификаторами `departments`.
- `$EmployeeController` — предоставляет методы для получения объектов *Справочника сотрудников*:
    - `getEmployee(employeeId: string, options?: RequestOptions)` — получает объект *Сотрудник* для сотрудника с идентификатором `employeeId`.
    - `getEmployees(employees: GenModels.CollectionRequestModelOf<string>, options?: RequestOptions)` — получает объекты *Сотрудник* для сотрудников с идентификаторами `employees`.
- `$PartnerEmployeeController` — предоставляет методы для получения объектов *Справочника контрагентов*:
    - `getPartnerEmployee(employeeId: string, options?: RequestOptions)` — получает объект Сотрудник контрагента для сотрудника с идентификатором `employeeId`.

- `getPartnerEmployees(employees: GenModels.CollectionRequestModel, options?: RequestOptions)` — получает объекты *Сотрудник контрагента* для сотрудников с идентификаторами `employees`.

Следующий код демонстрирует пример получения объекта *Подразделение* с идентификатором `6832216B-FED9-4D20-B1D4-0097C73D446E` при помощи сервиса `$DepartmentController`.

```
export function superHandler(sender: CustomButton, e: IEventArgs) {
    let departmentControllerService = sender.layout.getService($DepartmentController);
    let departmentModel = departmentControllerService.getStaffDepartment("6832216B-FED9-
4D20-B1D4-0097C73D446E");
}
```



Подробное описание сервисов приведено в справочнике по JS API.

Список специальных адресов Web-клиента

Адреса, перечисленные в данном разделе, позволяют вызвать специальные функции Web-клиента из адресной строки браузера.

Работа со строками справочников

- Открыть окно просмотра строки справочника: `#/RowView/%cardId%/%sectionId%/%rowId%`.
- Открыть окно создания строки справочника: `#/NewRow/%cardId%/%sectionId%/%parentRowId%/%parentTreeRowID%`.
- Открыть окно редактирования строки справочника: `#/RowEdit/%cardId%/%sectionId%/%rowId%`.

Здесь:

- `%cardId%` — идентификатор карточки.
- `%sectionId%` — идентификатор секции карточки cardId.
- `%rowId%` — идентификатор редактируемой/открываемой строки.
- `%parentRowId%` — идентификатор родительской строки родительской секции (если секция находится внутри другой секции).
- `%parentTreeRowID%` — идентификатор родительской строки текущей секции (в случае, если секция древовидная).

Работа со Справочником контрагентов

Добавление:

- Открыть окно добавления корневой организации: `#/NewOrganizationRow/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/c78abded-db1c-4217-ae0d-51a400546923/00000000-0000-0000-0000-000000000000/00000000-0000-0000-0000-000000000000`.
- Открыть окно добавления дочерней организации: `#/NewOrganizationRow/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/c78abded-db1c-4217-ae0d-51a400546923/00000000-0000-0000-0000-000000000000/%parent-id%`.

  Здесь `%parent-id%` — идентификатор родительской организации.

- Открыть окно добавления подразделения: `#/NewDepartmentRow/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/c78abded-db1c-4217-ae0d-51a400546923/00000000-0000-0000-0000-000000000000/%parent-id%`.

  Здесь `%parent-id%` — идентификатор организации.

- Открывает окно добавления сотрудника: `#/NewRow/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/1a46bf0f-2d02-4ac9-8866-5adf245921e8/%parent-id%/00000000-0000-0000-0000-000000000000`.

  Здесь `%parent-id%` — идентификатор организации/подразделения.

Просмотр:

- Открывает окно просмотра сотрудника: `#/RowView/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/1a46bf0f-2d02-4ac9-8866-5adf245921e8/%id%`.

  Здесь `%id%` — идентификатор сотрудника.

- Открывает окно просмотра организации: `#/OrganizationRowView/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/c78abded-db1c-4217-ae0d-51a400546923/%id%`.

  Здесь `%id%` — идентификатор организации.

- Открывает окно просмотра подразделения: `#/DepartmentRowView/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/c78abded-db1c-4217-ae0d-51a400546923/%id%`.

  Здесь `%id%` — идентификатор подразделения.

Редактирование:

- Открывает окно редактирования сотрудника: `#/RowEdit/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/1a46bf0f-2d02-4ac9-8866-5adf245921e8/%id%`.

  Здесь `%id%` — идентификатор сотрудника.

- Открывает окно редактирования организации: `#/OrganizationRowEdit/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/c78abded-db1c-4217-ae0d-51a400546923/%id%`.

  Здесь `%id%` — идентификатор организации.

- Открывает окно редактирования подразделения: `#/DepartmentRowEdit/65ff9382-17dc-4e9f-8e93-84d6d3d8fe8c/c78abded-db1c-4217-ae0d-51a400546923/%id%`.

  Здесь `%id%` — идентификатор подразделения.

Механизм внедрения зависимостей на клиенте

В данном разделе рассмотрены особенности передачи клиентских сервисов с использованием механизма внедрения зависимостей.

У сервиса имеется имя и интерфейс. Например, сервис для получения информации о текущей карточке имеет имя `$CardInfo`, и интерфейс `GenModels.CardInfoModel`. Получить сервис можно по имени при помощи метода `getService` у объекта элемента управления, разметки или глобального объекта `app`. Например, таким образом можно получить сервис `$CardInfo` в обработчике события:

```
export function onCustomButtonClick(sender: CustomButton, args: IEventArgs) {
    const cardInfo = sender.getService($CardInfo);
}
```

Сервисы, как правило, размещаются в контейнерах — специальных объектах. Имя сервиса — это ключ, по которому данный сервис можно получить в сервис-контейнере. Получить сервис-контейнер можно вызвав `getService` без параметров. Так альтернативный способ получения сервиса по ключу из контейнера будет выглядеть так:

```
export function onCustomButtonClick(sender: CustomButton, args: IEventArgs) {
    const cardInfo = sender.getService()[$CardInfo] as GenModels.CardInfoModel; ①
    const cardInfo2 = sender.getService<$CardInfo>().cardInfo; ②
}
```

① Получаем либо по ключу

② Либо иным образом

Сервис-контейнер уровня приложения доступен через глобальный объект app, однако его не рекомендуется использовать в большинстве случаев.

Сервис контейнер в элементах управления также доступен через параметр `services` (он же `props` при отрисовке элемента управления как React-компонента).

Все стандартные сервисы Web-клиента (приведены в JSDoc API).

Создание сервисов

Определение сервиса включает в себя три составляющих:

- Описание интерфейса сервиса и его имени.
- Определение реализации.
- Регистрация реализации.

1. Первое, что нужно сделать — выбрать название сервиса. Допустим, наш сервис будет называться `Greeting`. Тогда описание интерфейса и имени сервиса будет выглядеть следующим образом:

```
export interface IGreetingService {
     showGreeting(name: string): Promise<void>;
}
export type $Greeting = { greeting: IGreetingService };
export const $Greeting = serviceName<$Greeting, IGreetingService>(x => greeting);
```

2. Реализацию сервиса рекомендуется определять в отдельном от интерфейса файле (чтобы не создавать лишние зависимости). Реализацией будет определение класса, реализующего интерфейс `IGreetingService`.

```
export class GreetingSevice implements IGreetingService {
      constructor(private services: $MessageWindow) {
      }
      showGreeting(name: string): Promise<void>{
            return this.services.messageWindow.showInfo("Добро пожаловать, " +
name);
      }
 }
```

3. Регистрируются сервисы в файле `Index.ts` веб-расширения через вызов `registerExtension`. Если сервис имеет отношение к разметке, то следует передавать его в поле `layoutServices`, иначе в `globalServices`.

```
extensionManager.registerExtension({
      name: "Greeting",
      version: "1.0",
      layoutServices: [
           Service.fromFactory($Greeting,
                (services: $StandardServices & $StandardControllers) =>
                     new GreetingSevice(services))
      ]
})
```

Стандартные web-сервисы

В данном разделе рассмотрены основы работы с сервисами, а также основные web-сервисы, доступные для использования в web-расширениях.

Сервисы представляют основной API взаимодействия с системой в части функциональности, не связанной с конкретными элементами управления. Они позволяют получить информацию о текущем контексте выполнения, выполнять запросы к серверу, показывать модальные окна, взаимодействовать с разметкой и использовать множество других функций системы.

Сервисы информации о текущей карточке/строке/разметке

Информацию о текущей разметке можно получить с помощью следующих сервисов:

- `$CardId` — идентификатор текущей карточки текущей разметки.
- `$RowId` — идентификатор текущей строки текущей разметки (если открыта разметка строки, а не карточки)
- `$CardTimestamp` — `timestamp` текущей карточки текущей разметки.
- `$CardInfo` — дополнительная информация о текущей карточке, текущей разметки. Например, там можно найти тип и вид карточки, ее состояние и другие данные.
- `$RowInfo` — дополнительная информация о текущей строке, текущей разметки (если открыта разметка строки).

- `$LayoutInfo` — дополнительная информация о текущей разметке.
- `$Layout` — экземпляр текущей разметки. Содержит множество различных методов, среди которых сохранение (`save`), перезагрузка (`reloadFromServer`), смены состояния карточки (`changeState`) и другие.
- `$ControlStore` — предоставляет доступ к элементам управления на разметке по имени через методы `get` и `tryGet`. Первый метод генерирует ошибку, в случае отсутствия элемента управления с таким именем, второй просто возвращает `null`.
- `$EditOperationStore` — позволяет получить информацию о доступности операции редактирования.

Сервисы информации об окружении

Информацию в целом о текущем контексте работы приложения можно с помощью следующих сервисов:

- `$CurrentEmployeeId` — идентификатор текущего пользователя.
- `$CurrentEmployeeAccountName` — имя учётной записи текущего пользователя.
- `$CurrentEmployee` — расширенная информация о текущем пользователе. Состав полей можно найти в описании типа `CurrentEmployeeInfo`.
- `$ApplicationTimestamp` — строка, изменяющаяся при перезапуске веб-сервиса.
- `$ApplicationSettings` — системные настройки (см. `GenModels.ApplicationSettings`).
- `$Locale` — двухбуквенное обозначение текущей выбранной локализации веб-клиента (например: `ru`, `en`).
- `$CurrentFolder` — информация о текущей открытой папке (см. `ICurrentFolderService`).
- `$DeviceType` — обозначение текущего выбранного типа устройства (компьютер/телефон/планшет). См. `GenModels.DeviceType`.

Сервисы для выполнения запросов

Выполнение запросов к серверу осуществляется через сервис `$RequestManager`. Сформировать URL запроса поможет сервис `$UrlResolver`.

Подробнее см. "Сервисы для выполнения запросов".

Сервисы для отображения сообщений

Показать на экране простое информационное сообщение можно с помощью сервиса `$MessageWindow`. Основные методы этого сервиса: `showInfo`, `showWarning`, `showError`. Они различаются только цветом оформления и отображают диалоговое окно.

Подробнее см. "Сервисы для выполнения запросов".

Сервис навигации

Для перехода между страницами существует сервис `$RouterNavigation`. Он предоставляет следующие методы:

```
export interface IRouterNavigation {
    refresh(): Promise<void>;
    goTo(route: string, refresh?: boolean): Promise<void>;
    openUrl(url: string, mode: GenModels.OpenMode): Promise<void>;
    back(): Promise<void>;
}
```

- `refresh` — перезагружает содержимое текущей страницы. Например, если открыта карточка, то перезагружается разметка карточки.
- `goTo` — открывает страницу по указанному хеш-адресу. Для получения корректного адреса можно использовать следующие функции:
  - `getCardViewRoute`, `getCardEditRoute`, `getCardCreateRoute` - для отображения разметки карточки.
  - `makeFolderUrl` - для открытия папки
  - `getViewDirectoryRoute` - для открытия справочника
  - `getRowEditRoute`, `getRowViewRoute`, `getRowCreateRoute` — для отображения разметок строки. Опциональный параметр refresh определяет поведение, в случае если переданный ардес совпадает с текущим. Если `refresh = true`, то в такой ситуацию будет произведено обновление страницы.
- `openUrl` — расширение метода `goTo`, добавляющее возможность открывать адрес в новой вкладке или новом окне (параметр `mode`). При открытии в текущей вкладке работает также, как `goTo`.
- `back` — возврат на предыдущую страницу.

Настройка роутинга осуществляется с помощью сервиса `$Router`. В нем

присутствуют методы для регистрации своих обработчиков. Подробнее об этом сервисе будет рассказано в другой статье, посвященной реализации собственных страниц.

Изменение основного шрифта Web-клиента

Основным шрифтом Web-клиента является "Roboto". По умолчанию поддерживаются два уровня жирности шрифта: стандартный (`400`) и жирный (`700`).

Если требуется, можно изменить настройки шрифта по умолчанию:

1. Добавьте новые шрифты или шрифт *Roboto* с требуемой жирностью в подпапку своего *Решения* в `{Каталог установки Web-клиент}\Content`.
2. Откройте для редактирования файл `{Каталог установки Web-клиент}\Content\fonts.html`.

```
<link rel="preload" href="Content/App/Assets/Fonts/Roboto-Regular.ttf" as="font"
type="font/ttf" crossorigin>
<style type="text/css" media="screen, print">
    @font-face {
        font-family: "Roboto";
        src: url("Content/App/Assets/Fonts/Roboto-Regular.ttf") format("truetype"),
            url('Content/App/Assets/Fonts/Roboto-Bold.ttf') format("truetype");
    }

    body {
        font-family: "Roboto";
        font-style: normal;
        font-weight: 400;
        font-display: swap;  /* IE11 */
    }
</style>
```

3. Измените настройки шрифта:
   - Если нужно добавить шрифты "Roboto" с новой жирностью — укажите их в `@font-face`:

```
@font-face {
        font-family: "Roboto";
        src: url("Content/App/Assets/Fonts/Roboto-Regular.ttf") format("truetype"),
            url('Content/App/Assets/Fonts/Roboto-Bold.ttf') format("truetype"),
            url('Content/Modules/SomeSolution/Fonts/Roboto-Black.ttf') format
```

```
("truetype");
    }
```

- Если нужно изменить шрифт по умолчанию:
  a. Добавьте новое правило `@font-face` со своим шрифтом аналогично правилу для шрифта *Roboto*.
  b. Укажите название шрифта в разделе `body`.

4. Перезапустите **dvwebclient** для Linux и **Docsvision Web-client** для Windows.



При обновлении Web-клиента файл `fonts.html` будет перезаписан — после обновления нужно повторить изменения.

Библиотека классов

Руководство по разработке решений Web-клиента

Библиотека классов

Описание API Web-клиента Docsvision.

- [WebClient:classLib-AdvancedCardManager]
- [WebClient:classLib-ControlTypeDescription]
- [WebClient:classLib-CommonResponse]
- Подробнее описание библиотеки классов API Web-клиента в навигационном меню слева…

AdvancedCardManager — класс

Класс `AdvancedCardManager` предоставляет собственные методы Web-клиента для работы с карточками с использованием API Docsvision.

Пространство имён: `DocsVision.Platform.WebClient.Managers`

Сборка: `DocsVision.Platform.WebClient.dll`

Синтаксис

```
public class AdvancedCardManager
```

Методы

Имя	Описание
`CardReadAccessCheck(Guid)`	Проверяет наличие прав на чтение карточки `cardId`.
`ChangeState(IServiceProvider, Guid, Guid, CardOperationArguments)`	Изменяет состояние карточки `cardId` при применении операции `operationId`. Если при изменении состояния была обработанная ошибка, возвращает её описание.
`CheckCardExists(Guid)`	Проверяет существование карточки `cardId`.
`CheckCardTimestamp(Guid, System.Int64)`	Проверяет актуальность штампа времени `timestamp` для карточки `cardId`.
`CheckFieldNullValueAllowed(Guid, Guid, String)`	Проверяет возможность установки пустого значения в поле `fieldAlias`, секции `sectionId`, типа карточки `cardTypeId`.
`CheckRowTimestamp(Guid, Guid, System.Int64)`	Проверяет актуальность штампа времени `timestamp` для строки `rowId` секции `sectionId`.
`CreateCard(Guid, Guid?)`	Создаёт карточку типа `cardTypeId`, вида `cardKindId`. Возвращает идентификатор новой карточки.
`CreateCard(Guid)`	Создаёт карточку по шаблону `templateId`. Возвращает идентификатор новой карточки.
`DeleteCard(Guid, Boolean)`	Удаляет карточку `cardId` и все её дочерние карточки. Для удаления в корзину параметр `permanently` должен быть `false`.

Имя	Описание
`DeleteRow(Guid, Guid, Guid)`	Удаляет строку `rowId` из секции sectionId карточки `cardId`.
`DisableReadOnlyDictionary(Guid)`	Добавляет справочник cardTypeId в список справочников, для которых запрещен режим открытия "только для чтения". Проверка осуществляется методом `IsReadonlyDictionaryEnabled`.
`GetCard<T>(Guid, Boolean)`	Возвращает карточку `cardId` в формате объекта `T`. При установленном флаге `refresh` обновляет кэш для получения актуальных данных.
`GetCardCreateDate(Guid)`	Возвращает дату создания карточки `cardId`.
`GetCardData(Guid, Boolean)`	Возвращает данные карточки `cardId`. При установленном флаге `refresh` обновляет кэш для получения актуальных данных.
`GetCardTimestamp(Guid, Boolean)`	Возвращает штамп времени карточки `cardId`. При установленном флаге `refresh` обновляет кэш для получения актуальных данных.
`GetCardTimestampModel(Guid, Boolean)`	Возвращает штамп времени карточки `cardId` в формате модели `CommonResponseTimestamp`. При установленном флаге `refresh` обновляет кэш для получения актуальных данных.
`GetCardTypeId(Guid)`	Возвращает тип карточки `cardId`.
`GetDictionary<T>(Guid, Boolean)`	Возвращает справочник `cardTypeId` в формате объекта `T`.

Имя	Описание
`GetDictionaryAsReadonly<T>(Guid, Boolean)`	Возвращает справочник `cardTypeId` в формате объекта `T` и режим чтения. Если данные справочника будут изменяться, получайте его методом `GetDictionary`.
`GetDictionaryData(Guid, Boolean)`	Возвращает данные справочника `cardTypeId`.
`GetDictionaryDataAsReadonly(Guid, Boolean)`	Возвращает данные справочника `cardTypeId` в формате объекта `T` и режим чтения. Если данные справочника будут изменяться, получайте их методом `GetDictionaryData`.
`GetFieldDescription(Guid, Guid, String)`	Возвращает отображаемое локализованное название поля `fieldAlias` секции `sectionId` карточки `cardTypeId`.
`GetFieldValue(Guid, Guid, String)`	Возвращает значение поля `fieldAlias` секции `sectionId` карточки `cardTypeId`.
`GetFieldValue<T>(Guid, Guid, String)`	Возвращает значение поля `fieldAlias` секции `sectionId` карточки `cardTypeId` в формате объекта `T`.
`GetFieldValue(CardData, Guid, String)`	Возвращает значение поля `fieldAlias` секции `sectionId` из данных карточки `cardData`.
`GetOperations(Guid)`	Возвращает список операций (разрешённых и запрещённых) карточки в формате `OperationModel`.
`GetRow(Guid, Guid, Guid)`	Возвращает данные строки rowId секции `sectionId` карточки `cardId`.
`GetRowData(Guid, Guid, Guid, Boolean)`	Возвращает данные строки `rowId` секции `sectionId` карточки `cardId`. При установленном флаге `refresh` обновляет кэш для получения актуальных данных.

Имя	Описание
`GetRowTimestamp(Guid, Guid)`	Возвращает штамп времени для строки `rowId` секции `sectionId`.
`GetSection(Guid, Guid)`	Возвращает данные секции `sectionId` карточки `cardId`.
`IsOperationAllowed(Guid, BuiltInOperation)`	Проверяет возможность выполнения операции `builtInOperation` для карточки `cardId`.
`MarkAsRead(Guid)`	Устанавливает на карточку `cardId` признак "прочитана".
`SetFieldValue(Guid, Guid, String, Object)`	Устанавливает для поля `fieldAlias` секции `sectionId` карточки `cardId` значение `value`.
`SetWasRead(Guid, Boolean)`	Устанавливает на карточку `cardId` признак "прочитана"/"не прочитана" (`setRead`).
`TryGetCardSystemInfo(Guid)`	Возвращает данные системной секции карточки `cardId`.
`UpdateDigest(Guid, String)`	Обновляет дайджест карточки. Для карточек Базовые объекты и производных дайджест формируется в соответствии с настройками формирования дайджеста в Конструкторе разметок или устанавливает значение по умолчанию `defaultDigest`. Для других типов карточек в дайджест записывается название типа карточки.

ControlTypeDescription — класс

Класс `ControlTypeDescription` содержит данные описателя элемента управления.

Пространство имён: `DocsVision.Platform.Tools.LayoutEditor.ObjectModel.Descriptions`

Сборка: `DocsVision.LayoutEditor.ObjectModel.dll`

Синтаксис

```
public class ControlTypeDescription
```

Конструкторы

Имя	Описание
`ControlTypeDescription(String)`	Инициализирует новый экземпляр класса `ControlTypeDescription` с заданным названием типа элемента управления.

Свойства

Имя	Описание
`CheckAsChild`	Задаёт или возвращает функцию, которая проверяет возможность добавления данного элемента управления в другой элемент управления.
`CheckAsParent`	Задаёт или возвращает функцию, которая проверяет возможность добавления одного элемента управления в другой.
`ControlResolverType`	Задаёт или возвращает название типа резолвера для получения модели элемента управления.
`ControlTypeName`	Задаёт или возвращает тип элемента управления.
`ControlValidator`	Задаёт или возвращает функцию валидации элемента управления при сохранении разметки.
`DisplayName`	Задаёт или возвращает отображаемое название элемента управления.

Имя	Описание
GetAllowedOperations	Задаёт или возвращает функцию, которая проверяет возможность совершения операции с элементом управления при настройке разметки.
IsBlockLevel	Задаёт или возвращает признак того, что элемент управления относится к элементам разметки.
IsParentConrol	Задаёт или возвращает признак того, что элемент управления может являться родительским для других элементов управления.
OnChildAdd	Задаёт или возвращает функцию, которая выполняется при добавлении дочернего элемента управления.
OnPropertyChange	Задаёт или возвращает функцию обработчик изменения настроек элемента управления.
PropertyDescriptions	Возвращает коллекцию описателей свойств элемента управления.
Validator	Задаёт или возвращает функцию валидации разметки, вызываемую при сохранении разметки.
Visualizer	Задаёт или возвращает класс компонента, представляющего графический интерфейс элемента управления в редакторе разметок.

Методы

Имя	Описание
CreateProperties(String)	Заполняет свойства экземпляра элемента управления из свойств типа ЭУ при добавлении ЭУ на разметку.

CommonResponse — класс

Класс `CommonResponse` содержит ответ на вызов метода контроллера.

Пространство имён: `DocsVision.Platform.WebClient.Models`

Сборка: `DocsVision.Platform.WebClient.dll`

Синтаксис

```
public class CommonResponse
```

Свойства

Имя	Описание
`Success`	Признак завершения запроса: `true` — запрос завершен, `false` — запрос не выполнен, произошла ошибка.
`Timestamp`	Новый штамп времени, если при выполнении запроса была изменена карточка.
`TimestampCardId`	Идентификатор карточки, если при выполнении запроса была изменена карточка.
`Message`	Содержимое сообщения об ошибке, возникшей при выполнении запроса, или предупреждения — при завершении запроса.
`ExtendedDataSourceInfos`	Информация о дополнительных источниках данных.

Методы

Имя	Описание
`CreateSuccess`	Создаёт экземпляр объекта `CommonResponse` в состоянии Выполнено. В перегрузках метода в `CommonResponse` могут передаваться значения: `timestamp` — штамп времени. `sessionContext` — контекст сессии. `cardId` — идентификатор карточки. `extendedDataSourceInfoModels` — информация о дополнительных источниках данных. `data` — данные, возвращаемые запросом. `sectionId`, `rowId` — идентификаторы сессии и строки.
`CreateError`	Создаёт экземпляр объекта `CommonResponse` в состоянии Ошибка. В message передаётся информация об ошибке.
`InitializeSuccess`	Устанавливает `CommonResponse` в состояние Выполнено. В перегрузке метода также задаётся значение `Timestamp` и `TimestampCardId`.
`InitializeError`	Устанавливает `CommonResponse` в состояние Ошибка.

Пример

Ниже приведён код метода контроллера, в котором выполняется изменение карточки.

```
[HttpPost]
public CommonResponse AddReference([FromBody]AddReferenceRequestModel model)
{
    var currentObjectContextProvider = ServiceUtil.GetService
<ICurrentObjectContextProvider>(this.serviceProvider);
    var sessionContext = currentObjectContextProvider.GetOrCreateCurrentSessionContext();

    var responseModel = new CommonResponse(); ①

    var currentCardTimestamp = sessionContext.AdvancedCardManager.GetCardTimestamp(model
.CardId); ②

    if (currentCardTimestamp == model.CardTimestamp) ③
    {
④

        var commonResponseTimestamp = new CommonResponseTimestamp ⑤
        {
            CardId = model.CardId,
            Timestamp = sessionContext.AdvancedCardManager.GetCardTimestamp(model.CardId)
        };
        responseModel.InitializeSuccess(commonResponseTimestamp);
    }
    else
    {
        responseModel.InitializeError("Сообщение клиенту об ошибке"); ⑥
    }

    return responseModel;
}
```

① Создание модели `CommonResponse`.

② Получение текущего штампа времени.

③ Проверка актуальности данных загруженной карточки.

④ Здесь выполняются действия с карточкой.

⑤ Обновляется штамп времени в ответе.

⑥ Возврат ошибки.

В приведённом коде запрос со стороны клиента включает штамп времени изменяемого объекта (`model.CardTimestamp`), что позволяет убедиться в актуальности клиентских данных.

Если данные на клиенте не актуальны (например, карточка была изменена

другим пользователем) — клиенту возвращается ответ (`CommonResponse`) с состоянием `Ошибка`.

Если данные на клиенте соответствуют данным на сервере — выполняется операция с карточкой. При этом в ответе клиенту отправляется новый штамп времени (`CommonResponseTimestamp`) — на клиентской стороне хранится в поле `cardInfo.timestamp` (см. JSDocAPI).

NotificationRealtimeMessage — класс

Класс NotificationRealtimeMessage содержит данные сообщения, передаваемого с помощью сервиса IRealtimeCommunicationService.

Пространство имён: `DocsVision.Platform.WebClient.Models.RealTimeCommunication.NotificationMessage`

Сборка: `DocsVision.Platform.WebClient.dll`

Синтаксис

```
public class NotificationRealtimeMessage
```

Конструкторы

Имя	Описание
`NotificationRealtimeMessage()`	Инициализирует новый экземпляр класса `NotificationRealtimeMessage`.

Свойства

Имя	Описание
`NotificationType`	Тип передаваемого сообщения.
`Message`	Возвращает менеджер для работы с блокировками.

PropertyCategoryConstants — класс

Класс `PropertyCategoryConstants` предоставляет названия стандартных категорий для списка свойств элемента управления.

Пространство имён: `DocsVision.Platform.Tools.LayoutEditor.Infrostructure`

Сборка: `WebLayoutsDesigner.exe`

Синтаксис

```
public static class PropertyCategoryConstants
```

Поля

Имя	Описание
`AppearanceCategory`	Внешний вид.
`BehaviorCategory`	Поведение.
`DataCategory`	Данные.
`DefaultValueCategory`	Значение по умолчанию.
`EventsCategory`	События.
`GeneralCategory`	Общие.

PropertyDescription — класс

Класс `PropertyDescription` содержит описание свойства элемента управления в расширении программы Конструктор Web-разметок.

Пространство имён: `DocsVision.Platform.Tools.LayoutEditor.ObjectModel.Descriptions`

Сборка: `DocsVision.LayoutEditor.ObjectModel.dll`

Синтаксис

```
public class PropertyDescription : ICloneable
```

Конструкторы

Имя	Описание
`PropertyDescription(String)`	Инициализирует новый экземпляр класса `PropertyDescription`.

Свойства

Имя	Описание
`BindingResolverType`	Зарезервировано.
`Category`	Задаёт или возвращает категорию свойства.
`ChangeLocalizaionKeys`	Задаёт или возвращает функцию обновления ключей локализации при копировании свойства с сериализованным значением.
`CheckAvailablity`	Задаёт или возвращает функцию проверки доступности свойства для настройки в текущем контексте (к примеру, в текущем типе разметки).
`CollectLocalizationKeys`	Задаёт или возвращает функцию получения нужных ключей локализации при копировании свойства с сериализованным значением.
`DefaultValue`	Задаёт или возвращает значение по умолчанию.
`DeviceDependent`	Задаёт или возвращает флаг, указывающий, зависит ли значение свойства от типа клиентского устройства.
`DisplayName`	Задаёт или возвращает строковый ресурс, предоставляющий локализованное название элемента управления.
`Editor`	Задаёт или возвращает тип компонента редактирования (редактора) значения свойства. Редактор является компонентом UserControl с реализованным интерфейсом `Xceed.Wpf.Toolkit.PropertyGrid.Editors.ITypeEditor` (сборка `Xceed.Wpf.Toolkit.Fixed.dll`).

Имя	Описание
`ExcludeFromIL`	Задаёт или возвращает флаг, указывающий, будет ли данное свойство исключено из пакета свойств элемента управления, передаваемых в клиентскую часть Web-клиента. Свойство с установленным флагом `ExcludeFromIL` остается настраиваемым в программе Конструктор Web-разметок.
`GenerateDefaultValue`	Задаёт или возвращает метод, генерирующий для свойства значение по умолчанию.
`GenerateValueOnPaste`	Задаёт или возвращает метод, генерирующий значение свойства при копировании элемента управления, содержащего данное свойство.
`GetBindingLoader`	Задаёт или возвращает функцию, реализующую механизм получения и сохранения значения свойства.
`Hidden`	Задаёт или возвращает флаг, определяющий, является ли свойство скрытым. Скрытое свойство не отображается при настройке элемента управления.
`IsTemporarilyReadOnly`	Задаёт или возвращает функцию, проверяющую является ли элемент управления доступным только на чтение.
`ItemsSource`	Задаёт или возвращает тип компонента, который предоставляет список готовых значений свойства, из которого значение свойства выбирается.

Имя	Описание
`Localized`	Задаёт или возвращает флаг, указывающий, является ли свойство локализуемым. Свойство должно быть строкового типа. Для редактирования значения локализуемого свойства в Editor должен быть указан тип редактора `LocalizedPropertyEditor`.
`Name`	Задаёт или возвращает уникальное название свойства.
`Priority`	Задаёт или возвращает порядок отображения свойства в текущей категории свойств. Чем выше порядок, тем выше будет отображено свойство.
`PropertyResolverType`	Задаёт или возвращает название компонента, предоставляющего дополнительную возможность настройки свойства: `DeviceDependent` (зависящего от типа устройства), `Localized` (зависящего от локализации). Рекомендуется использовать вместо `PropertyResolverType` флаги `Localized` и `DeviceDependent` класса `PropertyDescription`.
`ReadOnly`	Задаёт или возвращает флаг, указывающий, является ли значение элемента управления доступным только на чтение.
`RefreshProperties`	Задаёт или возвращает флаг, указывающий, требуется ли обновлять отображаемое значение свойства при изменении значения. В `true` следует устанавливать, если для изменения свойства используется отдельное окно редактирования.

Имя	Описание
`Type`	Задаёт или возвращает тип значения свойства.
`Validate`	Задаёт или возвращает функцию валидации значения свойства.

Примеры

Следующий код демонстрирует пример свойства `Slider`, добавляемого в категорию *Данные*. Для свойства назначается специализированный редактор значения — `SliderEditor`.

```
private PropertyDescription GetSliderPropertyDescription()
{
 return new PropertyDescription ①
 {
  Type = typeof(string), ②
  Name = Constants.SampleImage.Slider, ③
  Category = PropertyCategoryConstants.DataCategory, ④
  DisplayName = Resources.ControlTypes_SliderProperty, ⑤
  DefaultValue = String.Empty, ⑥
  ExcludeFromIL = true, ⑦
  Editor = typeof(SliderEditor), ⑧

  CollectLocalizationKeys = args => ⑨
  {
   var value = (string)args.ControlInfo.GetProperty(args.PropertyDescription.Name); ⑩
   var items = SliderModeHelper.StringToItems(value); ⑪
   var keys = args.LocalizationKeys; ⑫
   foreach (var item in items) ⑬
   {
    if (!string.IsNullOrEmpty(item.DescriptionLocalizationKey))
     keys.Add(item.DescriptionLocalizationKey);
   }
  },
  ChangeLocalizationKeys = args => ⑭
  {
   var value = (string)args.ControlInfo.GetProperty(args.PropertyDescription.Name);

   var items = SliderModeHelper.StringToItems(value);
   var mapping = args.LocalizationMapping;
   foreach (var item in items)
   {
    if (!string.IsNullOrEmpty(item.DescriptionLocalizationKey))
```

```
    {
     string newKey;
     if (mapping.TryGetValue(item.DescriptionLocalizationKey, out newKey)) ⑮
      item.DescriptionLocalizationKey = newKey;
     }
    }
    return SliderModeHelper.ItemsToString(items); ⑯
   }
  };
 }
```

① Создаем экземпляр класса описания свойства .

② Определяем строковый тип данного свойства ЭУ. Реальное значение сериализуемое.

③ Указывается уникальное название.

④ Категория для размещения свойства Данные.

⑤ Отображаемое название свойства.

⑥ Пустое значение по умолчанию.

⑦ Свойство не должно передаваться на клиентскую сторону Web-клиента.

⑧ Указываем специфический редактор значения свойства.

⑨ Определяем метод, который будет вызываться при копировании свойства. Метод получает ключи локализации DescriptionLocalizationKey из объекта свойства args.PropertyDescription.Name и заполняет ими коллекцию args.LocalizationKeys.

⑩ Получаем значение требуемого свойства.

⑪ Десериализуем значение свойства в нужный класс.

⑫ Получаем из параметров (передается в args) элемента управления текущей разметки.

⑬ Заполняем keys ключами из описания свойства.

⑭ Определяем метод, который будет вызываться при копировании элемента управления и, соответственно, его свойств.

⑮ Создаем связь.

⑯ Сериализуем значение свойства.

SessionContext — класс

Класс `SessionContext` содержит методы и свойства контекста сессии.

Пространство имён: DocsVision.Platform.WebClient

Сборка: DocsVision.Platform.WebClient.dll

Синтаксис

```
public sealed class SessionContext : IServiceProvider
```

Конструкторы

Имя	Описание
SessionContext(IRequestContextProvider)	Инициализирует новый экземпляр класса SessionContext.

Свойства

Имя	Описание
AdvancedCardManager	Возвращает менеджер для работы с карточками.
AdvancedLockManager	Возвращает менеджер для работы с блокировками.
BaseAlias	Возвращает псевдоним базы данных, к которой выполнено подключение.
FullTextSearchEnabled	Возвращает флаг, указывающий, что включено полнотекстовое индексирование.
ObjectContext	Возвращает контекст объектов.
Session	Возвращает пользовательскую сессию.
UserInfo	Возвращает информацию о текущем пользователе.

Методы

Имя	Описание
RegisterObjectContextCreateHandler (Action<ObjectContext>)	Добавляет дополнительный обработчик для события инициализации контекста объектов.

Имя	Описание
`Close`	Освобождает объекты контекста объектов и пользовательской сессии.
`DisposeObjectContext`	Освобождает объект контекста объектов.
`GetService(Type)`	Зарезервирован.
`ResetRolesCache(ObjectBase)`	Вызывает метод `ResetRolesCache` системы Docsvision для сброса кэша ролевой модели.
`RollbackChanges`	Отменяет несохранённые изменения контекста объектов.

UserInfo — класс

Класс `UserInfo` содержит данные пользователя.

Пространство имён: `DocsVision.Platform.WebClient`

Сборка: `DocsVision.Platform.WebClient.dll`

Синтаксис

```
public class UserInfo
```

Поля

Имя	Описание
`AccountName`	Возвращает имя учётной записи пользователя.
`AccountSID`	Возвращает идентификатор безопасности пользователя.
`Employee`	Возвращает модель данных сотрудника.
`EmployeeId`	Возвращает идентификатор сотрудника.
`Roles`	Возвращает роли сотрудника.

Имя	Описание
`UserProfileCardId`	Возвращает идентификатор карточки настроек пользователя.

WebClientExtension — класс

Класс `WebClientExtension` является базовым классом расширения программы Конструктор Web-разметок.

Пространство имён: `DocsVision.WebClient.Extensibility`

Сборка: `DocsVision.WebClient.Extensibility.dll`

Синтаксис

```
public abstract class WebClientExtension
```

Конструкторы

Имя	Описание
`WebClientExtension(IServiceProvider)`	Инициализирует новый экземпляр класса WebClientExtension.

Свойства

Имя	Описание
`BindingConverters`	Возвращает коллекцию зарегистрированных в расширении конвертеров значений элементов управления. **Свойство устарело.**
`BindingResolvers`	Возвращает коллекцию зарегистрированных в расширении "резолверов" (`resolvers`) значений элементов управления и выполнения обратной процедуры. **Свойство устарело.**

Имя	Описание
`CardFactories`	Возвращает коллекцию зарегистрированных в расширении фабрик карточек. **Свойство устарело.**
`ControlResolvers`	Возвращает коллекцию зарегистрированных в расширении "резолверов" (`resolvers`) элементов управления. **Свойство устарело.**
`ExtensionName`	Возвращает название расширения.
`ExtensionVersion`	Возвращает версию расширения.
`LayoutExtensionResourceManagers`	Возвращает зарегистрированные в расширении диспетчеры ресурсов, ресурсы которых могут быть получены клиентскими компонентами Web-клиента.
`Namespace`	Возвращает пространство имён расширения. **Свойство устарело.**
`NavigatorExtension`	Возвращает информацию о данном серверном расширении. **Свойство устарело. Используйте класс `WebClientExtension`, свойства `ExtensionName` и `ExtensionVersion`.**
`PropertyResolvers`	Возвращает коллекцию зарегистрированных в расширении "резолверов" (`resolvers`) свойств элементов управления. **Свойство устарело.**

Имя	Описание
`RegisteredResourceManagers`	Возвращает зарегистрированные в расширении диспетчеры ресурсов, ресурсы которых могут быть получены в расширении Web-клиента.
`ServiceActivators`	Возвращает активаторы для реализованных в расширении сервисов. **Свойство устарело.**
`ServiceProvider`	Возвращает экземпляр сервис-провайдера. **Свойство устарело. Используйте метод с перезагрузкой `ILifetimeScope`.**

Методы

Имя	Описание
`OnLoad(IServiceProvider)`	Обработчик загрузки серверного расширения. Вызывается после загрузки расширения. Метод помечен устаревшим, работает. **Используйте метод с перезагрузкой `ILifetimeScope`.**
`OnLoad(ILifetimeScope lifetimeScope)`	Обработчик загрузки серверного расширения. Вызывается после загрузки расширения. Описание типа аргумента метода на странице https://autofac.org/apidoc/html/BA5D3489.htm. Метод устарел, не работает и не требуется, регистрация осуществляется в `InitializeServiceCollection`.

Имя	Описание
`OnObjectContextCreate(ObjectContext)`	Обработчик для события инициализации контекста объектов. Переопределение метода можно использовать для добавления новых фабрик библиотек карточек и фабрик сервисов в используемый Web-клиентом контекст объектов.
`GetApiControllerActivators(IServiceProvider)`	Предоставляет активаторы для реализованных в расширении контроллеров WebApi. Все контроллеры сборки регистрируются автоматически при загрузке расширения.
`GetBindingConverters`	Предоставляет методы получения значений элементов управления для передачи в их веб-компоненты. Через переопределение метода можно добавить функцию изменения значения элемента управления до его передачи клиенту (в браузер). **Устарел. Используйте IoC-контейнер.**

Имя	Описание
`GetBindingResolvers`	Предоставляет методы разрешения значений элементов управления и выполнения обратной процедуры. Через переопределение метода можно добавить функцию изменения значения элемента управления до его передачи клиенту (при получении), а также изменения значения перед его передачей серверу Web-клиента (при сохранении). **Устарел. Используйте IoC-контейнер.**
`GetCardFactories`	Предоставляет фабрики карточек, которые реализуют методы создания новых экземпляров карточек заданных типов. **Функция `GetCardFactories` устарела**: используйте `GetCardLifeCycles`.
`GetCardLifeCycles`	Предоставляет методы создания карточек, проверки данных карточки перед сохранением. **Устарел. Используйте IoC-контейнер.**
`GetControllerActivators(IServiceProvider)`	Предоставляет активаторы для реализованных в расширении контроллеров MVC. Все контроллеры сборки регистрируются автоматически при загрузке расширения.

Имя	Описание
`GetControlResolvers`	Предоставляет методы формирования моделей данных для веб-компонентов элементов управления. Через переопределение метода можно внести изменения в стандартный процесс загрузки свойств и значений свойств элемента управления. **Устарел. Используйте IoC-контейнер.**
`GetLayoutExtensionResourceManagers`	Предоставляет диспетчеры ресурсов, ресурсы которых могут быть получены клиентскими компонентами (клиентскими скриптами и веб-компонентами элементов управления).
`GetNavigatorExtension`	Предоставляет информацию о данном расширении. Используйте класс `WebClientExtension`, свойства `ExtensionName` и `ExtensionVersion`.
`GetPropertyResolvers`	Предоставляет методы получения значений свойств элементов управления. Через переопределение метода можно внести изменения в стандартный процесс загрузки значений свойств. **Устарел. Используйте IoC-контейнер.**
`GetRegisteredResourceManagers`	Предоставляет диспетчеры ресурсов, ресурсы которых могут быть получены в расширении Web-клиента.

Имя	Описание
`GetRowLifeCycles`	Предоставляет методы создания строк секций, проверки данных строки перед сохранением. **Устарел. Используйте IoC-контейнер.**
`GetServiceActivators(IServiceProvider)`	Предоставляет активаторы для реализованных в расширении сервисов. **Устарел. Используйте IoC-контейнер.**

WebLayoutsDesignerExtension — класс

Класс `WebLayoutsDesignerExtension` является базовым классом расширения программы Конструктор Web-разметок.

Пространство имён: `DocsVision.Platform.Tools.LayoutEditor.Extensibility`

Сборка: `WebLayoutsDesigner.exe`

Синтаксис

```
public abstract class WebLayoutsDesignerExtension : IWebLayoutsDesignerExtension
```

Конструкторы

Имя	Описание
`WebLayoutsDesignerExtension(IServiceProvider)`	Инициализирует новый экземпляр класса `WebLayoutsDesignerExtension`.

Свойства

Имя	Описание
AllowedOperations	Возвращает коллекцию функций, определяющих возможность совершения в программе Конструктор Web-разметок определённой операции с элементом управления.
ControlTypeDescriptions	Возвращает коллекцию описаний элементов управления, добавляемых в программы Конструктор Web-разметок данным расширением.
Editors	Возвращает коллекцию редакторов значений свойств элементов управления.
PropertyDescriptions	Возвращает коллекцию свойств и событий, добавляемых данным расширением.
ResourceManagers	Возвращает коллекцию ссылок на файлы ресурсов, предоставляемых данным расширением.
Visualizers	Возвращает коллекцию компонентов, содержащих графическое представление для отображения ЭУ в программе Конструктор Web-разметок.
PropertyFactory	Возвращает ссылку на фабрику свойств и событий элемента управления.
ServiceContainer	Возвращает ссылку на контейнер сервисов.

Методы

Имя	Описание
GetAllowedOperations	Метод должен возвращать коллекцию функций, определяющих возможность совершения в программе Конструктор Web-разметок определённой операции с элементом управления.

Имя	Описание
`GetControlTypeDescriptions`	Метод должен возвращать коллекцию описаний элементов управления, добавляемых в программы Конструктор Web-разметок данным расширением.
`GetEditors`	Метод должен возвращать коллекцию редакторов значений свойств элементов управления.
`GetPropertyDescriptions`	Метод должен возвращать коллекцию свойств и событий, добавляемых данным расширением.
`GetResourceManagers`	Метод должен возвращать коллекцию ссылок на файлы ресурсов, предоставляемых данным расширением.
`GetVisualizers`	Метод должен возвращать коллекцию компонентов, содержащих графическое представление для отображения ЭУ в программе Конструктор Web-разметок.

IApplicationTimestampService — интерфейс

Интерфейс `IApplicationTimestampService` определяет методы сервиса для работы со штампами времени приложения.

Пространство имён: `DocsVision.WebClient.Services`

Сборка: `DocsVision.BackOffice.WebClient.dll`

Синтаксис

```
public interface IApplicationTimestampService
```

Свойства

Имя	Описание
`ApplicationTimestamp`	Возвращает текущий штамп времени.

Методы

Имя	Описание
UpdateApplicationTimeStamp	Обновляет штамп времени, вызывающий обновление пользовательского кэша приложения.

Примечание

Реализация метода UpdateApplicationTimeStamp используется при вызове веб-метода GET http://Адрес-сервера-Web-клиента/WebClient/Navigator/ResetClientCache. Если требуется, стандартная реализация сервиса может быть переопределена собственной.

Стандартная реализация

```
public class ApplicationTimestampService : IApplicationTimestampService
{
    private string applicationTimestamp;

    public string ApplicationTimestamp =>
        applicationTimestamp ?? (applicationTimestamp = WebClientApplication
.ApplicationStartTimestamp);

    public void UpdateApplicationTimeStamp()
    {
        applicationTimestamp = DateTime.Now.Ticks.ToString();
    }
}
```

ICardLifeCycleEx — интерфейс

Интерфейс ICardLifeCycleEx определяет методы для работы с карточками из разметок.

Пространство имён: DocsVision.WebClientLibrary.ObjectModel.Services.EntityLifeCycle

Сборка: DocsVision.WebClientLibrary.ObjectModel.dll

Синтаксис

```
public interface ICardLifeCycleEx
```

Методы

Имя	Описание
`CardTypeId { get }`	Идентификатор целевой карточки для текущего жизненного цикла. **Тип:** `Guid`.
`Create(SessionContext, CardCreateLifeCycleOptions)`	Создаёт новую карточку и возвращает её идентификатор. **Тип:** `Guid`. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно. Обычно `OperationContext` является экземпляром `CreateRowOperationContext`.
`Validate(SessionContext, CardValidateLifeCycleOptions, out List<ValidationResult>)`	Проверяет данные карточки перед сохранением изменений. Возвращает `true`, если ошибки не обнаружены, иначе — `false`. **Тип:** `Boolean`. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно; `validationResults` — массив для формирования сообщения о результатах проверки.
`OnSave(SessionContext, CardSaveLifeCycleOptions)`	Обрабатывает событие сохранения карточки. **Параметры:** `sessionContext` — контекст сессии; `cardId` — идентификатор карточки.

Имя	Описание
`CanDelete(SessionContext, CardDeleteLifeCycleOptions, out String)`	Проверяет возможность удаления карточки. Возвращает `true`, если удаление возможно, иначе — `false`. **Тип:** `Boolean`. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно; `message` — сообщение о невозможности удаления.
`OnDelete(SessionContext, CardDeleteLifeCycleOptions)`	Обрабатывает событие удаления карточки. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно.
`GetDigest(SessionContext, CardDigestLifeCycleOptions);`	Метод генерации дайджеста. **Тип:** `String`. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно.

ICardsPresentationExtension — интерфейс

Интерфейс `ICardsPresentationExtension` представляет метод получения описания представления карточки.

Пространство имён: `DocsVision.WebClientLibrary.ObjectModel.Services.EntityLifeCycle`

Сборка: `DocsVision.WebClientLibrary.ObjectModel.dll`

Синтаксис

```
public interface ICardsPresentationExtension
```

Методы

Имя	Описание
`ExtensionName { get; set; }`	Идентификатор целевой карточки для текущего жизненного цикла. **Тип:** `String`.
`GetPresentation(SessionContext, CardsPresentationRequest);`	Получает представление карточки. **Тип:** `CardPresentation`. **Параметры:** `sessionContext` — контекст сессии; `request` — возвращает представление.

IImageGenerator — интерфейс

Для данной версии системы интерфейс недействителен.

ILinksService — интерфейс

Интерфейс `ILinksService` определяет методы сервиса для работы со ссылками карточек — сервиса ссылок.

Пространство имён: `DocsVision.BackOffice.WebClient.Links`

Сборка: `DocsVision.BackOffice.WebClient.dll`

Синтаксис

```
public interface ILinksService
```

Методы

Имя	Описание
`AddReference(SessionContext, Guid, Guid, Guid, Boolean)`	Добавляет ссылку на карточку `referencedCardId` с типом ссылки `linkTypeId` в список ссылок `referenceListCardId`. Если `hardLink` в значении `true`, будет добавлена жесткая ссылка.
`AddUrlReference(SessionContext, Guid, String, Guid)`	Добавляет ссылку на адрес `url` в список ссылок `referenceListCardId`. В `linkTypeId` может быть указан тип ссылки.
`GetAllowedCreateLinkKinds(SessionContext)`	Возвращает модели видов карточек, на которые можно создать ссылки.
`GetAllowedLinkCardTypes`	Возвращает идентификаторы типов карточек, на которые можно создать ссылки.
`GetLinkModels(SessionContext, Guid, Guid?, LinksDescriptionColumnViewMode, String)`	Возвращает модели ссылок для списка ссылок `referenceListCardId`. В параметре `descriptionColumnViewMode` указывается способ формирования описания ссылок, а в `descriptionColumnMethodKey` название метода формирования описания ссылок.
`GetLinkTypes(SessionContext)`	Возвращает модели типов ссылок.
`GetOrCreateReferenceListCardId(SessionContext , Guid, Guid, String)`	Возвращает или создаёт список ссылок карточки `cardId`. Идентификатор карточки Список ссылок может быть получен из поля `fieldAlias` секции `sectionId`.
`GetReferenceListCardId(SessionContext, Guid, Guid, String)`	Возвращает идентификатор карточки Список ссылок карточки `cardId`. Идентификатор карточки Список ссылок может быть получен из поля `fieldAlias` секции `sectionId`.

Имя	Описание
`RegisterDescriptionColumnGenerator(String, DescriptionColumnGeneratorDelegate)`	Регистрирует метод генерации описаний ссылок. В `key` должно быть указано название метода, по которому он может быть запрошен элементом Ссылки, в `delegateFunction` — делегат на метод генерации описания ссылки.
`RemoveReference(SessionContext, Guid, Guid, Guid)`	Удаляет ссылку `referenceId` из списка ссылок `referenceListCardId` карточки `cardId`. Возвращает true, если ссылка была удалена, иначе — `false`.
`UpdateReferenceDescription(SessionContext, Guid, String)`	Изменяет описание ссылки `referenceId` на `description`.

IRealtimeCommunicationService — интерфейс

Интерфейс `IRealtimeCommunicationService` определяет методы для работы с подсистемой оповещений Web-клиента.

Пространство имён: `DocsVision.Platform.WebClient.Services`

Сборка: `DocsVision.Platform.WebClient.dll`

Синтаксис

```
public interface IRealtimeCommunicationService
```

Методы

Имя	Описание
`GetCurrentSessionId`	Возвращает идентификатор текущей сессии в ASP NET. **Тип:** `String`.
`Send<T>(String, RealtimeCommunicationMessage<T>)`	Отправляет сообщение пользователю с идентификатором сессии sessionId в ASP NET.

Имя	Описание
`Send<T>(Guid, RealtimeCommunicationMessage<T>)`	Отправляет сообщение пользователю с идентификатором employeeId в *Справочнике сотрудников*.
`SendAll<T>(RealtimeCommunicationMessage<T>)`	Отправляет оповещение `message` всем подключенным пользователям (с активной и неактивной сессией) Web-клиента.
`SendToCurrentSession<T>(RealtimeCommunicationMessage<T>)`	Отправляет оповещение `message` текущему пользователю.

События

Имя	Описание
`Connected`	Вызывается при подключении пользователя. **Тип:** `EventHandler<RealtimeConnectionEventArgs>`.
`Disconnected`	Вызывается при отключении пользователя. **Тип:** `EventHandler<RealtimeConnectionEventArgs>`.
`MessageReceived`	Вызывается при получении оповещения. **Тип:** `EventHandler<RealtimeMessageEventArgs>`.
`MessageSent`	Вызывается при отправке оповещения. **Тип:** `EventHandler<RealtimeCommunicationMessage>`.

Примечание

Подсистема оповещений основана на библиотеке функций реального времени SignalR.

Отправленное сообщение будет показано в правом нижнем углу. Цветовое оформление сообщения зависит от типа сообщения, указанного в `RealtimeCommunicationMessage`.

В Web-клиент реализован единственный тип сообщения, передаваемого с помощью SignalR — `NotificationRealtimeMessage`.

На клиентской стороне аналогичную сервису `IRealtimeCommunicationService` функциональность предоставляет клиентский сервис `$RealtimeCommunicationService`.

Примеры

Следующий код демонстрирует пример реализации в серверном расширении функции отправки сообщений всем пользователям.

```
public void SendAll()
{
    var commMessage = new RealtimeCommunicationMessage<NotificationRealtimeMessage>(); ①
    var messageData = new NotificationRealtimeMessage()
    {
        NotificationType = NotificationType.Info,
        Message = message
    };
    commMessage.Initialize(NotificationRealtimeMessage.MessageType, messageData);
    communicationService.SendAll(commMessage);
}
```

① Создаём объект-оповещение. Тип содержимого оповещения — `NotificationRealtimeMessage` (текстовое оповещение).

Возможные типы оповещений перечислены в описании перечисления NotificationType.

IPropertyFactory — класс

Интерфейс `IPropertyFactory` определяет методы фабрики свойств.

Пространство имён: `DocsVision.Platform.Tools.LayoutEditor.Infrostructure`

Сборка: `WebLayoutsDesigner.exe`

Синтаксис

```
public interface IPropertyFactory
```

Методы

Имя	Описание
`CanCreate(String)`	Проверяет возможность получения описателя для свойства с указанным названием. Возвращает `true`, если свойство может быть зарегистрировано, иначе — `false`. **Тип:** `Boolean`. **Параметры:** `propertyName` — название свойства.
`Create(String)`	Возвращает экземпляр описателя свойства с указанным названием. Свойство должно быть зарегистрировано в фабрике свойств. **Тип:** `PropertyDescription`. **Параметры:** propertyName — название свойства.

Имя	Описание
`CreateEvent(String, String, String, String)`	Возвращает экземпляр описателя свойства с указанным названием. Предназначен для получения описателя свойства-события. В отличие от метода `Create`, `CreateEvent` не получает свойство из зарегистрированных в фабрике свойств, а создаёт новый экземпляр описателя свойства. В возвращаемом описателе свойства установлены категория свойства (в "События") и тип значения (в `String`), свойственные свойству-событию элемента управления. **Тип:** `PropertyDescription`. **Параметры:** `propertyName` — название свойства; `resourceKey` — ключ ресурса, предоставляющего локализованное отображаемое название элемента управления; `displayName` — отображаемое название элемента управления — используется, если не указан `resourceKey`; `defaultValue` — значение по умолчанию для свойства.
`Register(String, PropertyDescription)`	Регистрирует новое свойство в фабрике свойств. **Параметры:** `propertyName` — название свойства; `propertyDescription` — описатель регистрируемого свойства.

IRowLifeCycleEx — интерфейс

Интерфейс `IRowLifeCycleEx` определяет методы для работы со строками секций из

разметок.

Пространство имён: `DocsVision.WebClientLibrary.ObjectModel.Services.EntityLifeCycle`

Сборка: `DocsVision.WebClientLibrary.ObjectModel.dll`

Синтаксис

```
public interface IRowLifeCycleEx
```

Методы

Имя	Описание
`SectionId { get }`	Идентификатор целевой секции для текущего жизненного цикла. **Тип:** `Guid`.
`Create(SessionContext, RowCreateLifeCycleOptions)`	Создаёт новую строку в секции и возвращает её идентификатор. **Тип:** `Guid`. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно. Обычно `OperationContext` является экземпляром `CreateRowOperationContext`.

Имя	Описание
`Validate(SessionContext, RowValidateLifeCycleOptions, out List<ValidationResult>)`	Проверяет данные строки перед сохранением изменений. Возвращает `true`, если ошибки не обнаружены, иначе — `false`. **Тип:** `Boolean`. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно; `validationResults` — массив для формирования сообщения о результатах проверки.
`OnSave(SessionContext, RowValidateLifeCycleOptions)`	Обрабатывает событие сохранения строки секции. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно.
`CanDelete(SessionContext, RowDeleteLifeCycleOptions, out string)`	Проверяет возможность удаления строки из секции. Возвращает `true`, если удаление возможно, иначе — `false`. **Тип:** `Boolean`. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно; `message` — сообщение о невозможности удаления.

Имя	Описание
`OnDelete(SessionContext, RowDeleteLifeCycleOptions)`	Обрабатывает событие удаления строки из секции. **Параметры:** `sessionContext` — контекст сессии; `options` — параметр формируется самостоятельно.

ISelectedLayoutService — интерфейс

Интерфейс `ISelectedLayoutService` определяет методы для работы с разметками.

Пространство имён: `DocsVision.Platform.Tools.LayoutEditor.Infrostructure`

Сборка: `WebLayoutsDesigner.exe`

Синтаксис

```
public interface ISelectedLayoutService
```

Методы

Имя	Описание
`SelectedLibraryId`	Возвращает идентификатор библиотеки карточек. **Тип:** `Guid`.
`SelectedCardTypeId`	Возвращает идентификатор типа карточек. **Тип:** `Guid`.
`SelectedKindId`	Возвращает идентификатор вида карточек. **Тип:** `Guid`.
`LayoutInfo`	Возвращает информацию о разметке. **Тип:** `ILayoutInfo`.

AllowedOperationsFlag — перечисление

Перечисление `AllowedOperationsFlag` определяет операции, которые могут быть выполнены с элементом управления при настройке разметки.

Пространство имён: `DocsVision.Platform.Tools.LayoutEditor.ObjectModel.Descriptions`

Сборка: `DocsVision.LayoutEditor.ObjectModel.dll`

Синтаксис

```
[Flags]
public enum AllowedOperationsFlag
```

Члены

Имя	Описание
`None`	Все операции запрещены. **Значение:** `0x00`.
`Create`	Элемент управления может быть добавлен. **Значение:** `0x01`.
`Delete`	Элемент управления может быть удалён. **Значение:** `0x02`.
`ChangeOrder`	Элемент управления может быть перемещен в пределах своего родительского элемента. **Значение:** `0x04`.
`Move`	Элемент управления может быть перемещен к другому родителю. **Значение:** `0x08`.

Имя	Описание
`AddChildControl`	В элемент управления может быть добавлен дочерний элемент управления. **Значение:** `0x10`.
`All`	Разрешены все операции с элементом управления. **Значение:** `0x1F`.

NotificationType — перечисление

Перечисление `NotificationType` определяет типы оповещений, которые могут быть отправлены с помощью сервиса IRealtimeCommunicationService.

Пространство имён: `DocsVision.Platform.WebClient.Models.RealTimeCommunication.NotificationMessage`

Сборка: `DocsVision.Platform.WebClient.dll`

Синтаксис

```
public enum NotificationType
```

Члены

Имя	Описание
`Alert`	Предупреждение. **Значение:** `0`.
`Success`	Успех. **Значение:** `1`.
`Warning`	Внимание. **Значение:** `2`.

Имя	Описание
`Error`	Ошибка. **Значение:** `3`.
`Info`	Информационное сообщение. **Значение:** `4`.

Информация

Тип оповещения влияет на цветовое оформление блока, в котором отображается оповещение.

DescriptionColumnGeneratorDelegate — делегат

Делегат `DescriptionColumnGeneratorDelegate` устарел, вместо него используйте `ICardsPresentationExtension`.

[0] Рекомендуется использовать Visual Studio Code 2022.

[1] Рекомендуется использовать Visual Studio Code 2022.

[2] Рекомендуется использовать Visual Studio Code 2022.

[3] Рекомендуется использовать Visual Studio Code 2022.

[4] Рекомендуется использовать Visual Studio Code 2022.

[5] Рекомендуется использовать Visual Studio Code 2022.

[6] Рекомендуется использовать Visual Studio Code 2022.

[7] Рекомендуется использовать Visual Studio Code 2022.

[8] Рекомендуется использовать Visual Studio Code 2022.

[9] Рекомендуется использовать Visual Studio Code 2022.

[10] См. `PropertyFactory.GetNameProperty()`

[11] См. свойство `Slider, ImageHeight)`

[12] См. `PropertyFactory.GetNameProperty()`

[13] См. свойство `Url`

[14] Перечень изменений в реализации исходного кода стандартных компонентов не предоставляется.

[15] Репозиторий Docsvision с примерами пакетов Debian расположен на GitHub.

[16] См. `PropertyFactory.GetNameProperty()`

[17] См. свойство `WidgetId`