



Модуль интеграции с операторами ЭДО

Версия и дата сборки: 5.5.4, 2026-04-20

Общие сведения о модуле

Модуль интеграции с операторами ЭДО является дополнительным модулем Docsvision, который обеспечивает возможность обмена электронными документами с контрагентами через операторов ЭДО (электронного документооборота).



Оператор ЭДО — организация, предоставляющая услуги по обмену открытой и конфиденциальной информацией по телекоммуникационным каналам связи в рамках электронного документооборота.

Под *электронным документом* здесь и далее подразумеваются формализованные и неформализованные юридически значимые электронные документы (если не указан определенный тип *электронного документа*).

Модуль поддерживает работу через Web-клиент или Windows-клиент.

Назначение и функции модуля

Модуль предоставляет следующие возможности:

1. Отправка электронных документов:

а. Формализованные:

- УПД ([универсальный передаточный документ](#)).
- Счет-фактура.
- Акт.
- Накладная.
- Универсальный корректировочный документ (УКД).
- Исправление УПД.
- Корректировочный счет-фактура (КСФ).
- Исправление счета-фактуры (ИСФ).

б. Неформализованные.

2. Получение электронных документов:

а. Формализованные:

- УПД.
- Счет-фактура.
- Акт.
- Накладная.
- УКД.
- Исправление УПД.
- КСФ.
- ИСФ.

б. Неформализованные:

- Акт.
- Акт сверки.
- Детализация.
- Договор.
- Доверенность.
- Доп. соглашение к договору.
- Накладная.
- Неформализованный.
- Протокол согласования цены.
- Реестр сертификатов.
- Счет.
- Ценовой лист.

3. Создание запросов на аннулирование документов, формирование ответов на запросы об аннулировании.
4. Отправка запросов на уточнение полученных документов.
5. Загрузка подтверждения оператора ЭДО о получении титула продавца (с отправкой извещения о получении).
6. Электронный обмен через одного или нескольких операторов ЭДО^[1].
7. Загрузка информации о контрагентах и подразделениях организации из системы оператора ЭДО.

Разделы документации

Документация модуля содержит следующие разделы:

1. *Модуль интеграции с операторами ЭДО.* Руководство администратора.
2. *Модуль интеграции с операторами ЭДО.* Руководство пользователя при работе в Windows-клиенте.
3. *Модуль интеграции с операторами ЭДО.* Руководство пользователя при работе в Web-клиенте.
4. *Модуль интеграции с операторами ЭДО.* Руководство разработчика.
5. *Модуль интеграции с операторами ЭДО.* Описание полей карточек модуля.

Обратите внимание:

1. В списке поддерживаемых типов электронных документов приведены все типы поддерживаемых модулей. Фактическая возможность обмена определенными типами документов зависит от поддержки работы с данными типами оператором ЭДО.
2. Все перечисленные возможности предоставляются только после установки коннектора к оператору ЭДО, с которым осуществляется взаимодействие. Если коннекторы не установлены, обмен электронными документами будет невозможен.

Структура модуля

Компоненты модуля

Модуль состоит из следующих компонентов:

1. Библиотека карточек *Обмен юридически значимыми документами с карточками:*
 - а. *Справочник настроек операторов ЮЗДО* — карточка Docsvision, в которой:
 - Настраиваются параметры подключения к сервисам операторов ЭДО.
 - Указываются программные компоненты с реализацией функций обмена информацией с операторами ЭДО через коннекторы к операторам ЭДО.
 - Указываются данные организаций-участников ЭДО.
 - б. *Карточка обмена сообщениями* — карточка Docsvision с данными документа, отправляемого оператору ЭДО, на основе которых

формируются электронные сообщения для оператора.

- c. *Карточка приглашения к обмену* — карточка Docsvision
- d. *Расширение навигатора ЮЗДО* — добавляет на ленту Windows-клиента раздел *ЮЗДО* с кнопками отправки и подписания неформализованных документов.

2. Программные компоненты (библиотеки DLL):

- a. Компонент чтения — содержит функцию получения данных из карточек определенного типа/вида.
Используется при создании *Карточек обмена сообщениями*.
- b. Компонент изменения — содержит функцию изменения данных в карточках определенного типа/вида.
Используется при изменении документов, по которым поступили ответы от оператора ЭДО.
- c. Компонент создания — содержит функцию создания карточек определенного типа/вида.
Используется при создании карточек в системе Docsvision по документам, поступившим от оператора ЭДО.
- d. Сервис сообщений — содержит функции отправки и получения электронных сообщений оператору ЭДО.
- e. Компонент импорта — содержит функции получения конфигурации от оператора ЭДО: списка подразделений организации и связанных организаций, списка контрагентов и ящиков.
- f. Компонент работы с подписями — содержит функции подписания электронных сообщений и текстовых комментариев, отправляемых оператору ЭДО.

Web-разметки для видов карточек

1. Исходящий УПД

- a. Исходящий УПД (внутри)
- b. Исходящий УПД (просмотр)
- c. Исходящий УПД (создание)
- d. Исходящий УКД (внутри)
- e. Исходящий УКД (просмотр)

- f. Исходящий УКД (создание)
 - g. Исходящее исправление УПД (внутри)
 - h. Исходящее исправление УПД (просмотр)
 - i. Исходящее исправление УПД (создание)
2. Входящий УПД
- a. Входящий УПД (просмотр)
 - b. Входящий УПД (создание)
 - c. Входящий УКД
3. Приглашения ЭДО:
- a. Входящее приглашение ЭДО
 - b. Исходящее приглашение ЭДО (отправлено)
 - c. Исходящее приглашение ЭДО (проект)
4. Формализованный акт.

Плагин для грида

В пакет инсталляции модуля входит плагин **EDIMessagesLog** для ЭУ **Контроль табличных данных**. Плагин преобразует элемент управления в *Журнал обмена ЮЗДО* для Web-клиент. Подробнее про ЭУ см. в `undefined/webclient/5.5.17/layouts/ctrl/table/dataGridControl/[документации модуля Web-клиент]`.

Сведения о связях между составными частями модуля

При выполнении в карточке *Документ действий*, предусматривающих необходимость передачи документа оператору ЭДО, формируется *Карточка обмена сообщениями*. Данные, необходимые для создания *Карточки обмена сообщениями*, предоставляются компонентом чтения, который получает их из отправляемого документа. Для чтения каждого вида карточек может использоваться собственный компонент чтения.

БП Отправка сообщений ЮЗДО

При появлении в системе новой *Карточки приглашения к обмену* БП *Отправка сообщений ЮЗДО* формирует исходящее сообщение электронного обмена для оператора ЭДО и передаёт его по каналу связи. При отправке сообщения используется реализация сервиса сообщений коннектора к данному оператору ЭДО (устанавливается отдельно).

БП Получение сообщений ЮЗДО

БП *Получение сообщений ЮЗДО* получает входящие сообщения от операторов ЭДО.

Способ обработки входящих сообщений зависит от их содержимого:

- При получении электронного документа — в Docsvision создаётся карточка документа.
- При получении квитанции, подтверждающей получение оператором ЭДО отправленного формализованного документа — изменяется статус исходной карточки документа, а пользователю предоставляется возможность отправить оператору ЭДО подтверждение о получении извещения.
- При получении ответа на отправленный неформализованный документ — изменяется статус обмена.

БП Перенос файлов из сообщений ЮЗДО

БП *Перенос файлов из сообщений ЮЗДО* переносит полученные от контрагентов титулы покупателя в карточки УПД. БП также переносит изменения состояния документов.

БП Отправка приглашений ЮЗДО

При появлении в системе новой *Карточки приглашения ЭДО* БП *Получение приглашений ЮЗДО* формирует входящее приглашение к обмену электронным документооборотом для оператора ЭДО. При отправке приглашений ЭДО используется БП *Отправка приглашений ЭДО*. Для получения и отправки приглашений к обмену ЭДО по каналу связи требуется наличие коннектора к оператору ЭДО (устанавливается отдельно).

БП Получение приглашений ЮЗДО

При появлении в системе новой *Карточки обмена сообщениями* БП *Получение приглашений ЮЗДО* формирует исходящее сообщение электронного обмена для оператора ЭДО и передаёт его по каналу связи. При отправке сообщения используется реализация сервиса сообщений коннектора к данному оператору ЭДО (устанавливается отдельно).

Сведения о связях с другими системами

Модуль интеграции с операторами ЭДО является промежуточным звеном в связи системы Docsvision с операторами ЭДО.

Модуль предоставляет публичные интерфейсы, которые используются системой

Docsvision для вызова функций коннекторов к операторам ЭДО. Коннекторы непосредственно подключаются к операторам ЭДО через каналы связи для передачи и получения электронных сообщений.



Рисунок 1. Связь модуля с другими системами

[1] Для работы с оператором ЭДО требуется коннектор к данному оператору (устанавливается отдельно).

Необходимые ресурсы

Необходимое техническое обеспечение

Специальные требования к аппаратному обеспечению не предъявляются.

Необходимое программное обеспечение

- СКЗИ "КриптоПро CSP" должно быть установлено на клиентах и сервере/серверах СУБП (Workflow). См. подробнее требования в документации модуля [undefined/backoffice/5.5.5/requirements/#crypto-pro\[Базовые объекты\]](#).

Требования к системе Docsvision

Модули Docsvision

1. Серверная и клиентская части модуля *Платформа* версии 5.5.1 сборка 3291 и выше.
2. Серверная и клиентская части модуля *Базовые объекты* версии 5.5.2 сборки 4236 и выше.
3. Серверная и клиентская части приложения *Управление документами* версии 5.5.2 сборка 4234 и выше.
4. Серверная и клиентская части модуля *Управление процессами* версии 5.5.1 сборка 3163 и выше.
5. Серверная и клиентская части модуля *Конструктор согласований* версии 5.5.1 сборка 3185 и выше.
6. Серверная и клиентская части приложения *Делопроизводство 4.5* версии 5.5.1 сборка 3161 и выше.
7. Серверная и клиентская части модуля *Windows-клиент* версии 5.5.1 сборка 3206 и выше.

Лицензия

Для работы с модулем *Модуль интеграции с операторами ЭДО*:

- Лицензия на систему Docsvision должна содержать дополнительную опцию *Модуль интеграции с операторами ЭДО*.
- В зависимости от используемой подписи на стороне на стороне клиента могут потребоваться различные компоненты КриптоПро, см. раздел

"undefined/backoffice/5.5.5/admin/system-settings/#signature-cypher[Подписи и шифрование]" в документации по администрированию модуля Базовые объекты.

Изменения, обновления и исправленные ошибки

Модуль интеграции с операторами ЭДО

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

Новые виды документов

Добавлены два новых вида документов: **Входящий УКД (736)** и **Исходящий УКД (736)** в соответствии с изменениями о прекращении поддержки старого формата УКД.

Дополнительные функции документа

Добавлены новые функции документа **СвРК = 3** и **СвЗК = 4**.

Если модуль устанавливался ранее в поле карточки документа **Данные УПД / Функция документа** (`UniversalDocumentData/DocumentFunction`) можно добавить новые значения.

Если модуль устанавливается впервые или после обновления была создана новая БД, дополнительных действий не требуется.

Изменения в API

- Добавлен новый интерфейс `IImportService2` для реализации постраничной загрузки контрагентов.
- Описан класс `undefined/programmer/dv5/Edi/api/BoxesService/[BoxesService]` для выполнения стандартных операций.

Класс дополнен новыми методами:

- `ImportPartner(OperatorsOperator, string, string)` — импортирует одного контрагента по ИНН и КПП.
- `ImportPartner(OperatorsOperator, PartnersCompany)` — импортирует одного контрагента по организации контрагента.
- `ProcessAllNewInvitations(ILogWriter)` — находит и обрабатывает все новые входящие приглашения.
- `ProcessAllNewInvitationsToSend(ILogWriter)` — находит и обрабатывает все новые исходящие приглашения.
- `CheckInvitationsStatus(ILogWriter, DateTime?)` — проверяет статус приглашений с указанной даты.
- Добавлены `undefined/programmer/dv5/Edi/api/web-server-methods/[методы]` для серверного и клиентского расширения Web-клиента.
- Добавлено описание для следующих классов:
 - `undefined/programmer/dv5/Edi/api/DataReaderService/[DataReaderService]`
 - `undefined/programmer/dv5/Edi/api/DocumentUpdaterService/[DocumentUpdaterService]`
 - `undefined/programmer/dv5/Edi/api/InvitationService/[InvitationService]`
 - `undefined/programmer/dv5/Edi/api/LoaderService/[LoaderService]`
 - `undefined/programmer/dv5/Edi/api/OutgoingMessagesService/[OutgoingMessagesService]`

Исправленные ошибки

Здесь

Пока нет ничего

Накопительные обновления

Администрирование модуля

Установка и настройка

Состав установочного комплекта

1. Пакет установки серверной части модуля: `EDIIntegrationServer.msi`.
2. Пакет установки клиентской части модуля: `EDIIntegrationClient.msi`.

Обратите внимание

Модуль поддерживает работу через Web-клиент. Для работы через интерфейс Web-клиента требуется установить *Компоненты Модуля интеграции с операторами ЭДО* в модуле Web-клиент версии 5.5.5957.431 для Web-клиент 16 и 5.5.6478.105 для Web-клиент 17 или выше.

Краткая справка по установке

1. **Установите** серверную часть модуля *Модуль интеграции с операторами ЭДО* согласно инструкции ниже.
2. **Установите** клиентскую часть модуля *Модуль интеграции с операторами ЭДО* по следующей инструкции.
3. **Установите** серверную и клиентскую части модуля *Web-клиент* версии 5.5.5957.431 для Web-клиент 16 и 5.5.6478.105 для Web-клиент 17 или выше.

При установке не забудьте включить *Компоненты модуля интеграции с операторами ЭДО*.

Компоненты модуля можно установить из режима инсталлятора *Изменить*^[1].

Установка серверной части модуля

Серверная часть модуля *Модуль интеграции с операторами ЭДО* устанавливается на компьютер с сервером Docsvision, а также **на все компьютеры** с отдельной установкой СУБП (модуль *Управление процессами*), на которых обрабатываются БП данного сервера Docsvision.

Установка должна выполняться от имени пользователя, являющегося локальным администратором и администратором Docsvision.



Администратор Docsvision — пользователь, включенный в группу безопасности **DocsVision Administrators** на компьютере с сервером Docsvision.

1. Запустите пакет установки **EDIIntegrationServer.msi**.

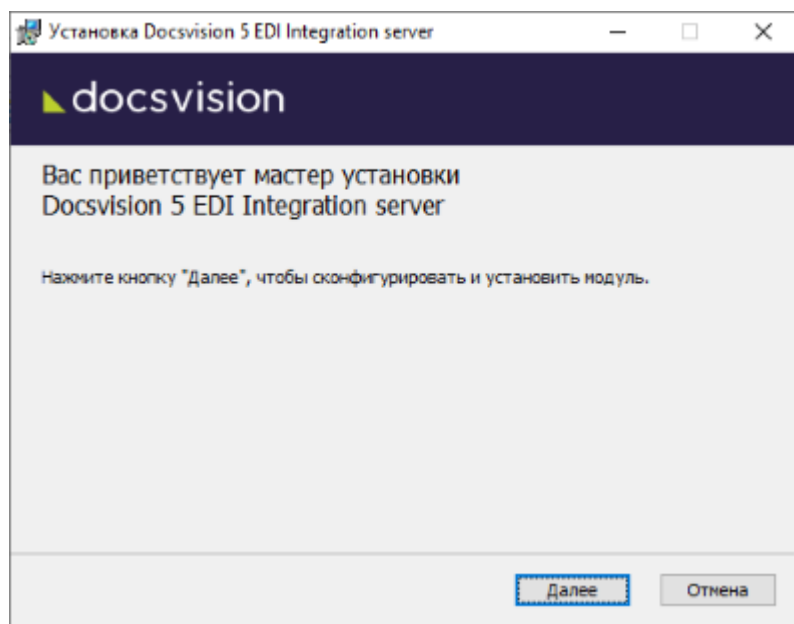


Рисунок 2. Мастер установки серверной части модуля

2. Примите условия лицензионного соглашения для продолжения установки. Нажмите **Далее**.

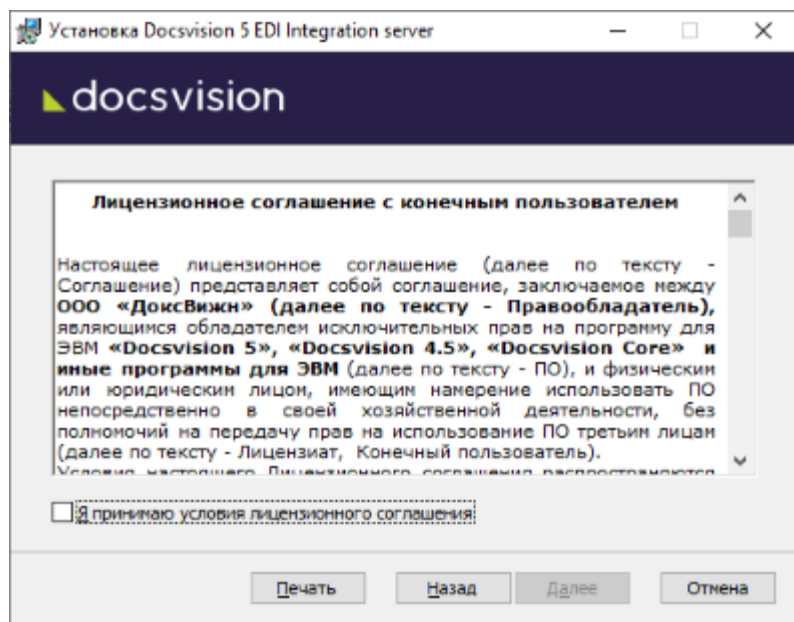


Рисунок 3. Условия лицензионного соглашения

3. В следующем окне нажмите **Установить** и дождитесь завершения установки модуля.
4. Выберите базы данных Docsvision, в которые будет установлен модуль, и нажмите **ОК**. В процессе загрузки также будет предложено перезагрузить сервисы Docsvision. Согласитесь с перезагрузкой сервисов.

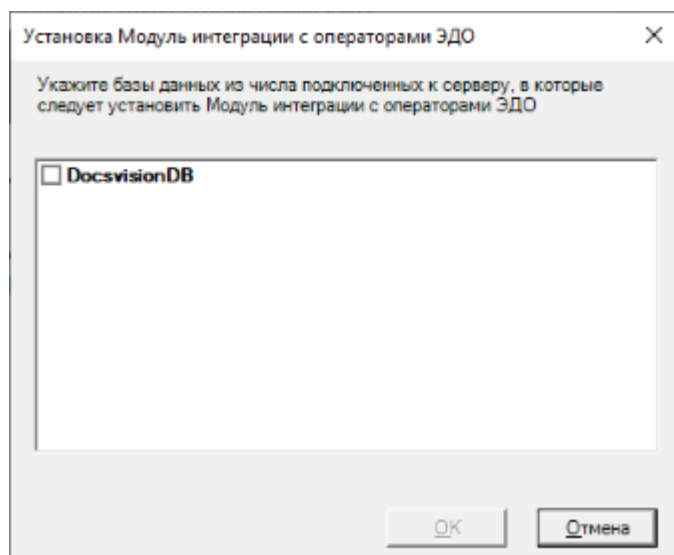


Рисунок 4. Окно выбора БД Docsvision для установки загружаемых компонентов модуля

5. Нажмите на кнопку **Готово**, чтобы закрыть мастер установки.

Установка клиентской части модуля

Клиентская часть модуля должна быть установлена на компьютеры пользователей Docsvision, работающих в Windows-клиенте. Если Windows-клиент

установлен для всех пользователей, то клиентский компонент модуля также должен быть установлен для всех пользователей.

Клиентская часть будет автоматически установлена при запуске Windows-клиента, а также может быть установлена вручную.

Чтобы установить клиентскую часть модуля "Модуль интеграции с операторами ЭДО" вручную:

1. Запустите пакет установки:

- `EDIIntegrationClient.msi`, чтобы установить модуль с областью установки *Для текущего пользователя*.
- `msiexec /i "EDIIntegrationClient.msi" ALLUSERS=1`, чтобы установить модуль с областью установки *Для всех пользователей*. Для данного варианта установки требуются административные права.

Будет открыт мастер установки.

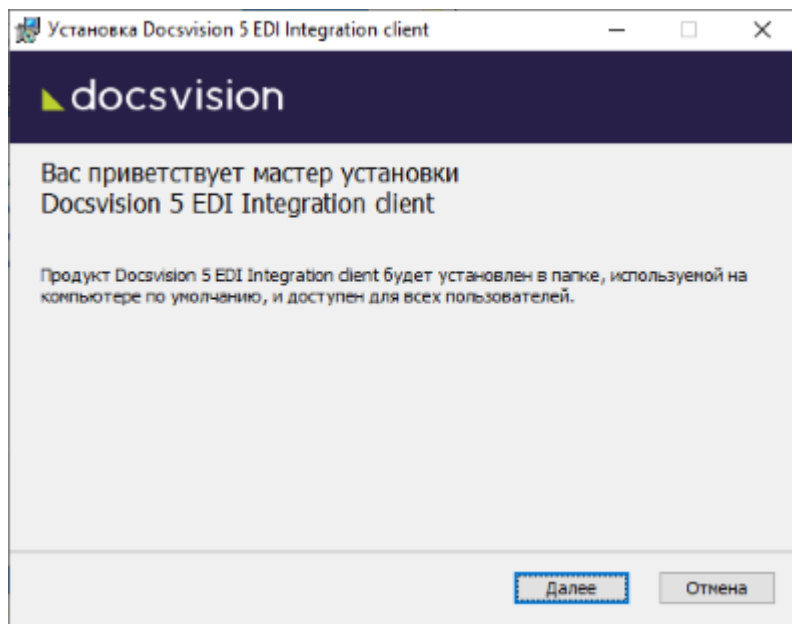


Рисунок 5. Мастер установки клиентской части "Модуля интеграции с операторами ЭДО"

2. Примите условия лицензионного соглашения для продолжения установки.

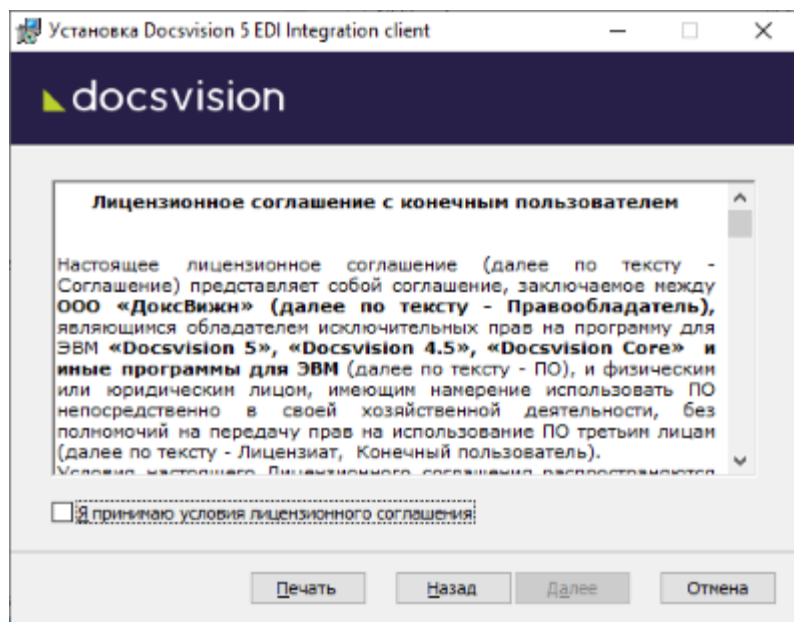


Рисунок 6. Условия лицензионного соглашения

3. В следующем окне нажмите **Установить** и дождитесь завершения установки модуля.
4. Нажмите на кнопку **Готово**, чтобы закрыть мастер установки.



Установите коннектор к оператору ЭДО (например, [Коннектор к Диадок](#)), через которого будет осуществляться обмен электронными документами, прежде чем приступить к настройке *Модуль интеграции с операторами ЭДО*.

Установка Web-клиента

Установите серверную и клиентскую часть модуля Web-клиент версии 5.5.5957.431 для Web-клиент 16 и 5.5.6478.105 для Web-клиент 17 стандартным образом, не забыв добавить *Компоненты модуля интеграции с операторами ЭДО* на шаге выбора компонентов.

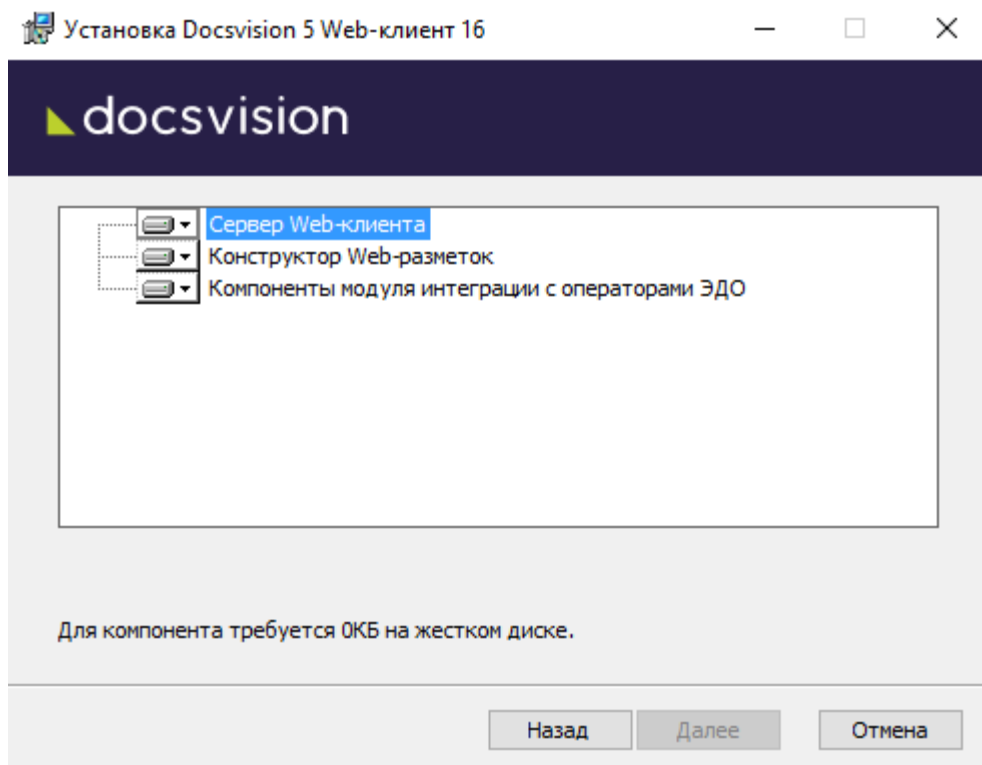


Рисунок 7. Установите "Компоненты модуля интеграции с операторами ЭДО"

Компоненты модуля интеграции с операторами ЭДО доступны для выбора только если ранее был установлен модуль *Модуль интеграции с операторами ЭДО*, поддерживающий работу в Web-клиент. Если порядок установки модулей был нарушен, компоненты можно добавить, повторно запустив инсталлятор и выбрав команду *Изменить*.

Если Web-клиент установлен на отдельной машине, для работы с модулем *Модуль интеграции с операторами ЭДО* через web-интерфейс, на машину с Web-клиентом потребуется установить серверную часть:

- [Коннектора к Диадок](#)
- [Модуля интеграции с операторами ЭДО](#)

Удаление модуля

1. Остановите и удалите из Docsvision бизнес-процессы БП *Отправка сообщений ЮЗДО* и БП *Получение сообщений ЮЗДО* подробнее см. [Сведения о структуре модуля](#).
2. Удалите:
 - Клиентский компонент модуля: *Модуль интеграции с операторами ЭДО. Клиент*.

- Серверный компонент модуля: *Модуль интеграции с операторами ЭДО*.
Сервер

При удалении серверной части модуля будет предложено выбрать БД Docsvision, из которой будут удалены компоненты модуля.

Обновление модуля

Список изменений в текущей версии см. здесь: [Изменения в релизной версии](#).

Условия обновления

- Установка новой версии должна выполняться в нерабочее время.
- Создайте резервную копию БД Docsvision.
- База данных Docsvision должна быть переведена на работу с расширенными метаданными.
- При обновлении версия клиентского дистрибутива должна быть равной версии серверного дистрибутива.
- Если к новой версии модуля приложена инструкция по обновлению, следуйте ей.
- Если комплект обновления включает обновление для модуля Платформа, перейдите к инструкции, приведенной в [undefined/platform/5.5.5/admin/update-module/\[документации\]](#) модуль Платформа. Руководство администратора, пункт "Установка новой версии модуля".

Порядок обновления



При выполнении данной операции в БД Docsvision будут загружены стандартные настройки Модуля интеграции с операторами ЭДО. Рекомендуем перед обновлением БД экспортировать *Справочник настроек операторов ЮЗДО* для последующего его импорта после обновления БД.

Чтобы установить новую версию модуля:

1. Создайте резервную копию БД.
2. Остановите все сервисы Docsvision. Также остановите экземпляры Службы фоновых операций и все сервисы в кластере Docsvision или СУБП, если таковые используются.

3. Установите новую версию модуля из пакета установки, не удаляя существующей версии.

При обновлении используется каталог и область установки текущей версии.

4. На последней странице мастера установки установите флаг **Обновить базу данных**. Запустите остановленные сервисы Docsvision, затем нажмите **Готово**.

После нажатия кнопки **Готово** будет запущена Консоль настройки Docsvision в режиме обновления БД, подробнее про режимы обновления БД см. в разделе "undefined/system/dv5/db-update/[Режимы обновления баз данных]". Если работа инсталлятора была прервана до полного завершения, БД не будет обновлена, Консоль настройки Docsvision не будет запущена.

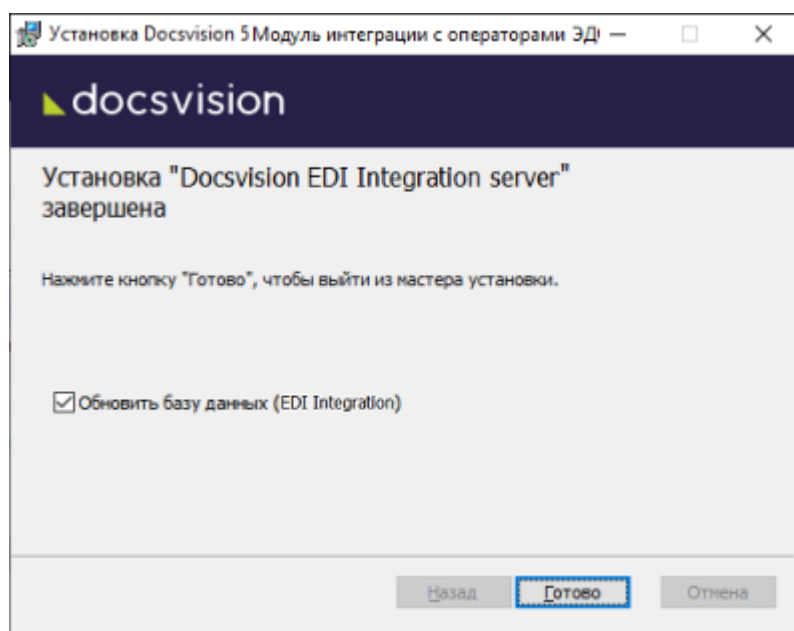


Рисунок 8. Флаг обновления базы данных

5. Выберите базы данных, в которых будут обновлены библиотеки карточек модуля, и нажмите **ОК**.

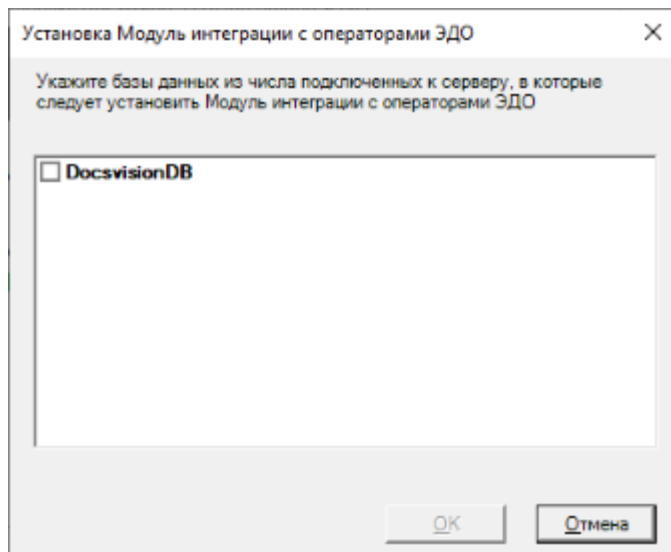


Рисунок 9. Список баз данных для обновления

Будет запущен процесс обновления БД. По окончании обновления появится окно с результатом обновления, а также автоматически будут перезапущены сервисы Docsvision.

6. Запустите "Консоль настройки Docsvision" и завершите настройку Docsvision стандартным образом согласно документации модуля Платформа [undefined/platform/5.5.5/admin/config-master/\[здесь\]](#) и [undefined/platform/5.5.5/admin/post-config-server/\[здесь\]](#).

Пользователь, от имени которого запускается "Консоль настройки Docsvision" должен являться администратором Docsvision, а также входить в группы Docsvision в Справочнике сотрудников: *Администраторы УД, Администраторы КС и Системная для WF*.

7. Установите новую версию модуля, не обновляя БД (флаг **Обновить базу данных** должен быть снят):
 - На оставшихся узлах кластера Docsvision.
 - На сервере СУБП, Web-клиента и других модулей, использующих компоненты модуля *Базовые объекты*.
8. Клиентская часть модуля обновится автоматически при открытии Windows-клиента. Самостоятельно обновить клиентские компоненты можно из пакета установки **Docsvision 5 EDIIntegrationClient.msi**.

Обновление нескольких модулей

Если планируется обновлять несколько модулей, следующие действия следует выполнять один раз после установки новых версий всех модулей:

1. Откройте Консоль настройки Docsvision.

Программа будет запущена в режиме мастера настройки.



Пользователь, от имени которого запускается *Консоль настройки Docsvision* должен являться администратором Docsvision, а также входить в группы Docsvision в Справочнике сотрудников: *Администраторы УД, Администраторы КС и Системная для WF.*

2. На странице *Базы данных* выберите рабочую БД Docsvision, установите переключатель в режим **Использовать выбранную в списке базу данных** и нажмите **Далее**.

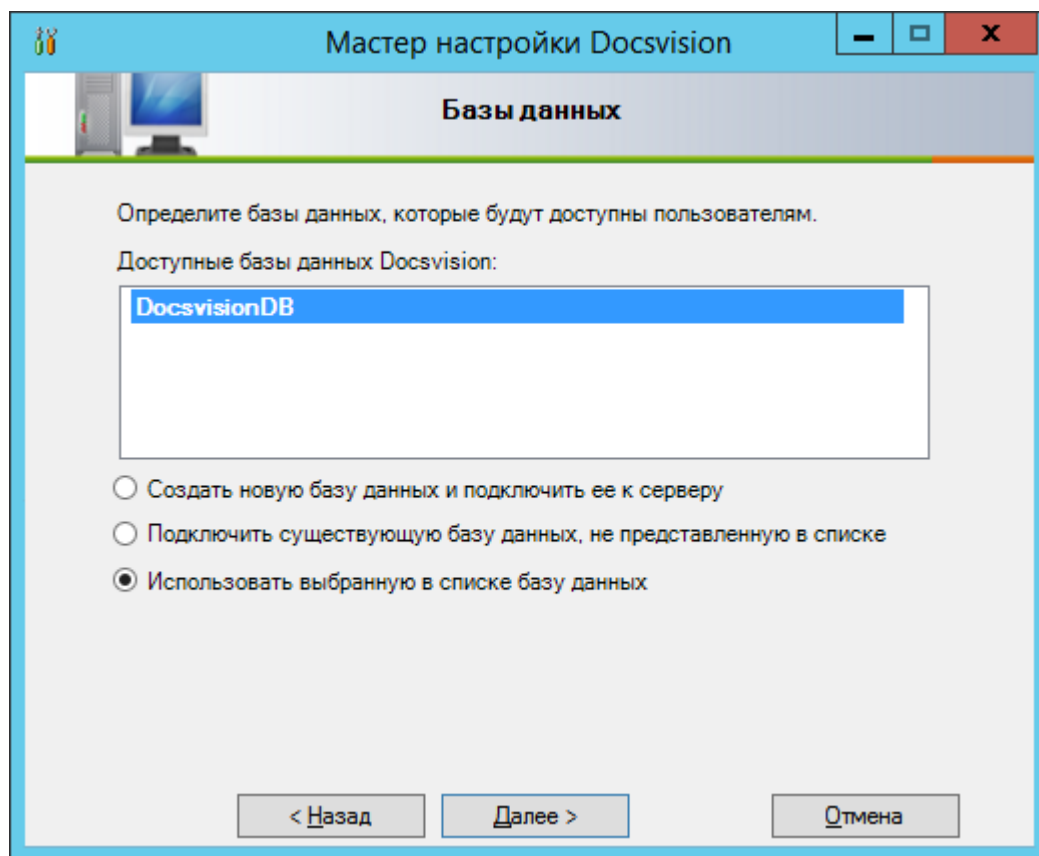


Рисунок 10. Обновление базы данных

3. Будет предложено обновить базу данных до новой версии. Выберите вариант **ДА**.
4. На следующей странице не изменяйте выбор обновляемых библиотек карточек. Нажмите **Далее**.

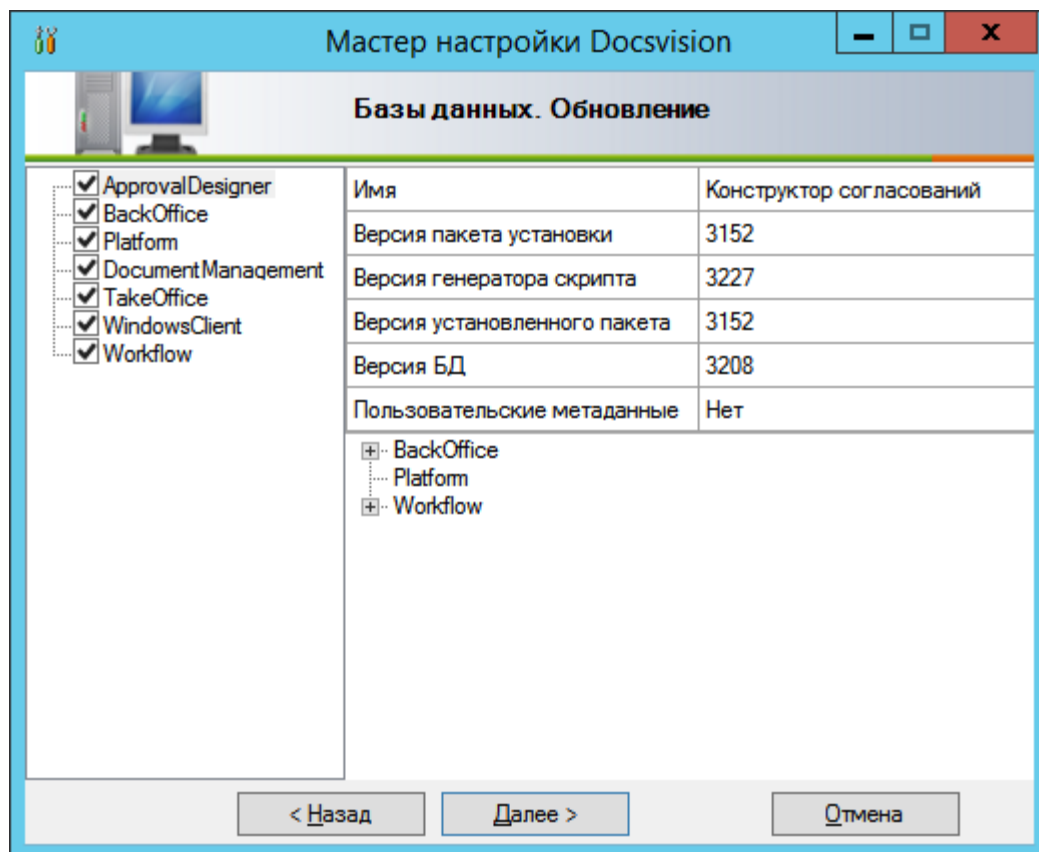


Рисунок 11. Обновление библиотек карточек

5. На странице *Параметры базы данных* нажмите **Далее**. Появится запрос на подтверждение обновления — согласитесь. Процесс обновления займет некоторое время.
6. На странице настроек производительности и перезапуска IIS нажмите **Далее**.
7. На странице *Загрузка специальной конфигурации Управление документами* нажмите **Далее**, чтобы загрузить стандартную конфигурацию приложения Управление документами.

Обратите внимание:

На шаге загрузки специальной конфигурации модулей примените или пропустите загрузку стандартных настроек. Если требуется сохранить существующие настройки, на шаге загрузки специальной конфигурации нажмите **Пропустить**.

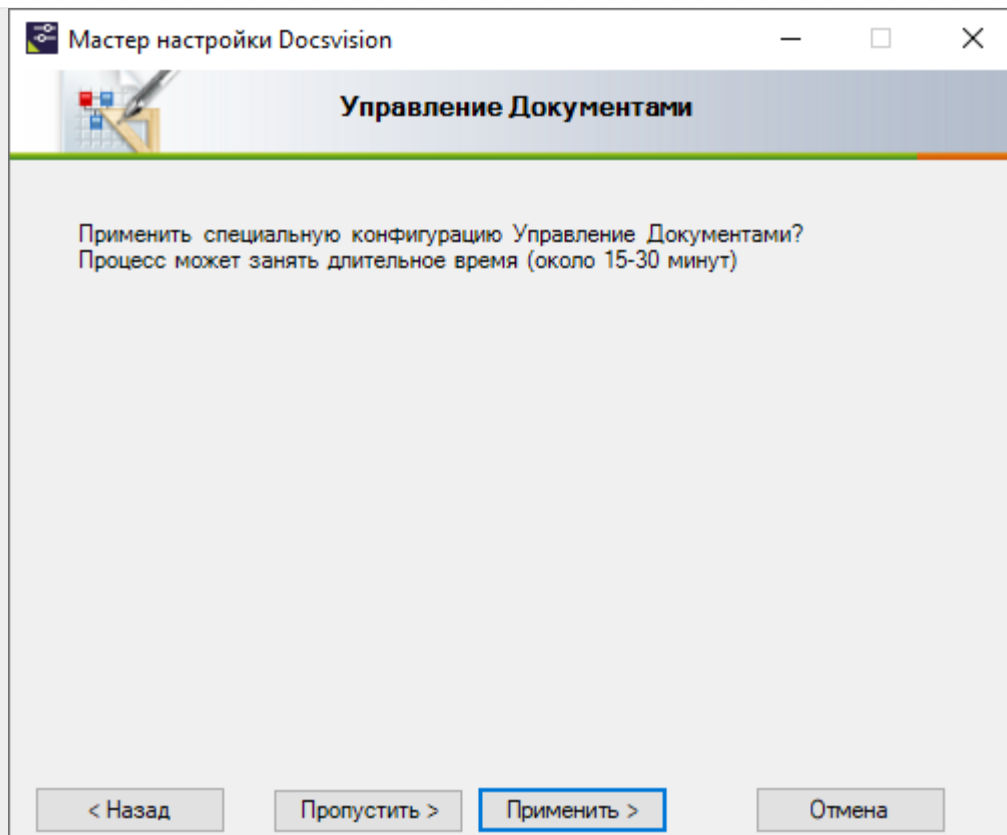


Рисунок 12. Специальная конфигурация "Управление документами"

8. Аналогичный выбор будет предложено представлен для специальной конфигурации модулей *Конструктор согласований*.
9. Обновите версию модуля на всех узлах кластера Docsvision, СУБП и Web-клиента.
10. Клиентские компоненты модуля на компьютерах пользователей будут обновлены автоматически при запуске Windows-клиента.

Самостоятельно обновить клиентские компоненты можно из пакета установки *Docsvision 5 EDIIntegrationClient.msi*.

Функции администратора

Модуль интеграции с операторами ЭДО

Функции администратора

Описание настроек модуля, выполняемых администратором Docsvision.

- [Настройка модуля в Справочнике настроек операторов ЮЗДО](#)
- [Настройка ролевой модели для группы входящих документов](#)
- [Настройка функций обмена неформализованными документами](#)
- [Больше подробностей в навигационном меню слева...](#)

Настройка модуля в Справочнике настроек операторов ЮЗДО

Справочник настроек операторов ЮЗДО предназначен для настройки подключения к операторам ЭДО, настройки параметров отправки и получения электронных документов, сопоставления организации и контрагентов в системе оператора ЭДО и системе Docsvision.

На следующем рисунке представлено основное окно Справочника настроек операторов ЮЗДО.

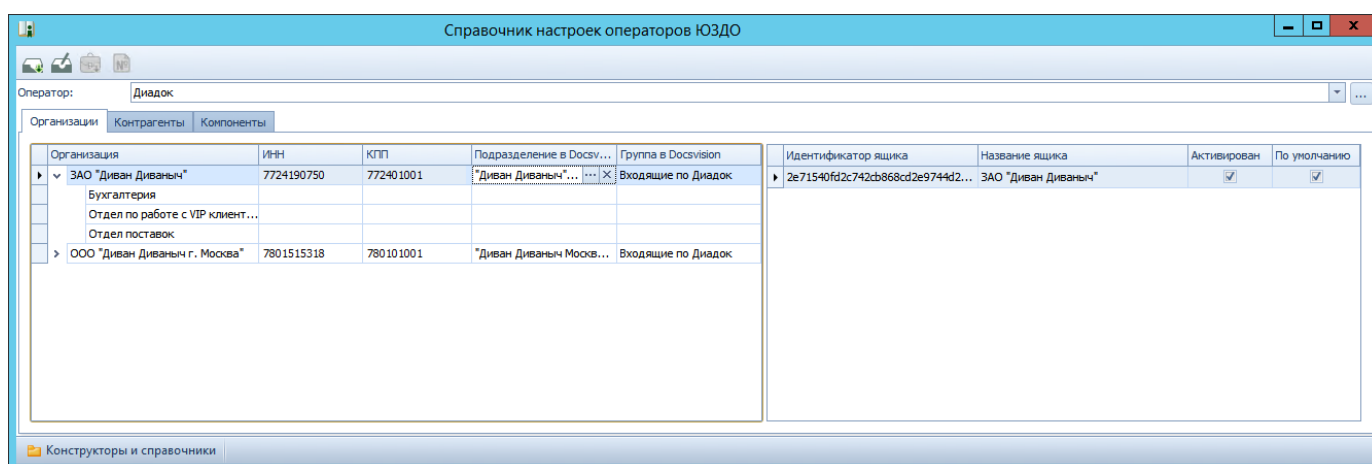


Рисунок 13. Окно Справочника настроек операторов ЮЗДО

Справочник содержит следующие элементы управления:

1. Панель инструментов с элементами:

-  — загружает из системы оператора ЭДО список организаций и их ящики электронного обмена.



Ящик (электронного обмена) — единица маршрутизации в системе оператора ЭДО, через которую осуществляется обмен электронными документами.

Ящики могут быть двух типов: собственный ящик организации, и ящик контрагента, которому отправляется или от которого принимается электронный документ.

-  — открывает окно редактирования выбранного ящика для активации и выбора ящика по умолчанию.
-  — загружает из системы оператора ЭДО список контрагентов и их ящики.
-  — загружает из системы оператора ЭДО одного контрагента.

2. Поле выбора оператора ЭДО, для которого выполняется настройка^[2].

- Кнопка  — открывает окно настройки подключения к оператору ЭДО.

3. Вкладки:

- *Организации* — содержит список организаций и их ящиков электронного обмена, загруженных из системы оператора ЭДО. На данной странице настраиваются связи организаций в системе оператора ЭДО с организациями в системе Docsvision.
- *Контрагенты* — содержит список контрагентов организации и их ящиков электронного обмена, загруженных из системы оператора ЭДО. На данной странице настраиваются связи контрагентов в системе оператора ЭДО с контрагентами в системе Docsvision.
- *Компоненты* — содержит элементы выбора компонентов отправки, получения и обработки электронных сообщений. На данной странице выбираются программные компоненты, которые будут использоваться при формировании, отправке, получении и обработке полученных электронных сообщений.

Открытие Справочника настроек операторов ЮЗДО

1. Запустите Windows-клиент от имени администратора Docsvision.
2. Откройте **Конструкторы и справочники** > **Справочник настроек операторов ЮЗДО**.

Настройка подключения к оператору ЭДО

Чтобы получить возможность работать с оператором ЭДО:

1. Установить коннектор к данному оператору ЭДО.
2. Настроить подключение к оператору в Справочнике настроек операторов ЮЗДО.

Порядок настройки подключения определяется требованиями коннектора к данному оператору ЭДО. См. подробности в документации к *коннектору* к

оператору ЭДО (например, "Коннектор к Диадок").

Настройка справочника организаций

Загрузка справочника организаций из системы оператора ЭДО

Администратору нужно загрузить справочник организаций, зарегистрированных у оператора ЭДО, в *Справочник настроек операторов ЮЗДО*.

Загрузку нужно выполнить при настройке модуля. Загрузку также можно выполнить в процессе эксплуатации модуля с целью актуализации справочника.

- 1. Перейдите на страницу *Организации* в *Справочнике настроек операторов ЮЗДО*.
- 2. Нажмите кнопку  на панели инструментов справочника.

Если список организаций был загружен ранее, будет предложено подтвердить обновление списка — актуализировать справочник.

Будет загружен список организаций с подразделениями и ящиками (в системе оператора ЭДО).

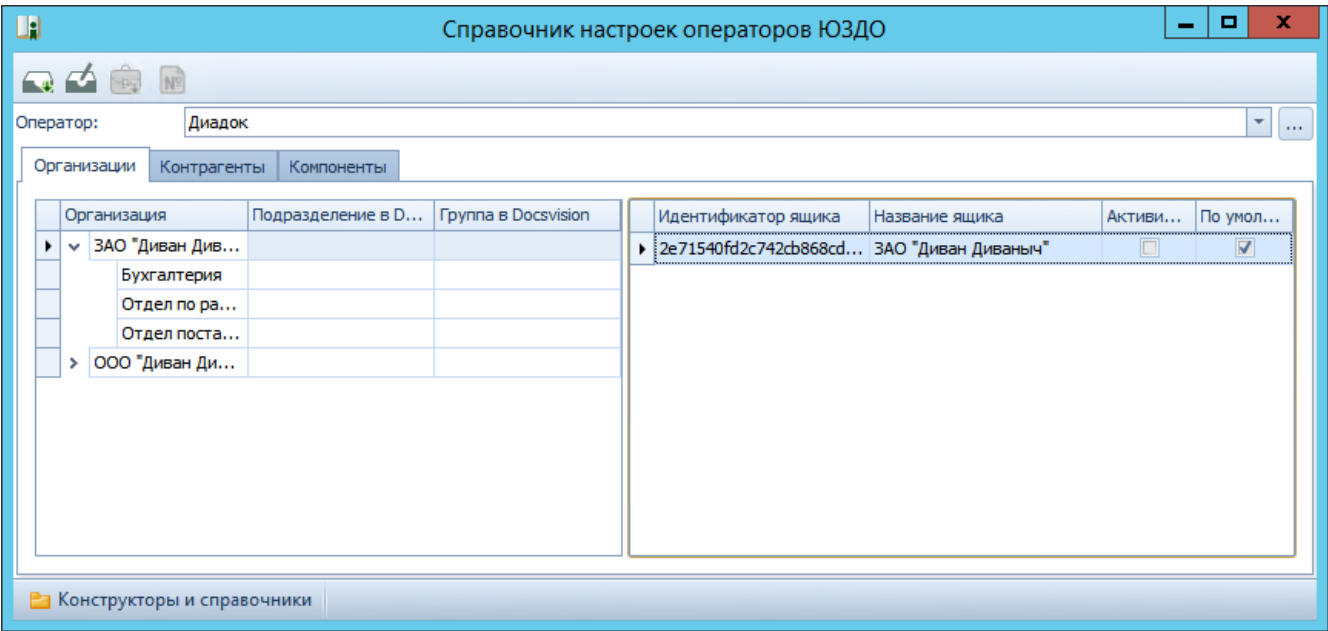


Рисунок 14. Список организаций, загруженных из системы оператора ЭДО

В левой части окна приведен список организаций, загруженных из системы оператора обмена, в правой — список ящиков выделенной организации.

Сопоставление организаций в системе оператора ЭДО с организациями в системе Docsvision

Администратору нужно сопоставить организации, загруженные в *Справочник*

настроек операторов ЮЗДО, с данными организациями в Справочнике сотрудников Docsvision. Для организаций, у которых ИНН/КПП в системе оператора ЭДО и в системе Docsvision совпадают, сопоставление будет выполнено автоматически.



Под сопоставлением организаций подразумевается выбор для каждой организации в системе оператора ЭДО её эквивалента в системе Docsvision.



Сопоставление должно быть выполнено для всех загруженных организаций и их подразделений.

Чтобы сопоставить организации или подразделения:

1. Перейдите на страницу *Организации* в *Справочнике настроек операторов ЮЗДО*.
2. Щелкните по полю *Подразделение в Docsvision* в строке сопоставляемой организации или подразделения.
3. Выберите организацию из *Справочника сотрудников*.

Выбор группы получателей входящих документов

Администратор должен выбрать группу сотрудников, которым будут передаваться поступившие от оператора ЭДО документы.

1. Перейдите на страницу *Организации* в *Справочнике настроек операторов ЮЗДО*.
2. Нажмите в поле *Группа в Docsvision* в строке организации, для документов которой указываются получатели.
3. Выберите группу сотрудников *Получатели документов* из *Справочника сотрудников*.

Группа *Получатели документов* является группой по умолчанию для добавления сотрудников, ответственных за обработку поступающих документов. Если в списке *Организации* несколько организаций, для каждой может быть назначена собственная группа ответственных сотрудников — администратору нужно создать группу и включить туда ответственных сотрудников.

При создании входящего документа его получателями назначаются сотрудники выбранной группы, а не сама группа.

Назначение ящика электронного обмена по умолчанию

Если у организации в системе оператора ЭДО зарегистрировано несколько электронных ящиков, администратор должен выбрать ящик, с которого из системы Docsvision по умолчанию будут отправляться электронные документы.

1. Перейдите на страницу *Организации* в *Справочнике настроек операторов ЮЗДО*.
2. Выберите организацию, настройка которой выполняется.
3. Выберите ящик, который назначается ящиком по умолчанию, и нажмите кнопку  на панели инструментов. Будет открыто окно настройки ящика.

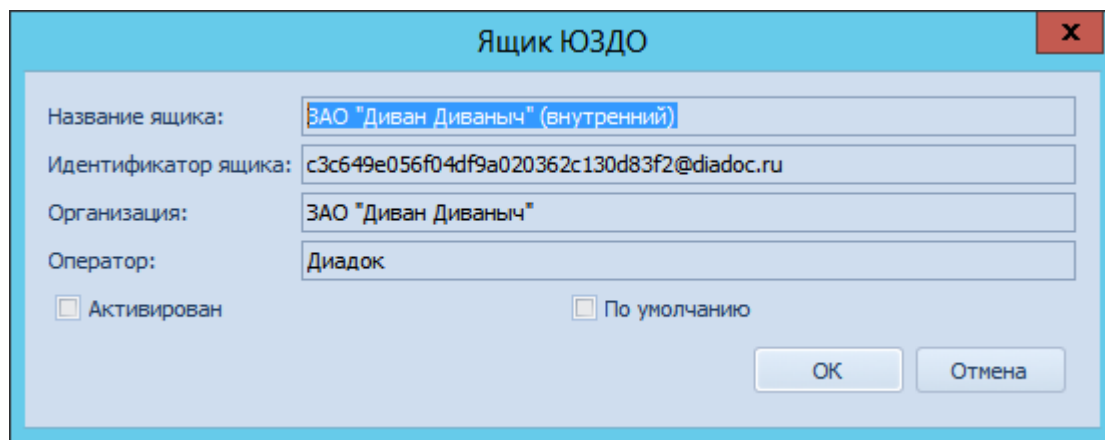
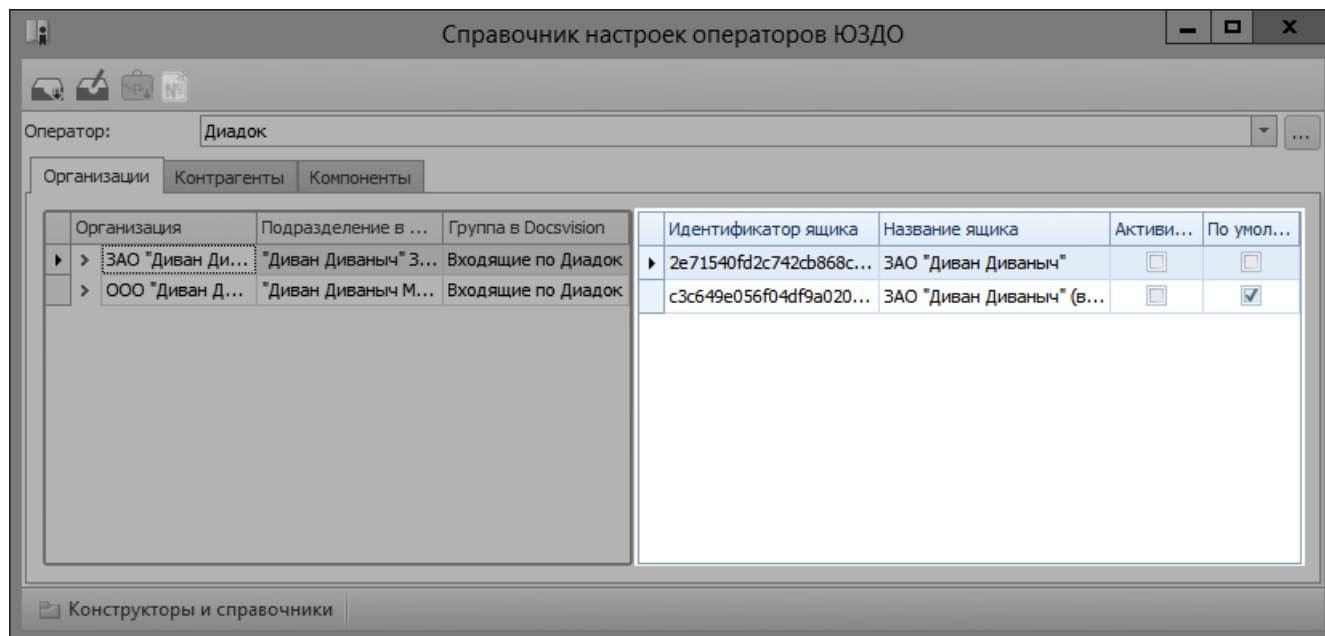


Рисунок 15. Настройка ящика

4. Установите флаг **По умолчанию** и нажмите кнопку **ОК**. В списке ящиков у данного ящика будет установлен флаг **По умолчанию**.



Организация	Подразделение в ...	Группа в Docsvision	Идентификатор ящика	Название ящика	Активи...	По умол...
ЗАО "Диван Ди..."	"Диван Диваныч" З...	Входящие по Диадок	2e71540fd2c742cb868c...	ЗАО "Диван Диваныч"	☐	☐
ООО "Диван Д..."	"Диван Диваныч М..."	Входящие по Диадок	c3c649e056f04df9a020...	ЗАО "Диван Диваныч" (в...	☐	☒

Рисунок 16. Ящики организаций

Активация ящика организации

Администратор должен активировать ящики организаций, из которых требуется загружать электронные сообщения: входящие документы, ответы. Сообщения из неактивированных ящиков не загружаются.

1. Перейдите на страницу *Организации* в *Справочнике настроек операторов ЮЗДО*.
2. Выберите организацию, настройка которой выполняется.
3. Выберите активируемый ящик и нажмите кнопку  на панели инструментов. Будет открыто окно настройки ящика.

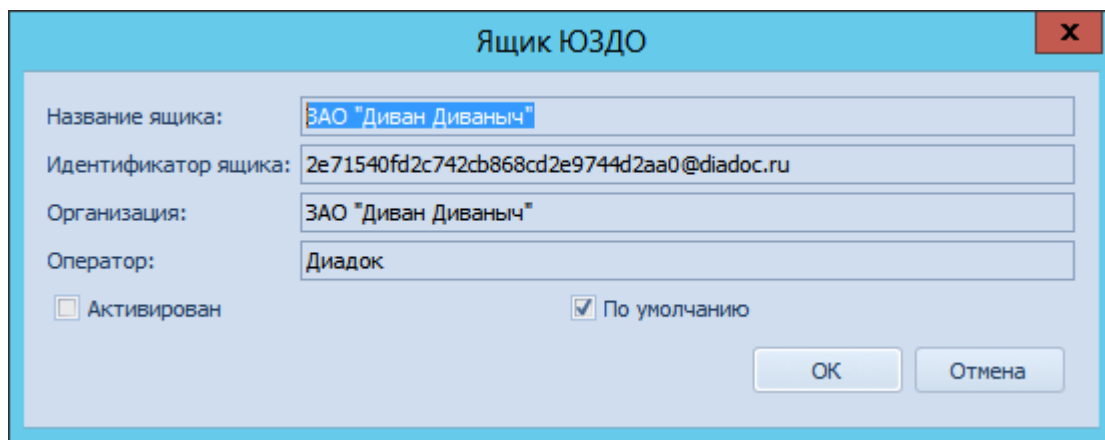


Рисунок 17. Настройка ящика

4. Установите флаг **Активирован** и нажмите кнопку **ОК**. Будет предложено выбрать вариант загрузки событий (электронных сообщений) из ящика.

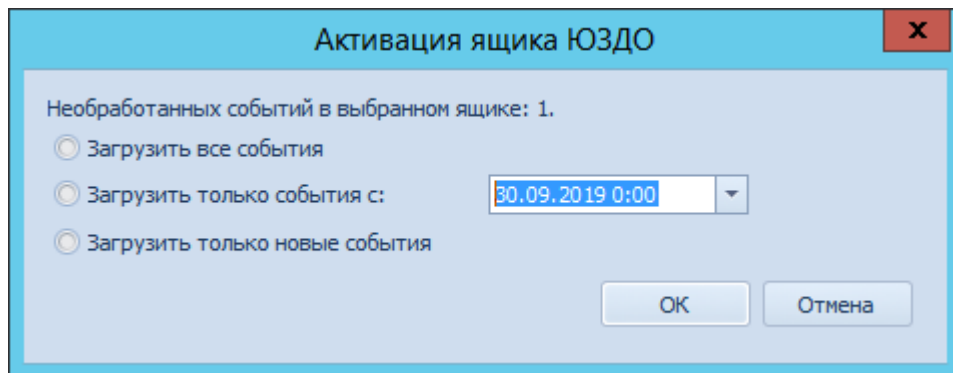


Рисунок 18. Настройка активации ящика

Варианты загрузки событий:

- **Загрузить все события** — будут загружены все имеющиеся в ящике сообщения.
- **Загрузить только события с** — будут загружены сообщения, появившиеся начиная с указанной даты.

- **Загрузить только новые события** — будут загружены сообщения, появившиеся начиная с текущего момента времени.



Сообщения будут загружены (в соответствии с выбранным вариантом загрузки) при запуске БП *Получение сообщений ЮЗДО*.

5. Выберите вариант загрузки событий и нажмите кнопку **ОК**. В списке ящиков у данного ящика будет установлен флаг **Активирован**.

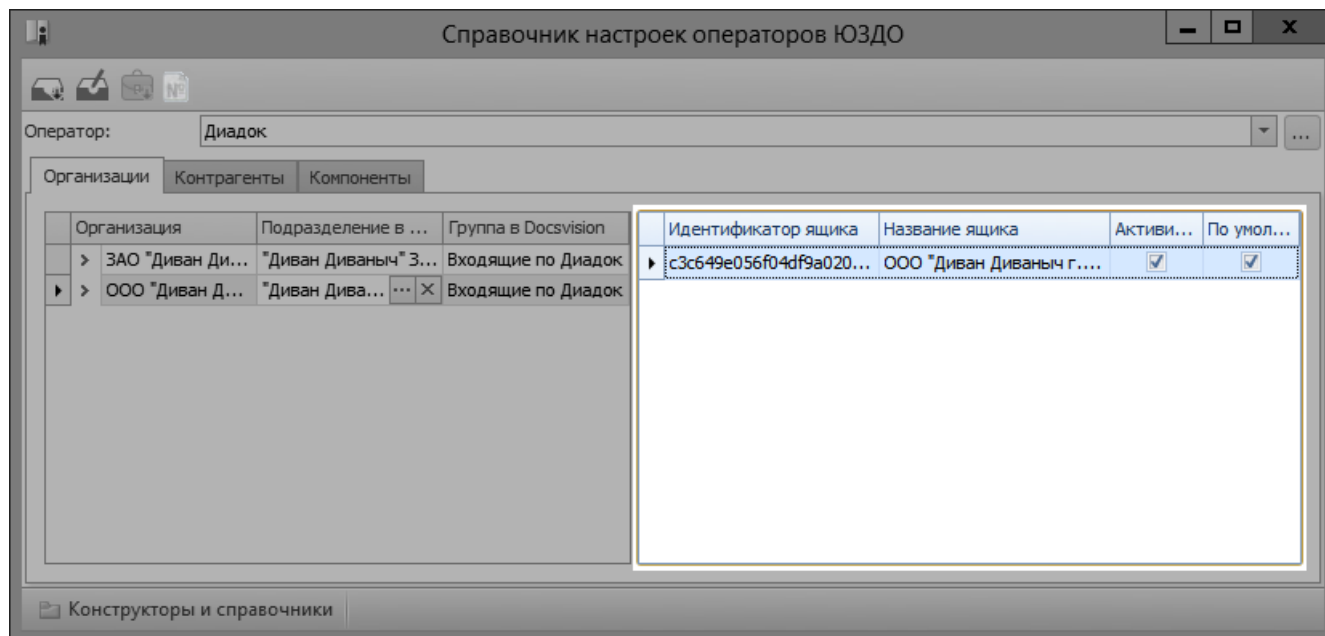


Рисунок 19. Ящики организаций

Настройка справочника контрагентов

Загрузка справочника контрагентов из системы оператора ЭДО

Администратору нужно загрузить справочник контрагентов, зарегистрированных у оператора ЭДО, в *Справочник настроек операторов ЮЗДО*.

Загрузку нужно выполнить при настройке модуля. Загрузку также можно выполнить в процессе эксплуатации модуля с целью актуализации справочника.

1. Перейдите на страницу *Контрагенты* в *Справочнике настроек операторов ЮЗДО*.
2. Нажмите кнопку  на панели инструментов справочника.

Если список контрагентов был загружен ранее, будет предложено подтвердить обновление списка — актуализировать справочник.

Будет загружен список контрагентов с подразделениями и ящиками (в системе оператора ЭДО).

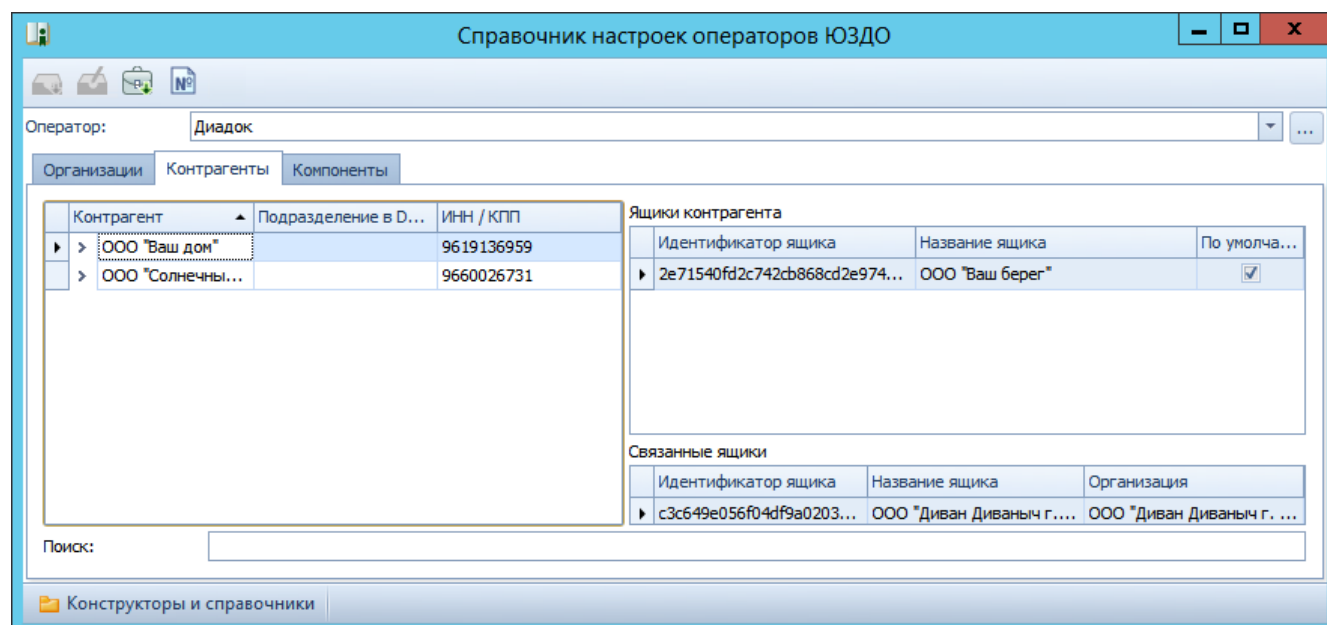


Рисунок 20. Список контрагентов, загруженных из системы оператора ЭДО

В левой части окна приведен список контрагентов, загруженных из системы оператора обмена, в правой — список ящиков выделенного контрагента и список ящиков организации (связанные ящики), с которых данному контрагенту разрешено отправлять электронные документы.

Сопоставление контрагентов в системе оператора ЭДО с контрагентами в системе Docsvision

Администратору нужно сопоставить подразделения, загруженные в *Справочник настроек операторов ЮЗДО*, с данными организациями в *Справочнике контрагентов Docsvision*. Для организаций, у которых ИНН/КПП в системе оператора ЭДО и в системе Docsvision совпадают, сопоставление будет выполнено автоматически.



Сопоставление должно быть выполнено для всех загруженных организаций и их подразделений.

Чтобы сопоставить контрагентов и их подразделения:

1. Перейдите на страницу *Контрагенты* в *Справочнике настроек операторов ЮЗДО*.
2. Нажмите в поле *Подразделение в Docsvision* в строке сопоставляемого контрагента или его подразделения.
3. Выберите организацию из *Справочника контрагентов*.

Загрузка одного контрагента из системы оператора ЭДО

Кнопка  позволяет загрузить из системы оператора ЭДО только одного контрагента.

При нажатии на кнопку появляется окно выбора контрагента из справочника контрагентов. ИНН и КПП будут подставлены автоматически после выбора записи. Выбранная организация при успешной синхронизации всегда автоматически подставляется в столбец *Подразделение в Docsvision*.

После нажатия на кнопку **Загрузить** происходит загрузка из системы оператора ЭДО контрагента и его ящика. По итогу появится сообщение об успешной или неудачной загрузке.

Если в реестре указать ключ `NoServerExtension` с типом `DWORD 1` для всего справочника, он будет работать напрямую с оператором ЭДО. Ключ может быть указан в ветке `HKEY_LOCAL_MACHINE\SOFTWARE\DocsVision\Edi\5.5` или той же в `HKEY_CURRENT_USER`.

Назначение ящика электронного обмена по умолчанию

Если у контрагента в системе оператора ЭДО зарегистрировано несколько электронных ящиков, администратор должен выбрать ящик, на который по умолчанию из системы Docsvision будут отправляться электронные документы.

1. Перейдите на страницу *Контрагенты* в *Справочнике настроек операторов ЮЗДО*.
2. Выберите контрагента, настройка которого выполняется.
3. Установите флаг **По умолчанию** в строке ящика контрагента.

Настройка отправки электронных документов

Администратор может добавить новые виды (типы) карточек, из которых возможна отправка электронных документов, а также изменить текущие настройки отправки.

Чтобы добавить настройки отправки нового вида карточек:

1. Перейдите на страницу *Компоненты* в *Справочнике настроек операторов ЮЗДО*.
2. В секции *Отправка* нажмите кнопку .

Будет предложено выбрать вид или тип карточек, для которого добавляется настройка.

3. Выберите вид/тип карточек и нажмите кнопку **ОК**.

В список настроек будет добавлена новая запись с пустыми полями *Имя класса компонента чтения* и *Имя класса компонента изменения*, и выбранным типом/видом карточек в поле *Тип карточки*.

Отправка:

 		
Тип карточки	Имя класса компонента чтения	Имя класса компонента изменения
Документ	DocsVision.Edi.Runtime.BackOffice.DocumentDataRead...	DocsVision.Edi.Runtime.BackOffice.DocumentUpdater, D...
Исходящий УПД	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice...	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice...
Исправление УПД	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice...	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice...
Исходящий УКД	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice...	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice...
Входящий УПД	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice...	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice...
Входящий УКД	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice...	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice...
▶ Рабочий	...	

Рисунок 21. Настройки отправки электронных документов

4. Укажите компоненты чтения и изменения данных карточки.

- а. В поле *Имя класса компонента чтения* выберите файл DLL, предоставляющий функции чтения данных карточки выбранного типа/вида.

Файл DLL должен содержать класс, реализующий программный интерфейс **IDocumentDataReader**.

Для неформализованных документов можно использовать стандартную реализацию компонента — файл **DocsVision.Edi.Runtime.BackOffice.dll** (только для видов карточки *Документ*). Для формализованных документов потребуется собственная реализация.

- При отправке неформализованных документов данные о контрагенте для отправки берутся из секции *Получатели — контрагенты*, *Получатель — организация контрагента* — поле в этой секции.

- б. В поле *Имя класса компонента изменения* выберите файл DLL, предоставляющий функции изменения данных карточки выбранного типа/вида.

Файл DLL должен содержать класс, реализующий программный интерфейс **IDocumentUpdater**.

Для неформализованных документов можно использовать стандартную реализацию компонента — файл **DocsVision.Edi.Runtime.BackOffice.dll** (только для видов карточки *Документ*). Для формализованных документов

потребуется собственная реализация.



Инструкция по разработке компонентов чтения и изменения данных карточки приведена в руководстве программиста "Модуля интеграции с операторами ЭДО".

Ограничение выбора организации

Выбор организаций в диалоге отправки документов можно ограничить с помощью скрипта (раньше можно было ограничить только выбор подразделения).

Чтобы ограничить выбор организации, используйте свойство:

```
public OperatorsUnit FixedUnit { get; set; }
```

Свойство добавлено во все формы **Send*Form** и **SignAndSend*Form**.

Свойство можно установить перед инициализацией в нужную организацию требуемого оператора, тогда отправка будет строго от нее.

Настройка получения электронных документов

Администратор может добавить новые типы входящих электронных документов (формализованных или неформализованных), которые могут быть получены с помощью модуля.

Чтобы добавить настройки получения нового типа документов:

1. Перейдите на страницу *Компоненты* в *Справочнике настроек операторов ЮЗДО*.
2. В секции *Получение* нажмите кнопку . В список настроек будет добавлена новая запись с пустыми полями *Тип документа*, *Версия*, *Вид карточки* и *Имя класса компонента создания*.

Получение:

Тип документа	Версия	Вид карточки	Имя класса компонента создания
AcceptanceCertificate		Входящий	DocsVision.Edi.Runtime.BackOffice.DocumentCreator, DocsVision.Edi.Runtime.BackOffice, ...
AcceptanceCertificate		Входящий	DocsVision.Edi.Runtime.BackOffice.DocumentCreator, DocsVision.Edi.Runtime.BackOffice, ...
CertificateRegistry		Входящий	DocsVision.Edi.Runtime.BackOffice.DocumentCreator, DocsVision.Edi.Runtime.BackOffice, ...
CertificateRegistry		Входящий	DocsVision.Edi.Runtime.BackOffice.DocumentCreator, DocsVision.Edi.Runtime.BackOffice, ...
Contract		Входящий	DocsVision.Edi.Runtime.BackOffice.DocumentCreator, DocsVision.Edi.Runtime.BackOffice, ...
Invoice		Входящий УПД	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoiceCreator, DocsVision.Edi.Runtime.U...
Invoice	utd820_05_01_01	Входящий УПД (820)	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice820Creator, DocsVision.Edi.Runtim...
Invoice	utd820_05_01_01 hvobn	Входящий УПД (820)	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice820Creator, DocsVision.Edi.Runtim...

Рисунок 22. Настройки получения электронных документов

3. В поле *Тип документа* введите название типа документа (формализованного или неформализованного) в системе оператора ЭДО.



Список типов документов оператора Диадок приведен на странице [по ссылке](#).

4. В поле *Версия* для получаемых формализованных документов, сформированных в формате **820 приказа** (для Диадок) нужно указывать версию формата: **utd820_05_01_01** или **utd820_05_01_01_hyphen**. Рекомендуется дублировать запись, указав оба формата как на рисунке выше.
5. В поле *Вид карточки* выберите виды карточек, которые будут создаваться в Docsvision при поступлении входящего документа.
6. В поле *Имя класса компонента создания* выберите файл DLL, предоставляющий функции создания карточек выбранного вида из входящих электронных документов. Файл DLL должен содержать класс, реализующий программный интерфейс **IDocumentCreator**.

Для неформализованных документов можно использовать стандартную реализацию компонента — файл **DocsVision.Edi.Runtime.BackOffice.dll** (только для видов карточки *Документ*). Для формализованных документов потребуется собственная реализация.



Инструкция по разработке компонента создания карточек приведена в [undefined/programmer/dv5/Edi/\[руководстве разработчика модуля\]](#).

Настройка загрузки документов из оператора ЭДО

Для оператора ЭДО Диадок доступна возможность загружать документы для хранения в архиве Docsvision.

Загрузка документов из оператора ЭДО выполняется на странице *Загрузка* в *Справочнике настроек операторов ЮЗДО*. Для загрузки доступны исходящие документы и входящие документы.

Загрузка входящих документов

1. В справочнике настроек операторов ЭДО в таблице *Загрузка входящих* найдите колонку *Дополнительные атрибуты* и добавьте XSLT-шаблон:

- **CardPackage\CardDocument_AdditionalExample.xml** — шаблон документа с

дополнительными секциями и полями.

- `CardDefs\Xslt\UniInvoice.ru.xslt` — шаблон преобразования для УПД.
- `CardDefs\Xslt\UniInvoiceAct.ru.xslt` — шаблон преобразования для Формализованного акта.
- `CardDefs\Xslt\UniInvoiceCorrection.ru.xslt` — шаблон преобразования для УКД.
- `CardDefs\Xslt\UniInvoiceCorrFields.xslt` — шаблон преобразования для УКД.
- `CardDefs\Xslt\UniInvoiceFields.xslt` — шаблон преобразования для УПД.
- `CardDefs\Xslt\UniInvoiceSF.xslt` — шаблон преобразования для Счета-фактуры.
- `CardDefs\Xslt\UniInvoiceTorgRu.xslt` — шаблон преобразования для Товарной накладной ТОРГ-12.

Шаблоны построены по следующей логике:

```
<CardDocument> ①  
  
②  
<MainInfo> ③  
<xsl:attribute name="TransferNumber"> ④  
  
⑤  
<Goods> ⑥  
<GoodsRow> ⑦  
<xsl:attribute name="Order"> ⑧
```

- ① Тип карточки.
- ② Для плоских секций:
- ③ Псевдоним секции.
- ④ Поле.
- ⑤ Для коллекционных секций:
- ⑥ Псевдоним секции.
- ⑦ Новая строка секции.
- ⑧ Поле.

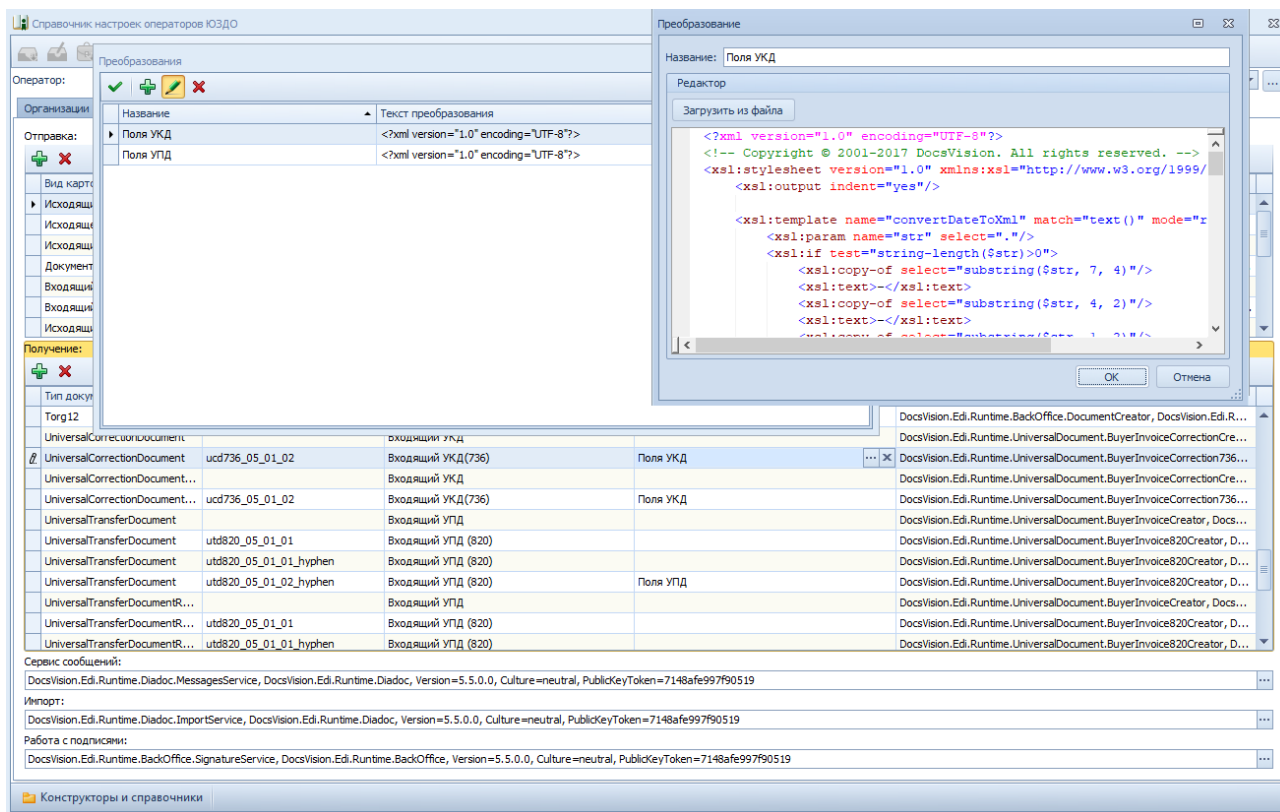


Рисунок 23. Пример проверки загрузки атрибутов

- В колонке укажите XSLT для загрузки дополнительных параметров. В поставку включён пример XSLT и описание формата XML, откуда загружаются данные в карточку.

Указанный шаблон XSLT будет использоваться при загрузке документооборота, а так же при обычной синхронизации. Если шаблон не задан, будет выполнена загрузка стандартных полей. Если шаблон задан, в дополнение к стандартным полям будут также загружены те, что указаны в преобразовании. Для стандартных видов документов XSLT не используется.

Загружаются все документы, попадающие в диапазон дат, указанный при [активации ящика организации](#). Для каждого загружаемого документа в системе создаётся карточка документа определённого вида. Место хранения в системе и вид документа указываются в *Справочнике настроек операторов ЮЗДО* в зависимости от атрибутов документа оператора ЭДО.

При попытке загрузить один и тот же документ с более поздним статусом, новая карточка не будет создана. Все файлы и атрибуты документа обновляются в уже существующей карточке Docsvision.

В карточку по каждому документу загружаются следующие файлы:

1. Файл документа — загружается как "основной".
2. Электронная подпись файла документа — после загрузки может выполняться проверка ЭП.

Загрузка исходящих документов

Загружаются все документы, попадающие в диапазон дат, указанный при [активации ящика организации](#).

Загружаются следующие виды документов:

i. Формализованные документы:

- УПД.
- Акт.
- Накладная.
- Счет-фактура.
- УКД.
- ИСФ.
- КСФ.
- Исправление к УПД.
- Другие аналоги корректировочных документов.

ii. Неформализованные документы:

- Акт.
- Накладная.
- Транспортная накладная.
- Счет.
- Акт сверки.
- Детализация.
- Доверенность.
- Письмо.
- Протокол согласования цены.
- Реестр сертификатов.
- Ценовой лист.

iii. Договорные документы.

iv. Приглашения ЭДО:

- Входящее приглашение ЭДО.
- Исходящее приглашение ЭДО.

Чтобы загрузить исходящие документы:

1. Выберите *Имя класса компонента создания*.

По умолчанию колонка не содержит настройки, необходимо самостоятельно добавить те виды, которые требуется загружать. В качестве примера можно использовать настройки для входящих документов.

По умолчанию это:

- `DocsVision.Edi.Runtime.BackOffice.OutgoingDocumentCreator` для неформализованных документов.
- `DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice820Creator` для УПД 820.
- `DocsVision.Edi.Runtime.UniversalDocument.SellerInvoiceCreator` для УПД старого формата.
- `DocsVision.Edi.Runtime.BackOffice.InvitationDataReader` для приглашений к обмену ЭДО.

При добавлении двух видов "Исходящее приглашение к обмену" и "Входящее приглашение к обмену" нужно в блоке *Отправка* прописать оба вида, а в блоке *Получение* только входящий.

2. В таблице *Загрузка исходящих* найдите колонку *Дополнительные атрибуты* и добавьте XSLT шаблон по аналогии с входящими документами. В папке инсталляции это шаблон `CardDefs\Xslt\UniInvoiceFields.xslt`.

3. Чтобы использовать существующие виды документов, а не делать новые сразу после загрузки, для состояния УПД `SignedAndSent` можно использовать стандартный шаблон.

Для собственных видов в шаблоне необходимо прописать следующий ID состояния `BuiltInStateId` = `"80F6D41E-379C-44EB-B858-8A9CB1CC15F5"` самостоятельно. Допускается, например, сделать вид документа с начальным состоянием `Загружен` и переходами в остальные состояния. В таком случае состояние `Подготавливается` будет отсутствовать.

Загрузку документов выполняет БП `CardPackage\LoadOutgoingDocuments.xml`. Для

входящих документов используется `CardPackage\LoadIncomingDocuments.xml`.
Процедуру загрузки можно вызвать кодом. Для входящих и исходящих приглашений ЭДО используется `CardPackage\ReceiveEdiInvitations.xml`, `CardPackage\SendEdiInvitations.xml` соответственно.

Настройка функций обмена приглашениями ЭДО



Работа с приглашениями по умолчанию производится только из Web-клиента. Для работы с приглашениями через Windows-клиент администратор должен самостоятельно настроить разметки.

Для работы с приглашениями к обмену предназначены карточки:

- Исходящее приглашение ЭДО.
- Входящее приглашение ЭДО.

Указанные виды карточек предоставляются с настроенными разметками, автоматом состояний и ролевой моделью.

Настройка папок для приглашений ЭДО

Поисковые запросы и представления для папок входящих приглашений ЭДО, если требуется, могут быть разработаны самостоятельно.

Настройка разметки вида "Входящее приглашение ЭДО"

Карточка входящего приглашения запрещена для создания пользователем.

1. В справочнике Операторов ЮЗДО на вкладке *Компоненты* в поле *Отправка* добавьте виды *Входящее приглашение ЭДО* и *Исходящее приглашение ЭДО*.
2. Укажите компонент чтения `InvitationDataReader`, компонент обновления `DocumentUpdater`.

Отправка:		
Вид карточки	Имя класса компонента чтения	Имя класса компонента изменения
Исходящий УПД (820)	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice820DataReader, DocsVision.Edi.Runti...	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice820Updater, DocsVision.Edi.Runtime.U...
Входящий УПД (820)	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice820DataReader, DocsVision.Edi.Runti...	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice820Updater, DocsVision.Edi.Runtime.U...
Доверенность (версия 002)	DocsVision.Edi.Runtime.BackOffice.PowerOfAttorneyDataReader, DocsVision.Edi.Runtime.Back...	DocsVision.Edi.Runtime.BackOffice.PowerOfAttorneyUpdater, DocsVision.Edi.Runtime.BackOffi...
Входящий УКД (736)	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoiceCorrection736DataReader, DocsVision...	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoiceCorrection736Updater, DocsVision.Edi...
Исходящий УКД (736)	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoiceCorrection736DataReader, DocsVision...	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoiceCorrection736Updater, DocsVision.Edi...
Исходящее исправление УПД (820)	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice820DataReader, DocsVision.Edi.Runti...	DocsVision.Edi.Runtime.UniversalDocument.SellerInvoice820Updater, DocsVision.Edi.Runtime.U...
Формализованный акт	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice820DataReader, DocsVision.Edi.Runtime...	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice820Updater, DocsVision.Edi.Runtime.Unive...
Формализованный акт (820)	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice820DataReader, DocsVision.Edi.Runti...	DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoice820Updater, DocsVision.Edi.Runtime.U...
Входящее приглашение ЭДО	DocsVision.Edi.Runtime.BackOffice.InvitationDataReader, DocsVision.Edi.Runtime.BackOffice, ...	DocsVision.Edi.Runtime.BackOffice.DocumentUpdater, DocsVision.Edi.Runtime.BackOffice, Versi...
Исходящее приглашение ЭДО	DocsVision.Edi.Runtime.BackOffice.InvitationDataReader, DocsVision.Edi.Runtime.BackOffice, ...	DocsVision.Edi.Runtime.BackOffice.DocumentUpdater, DocsVision.Edi.Runtime.BackOffice, Versi...

Рисунок 24. Виды "Входящее приглашение ЭДО" и "Исходящее приглашение ЭДО"

3. В разметку *Исходящего приглашения ЭДО* добавьте элемент управления *Подразделение Контрагента*.
4. В поле *Источник данных* укажите **Договор**, поле данных — **Организация контрагента**.

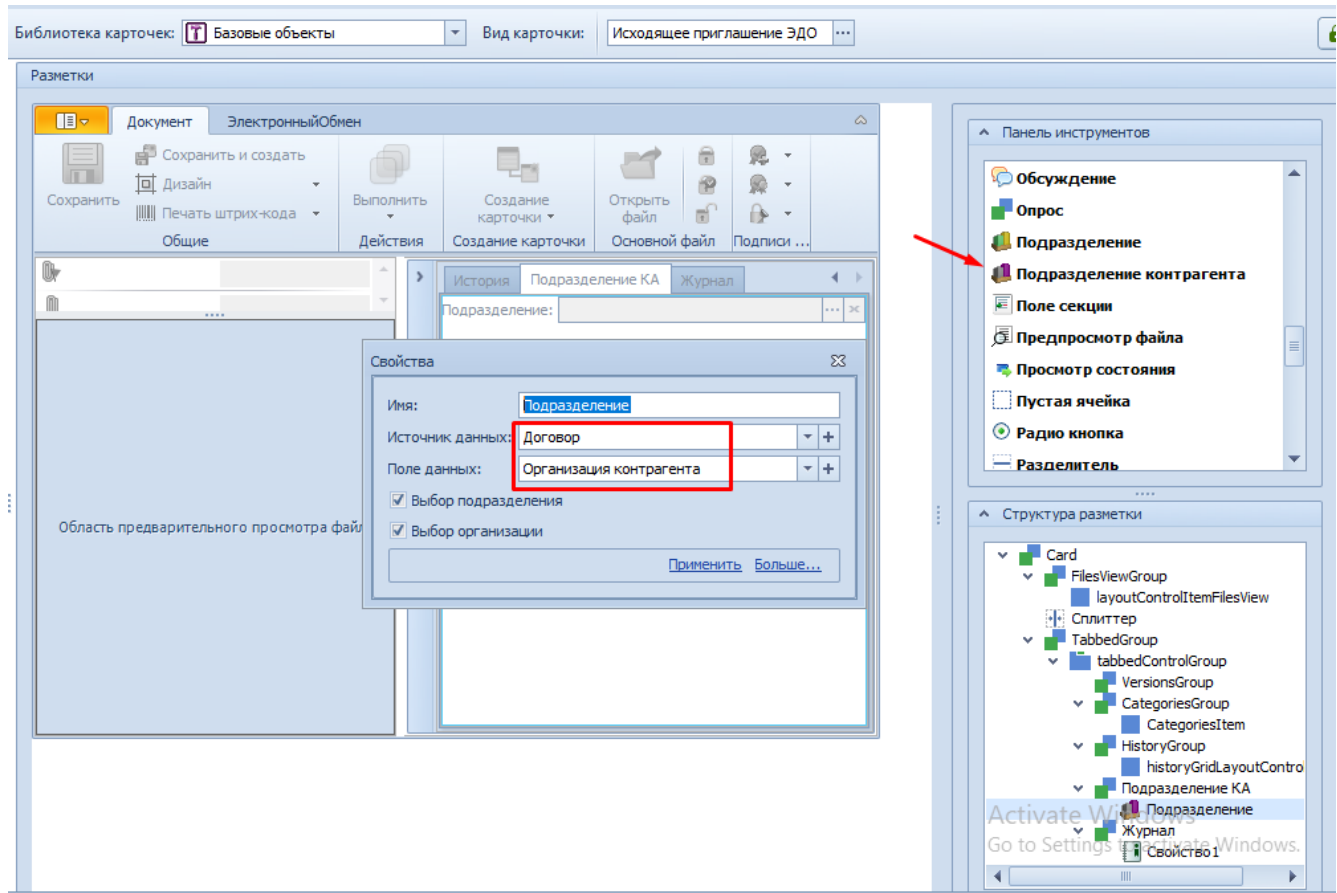


Рисунок 25. Указание источника данных

5. Откройте в *Конструкторе разметок* настройки вида *Входящее приглашение ЭДО*, *Исходящее приглашение ЭДО*.
6. Добавьте на ленту карточки новую группу с названием *Электронный обмен*.
7. Добавьте в группу *Электронный обмен* команды:
 - Для разметки *Исходящего приглашения*:
 - **Отправить** приглашение.
 - **Отозвать** приглашение.
 - Для разметки *Входящего приглашения*:
 - **Принять** приглашение.
 - **Отказать** в принятии приглашения.

8. Укажите операции доступа для добавленных команд.
9. Добавьте в скрипт родительского вида, *Обмен приглашениями*, ссылку на дополнительную сборку `DocsVision.Edi.DocumentScript.dll`.
10. В скрипты *Исходящего* и *Входящего* приглашений добавьте пространство:

```
using DocsVision.Edi.DocumentScript;
```

11. В раздел подключаемых пространств имен добавьте:

```
using DocsVision.Edi.EdiScriptHelpers;
```

12. Добавьте обработчики для новых команд:

Для команды Отправить приглашение:

```
EdiScriptHelper.SendInvitation();
```

Для команды Отозвать приглашение:

```
EdiScriptHelper.CancelInvitation();
```

Для команды Принять приглашение:

```
EdiScriptHelper.AcceptInvitation();
```

Для команды Отказать в принятии приглашения:

```
EdiScriptHelper.RejectInvitation();
```

13. Добавьте в разметку карточки элемент управления *Журнал работы с приглашениями*. Рекомендуется размещать элемент управления на новой вкладке (например, *Журнал работы с приглашениями ЮЗДО*). Настраивать элемент управления не требуется.
14. Загрузите БП для обмена приглашениями:
 - `CardPackage\ReceiveEdiInvitations.xml` — для получения входящих приглашений к обмену.
 - `CardPackage\SendEdiInvitations.xml` — для отправки исходящих приглашений к обмену.

Настройки web-разметок

Для работы с приглашениями ЭДО в Web-клиенте можно самостоятельно настроить web-разметки.

События и обработчики событий исходящих приглашений:

- Для кнопки **Отправить** необходимо назначить имя `ediSignAndSendButton`
- Для кнопки **Отозвать** необходимо назначить имя `ediRecallButton`
- `edi_OutgoingInvitation_cancelClick` — событие нажатия на кнопку **Отозвать**.
- `edi_OutgoingInvitation_signAndSendClick` — событие нажатия на кнопку — **Подписать и отправить**.
 - Событие требует заполнения поля `Contract.PartnerCompany` — организация контрагента, получатель приглашения.
- `edi_OutgoingInvitation_cardOpening` — перед открытием карточки.
- `edi_OutgoingInvitation_cardOpened` — после открытия карточки.

Кнопка **Отправить** доступна только если ещё приглашение не было отправлено ранее. Для повторной отправки потребуется создать новую карточку. Кнопка **Отозвать** доступна в отправленном приглашении, пока приглашение не принято или не отправлен отказ.

События и обработчики событий входящих приглашений:

- `edi_IncomingInvitation_cardOpening` — событие перед открытием карточки.
- `edi_IncomingInvitation_cardOpened` — событие после открытия карточки.
- `edi_IncomingInvitation_acceptInvitationClick` — событие нажатия на кнопку — **Принять**.
- `edi_IncomingInvitation_declineInvitationClick` — событие нажатия на кнопку — **Отклонить**.
- Кнопка **Принять** приглашение должна иметь имя `ediAcceptInvitationButton`.
- Кнопка **Отклонить** приглашение должна иметь имя `ediDeclineInvitationButton`.

Настройка ролевой модели для группы входящих документов

Администратор должен настроить ролевую модель для вида карточек входящего неформализованного документа.

1. Откройте *Конструктор ролей*.

2. Выберите вид карточки входящего неформализованного документа, указанный в **блоке настроек Компоненты > Получение** в *Справочнике настроек операторов ЮЗДО*.

По умолчанию для всех типов неформализованных документов используется вид *Документ УД / Входящий*. Если для разных типов неформализованных документов выбраны разные виды карточек Docsvision, следующие действия нужно будет повторить для каждого вида.

3. Создайте новую роль (например, *Регистратор ЮЗДО*).

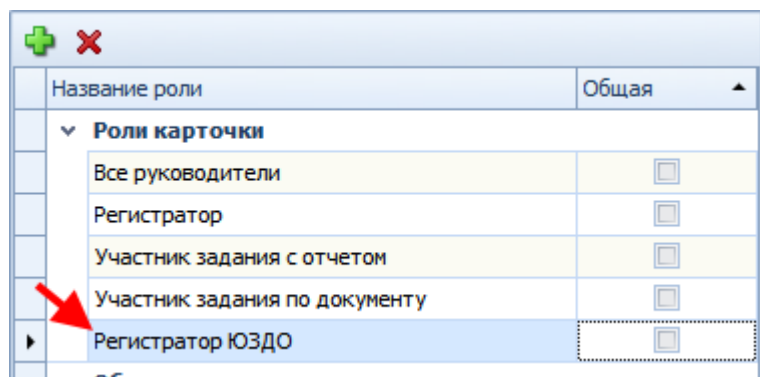


Рисунок 26. Новая роль в Конструкторе ролей

4. Добавьте для роли условие: *Я в группе "Получатели документов"*.

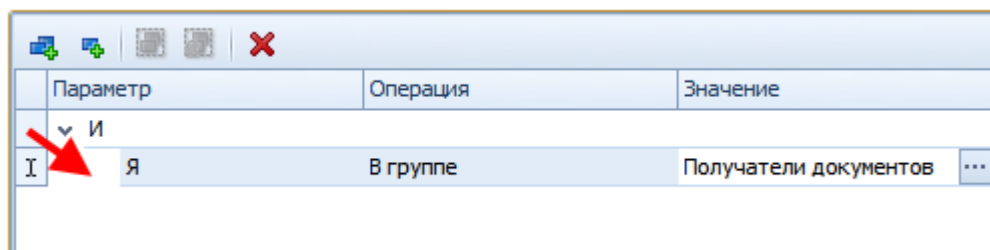


Рисунок 27. Новое условие в Конструкторе ролей

Если кроме группы сотрудников *Получатели документов* были созданы другие группы, сотрудники которых будут осуществлять обработку поступающих от контрагентов документов, включите их в условие с типом **ИЛИ**.

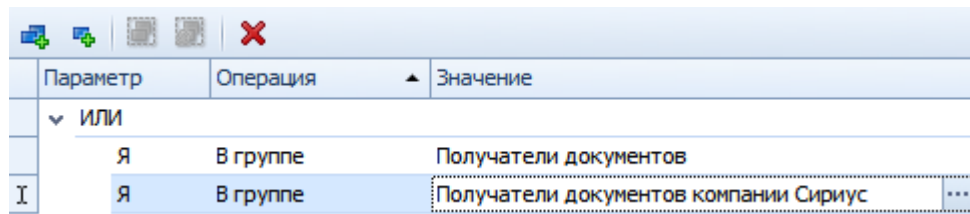


Рисунок 28. Условие с двумя группами сотрудников в Конструкторе ролей

5. Настройте матрицу доступа для созданной роли.

Роли должны быть предоставлены права для состояния **Подготовка** на следующие операции:

- Выгрузка документа.
- Запись.
- Зарегистрировать.
- Назначение категории.
- Отправка документа.
- Отправка сообщения по документу.
- Печать документа.
- Печать штрих-кода.
- Просмотр истории.
- Редактирование регистрационных данных.
- Удаление категории.
- Чтение.
- Чтение дополнительного файла.
- Чтение основного файла.

Библиотека карточек: **Базовые объекты** Вид карточки: **Входящий**

Роли Матрица доступа

Тип объекта: **Роль** Объект: **Регистратор ЮЗДО**

				В архиве	Подготовка	Зарегистрирован
Вернуть на подготовку	✓	☐	✗	☐	☐	☐
Выгрузка документа	✓	☐	✗	☐	✓	☐
Длительная блокировка основного файла	✓	☐	✗	☐	☐	☐
Добавление дополнительного файла	✓	☐	✗	☐	☐	☐
Добавление комментария к версии основного файла	✓	☐	✗	☐	☐	☐
Добавление основного файла	✓	☐	✗	☐	☐	☐
Запись	✓	☐	✗	☐	✓	☐
Зарегистрировать	✓	☐	✗	☐	✓	☐
Изменение штрих-кода	✓	☐	✗	☐	☐	☐
Назначение категории	✓	☐	✗	☐	✓	☐
Ответ	✓	☐	✗	☐	☐	☐
Отправка документа	✓	☐	✗	☐	✓	☐
Отправка на ознакомление	✓	☐	✗	☐	☐	☐
Отправка сообщения по документу	✓	☐	✗	☐	✓	☐
Переименование файлов	✓	☐	✗	☐	☐	☐
Печать документа	✓	☐	✗	☐	✓	☐
Печать штрих-кода	✓	☐	✗	☐	✓	☐

Рисунок 29. Матрица доступа для новой роли в Конструкторе ролей

6. Сохраните изменения.

Настройка функций обмена неформализованными документами

Для работы с неформализованными документами администратору нужно настроить разметки карточек Docsvision, которые будут использоваться для обмена электронными документами: для отправки, получения и аннулирования.

Стандартные настройки модуля предусматривают, что для получения электронных документов будет использоваться вид *Документ/Входящий*, а для отправки — любой вид документа, например, *Документ/Исходящий*.



Виды *Входящий* и *Исходящий* являются стандартными видами для организации ЭДО неформализованными документами, но не обязательными.

При организации ЭДО на других видах карточек настройте [отправку](#) и [получение](#) с их использованием в *Справочнике настроек операторов ЮЗДО*.

Настройка разметки вида "Документ/Входящий"

1. Откройте в *Конструкторе разметок* настройки вида *Документ/Входящий*.
2. Добавьте на ленту карточки новую группу с названием *Электронный обмен*.
3. Добавьте в группу *Электронный обмен* команды:
 - **Подписание документа.**
 - **Отказ в подписании документа.**
 - **Запросить аннулирование.**
 - **Подтвердить аннулирование.**
 - **Отказать в аннулировании.**
4. Укажите операции доступа для добавленных команд.
5. Добавьте в скрипт карточки обработчики для новых команд.
 - а. Добавьте в скрипт карточки ссылку на дополнительную сборку `DocsVision.Edi.DocumentScript.dll` располагается в каталоге клиента Docsvision.



Рекомендуется подключать сборку по её полному имени, а не с указанием абсолютного пути.

b. В раздел подключаемых пространств имен добавьте:

```
using DocsVision.Edi.DocumentScript;
```

c. Добавьте обработчики для новых команд:

- Для команды *Подписание документа*:

```
private void sendReplySignature_ItemClick(System.Object sender, DevExpress
.XtraBars.ItemClickEventArgs e)
{
    EdiScriptHelper ediScriptHelper = new EdiScriptHelper(CardControl);
    ediScriptHelper.SendReplySignature();
}
```

- Для команды *Отказ в подписании документа*:

```
private void sendReplySignatureRejectionButton_ItemClick(System.Object sender,
DevExpress.XtraBars.ItemClickEventArgs e)
{
    EdiScriptHelper ediScriptHelper = new EdiScriptHelper(CardControl);
    ediScriptHelper.SendReplySignatureRejection(); ①
}
```

① `ediScriptHelper.SendReplySignatureRejection();` следует использовать только, если точно известно, что ответ делается внутри Диадок. Для надежности проще всегда использовать `EdiScriptHelper.SendInvoiceReplyRejection();` для неформализованных документов по аналогии с формализованными. Web-клиент работает похожим образом: `SendInvoiceReplyRejection` через `boxesService.GenerateSignatureRejection`.

- Для команды *Запросить аннулирование*:

```
private void sendRequestRevocation_ItemClick(System.Object sender, DevExpress
.XtraBars.ItemClickEventArgs e)
{
    EdiScriptHelper ediScriptHelper = new EdiScriptHelper(CardControl);
    ediScriptHelper.SendRevocation();
}
```

- Для команды *Подтвердить аннулирование*:

```
private void sendApproveRevocation_ItemClick(System.Object sender,
DevExpress.XtraBars.ItemClickEventArgs e)
{
    EdiScriptHelper ediScriptHelper = new EdiScriptHelper(CardControl);
    ediScriptHelper.SendReplyRevocationSignature();
}
```

- Для команды *Отказать в аннулировании*:

```
private void sendRejectRevocation_ItemClick(System.Object sender, DevExpress
.XtraBars.ItemClickEventArgs e)
{
    EdiScriptHelper ediScriptHelper = new EdiScriptHelper(CardControl);
    ediScriptHelper.SendReplyRevocationRejection();
}
```

6. Добавьте в разметку карточки элемент управления *Журнал обмена сообщениями*. Рекомендуется размещать элемент управления на новой вкладке (например, *Журнал обмена ЮЗДО*). Настраивать элемент управления не требуется.

Настройка разметки вида "Документ/Исходящий"

1. Откройте в *Конструкторе разметок* настройки вида *Документ/Исходящий*.
2. Добавьте на ленту карточки новую группу с названием *Электронный обмен*.
3. Добавьте в группу *Электронный обмен* команды:
 - **Отправка документов.**
 - **Отправка документов с выбором файлов.**
 - **Отправка документов с подписанием.**
 - **Отправка документов с подписанием и выбором файлов.**
 - **Запросить аннулирование.**
 - **Подтвердить аннулирование.**
 - **Отказать в аннулировании.**
4. Укажите операции доступа для добавленных команд.
5. Добавьте в скрипт карточки обработчики для новых команд.

- a. Добавьте в скрипт карточки ссылку на дополнительную сборку `DocsVision.Edi.DocumentScript.dll`.
- b. В раздел подключаемых пространств имен добавьте:

```
using DocsVision.Edi.DocumentScript;
```

- c. Добавьте обработчики для новых команд:

- Для команды *Отправка документов*:

```
private void sendDocumentButton_ItemClick(System.Object sender, DevExpress
.XtraBars.ItemClickEventArgs e)
{
    if (!CardControl.Save())
        return;

    EdiScriptHelper ediScriptHelper = new EdiScriptHelper(CardControl);
    ediScriptHelper.SendDocument();
}
```

- Для команды *Отправка документов с выбором файлов*:

```
private void sendDocumentFilesButton_ItemClick(System.Object sender,
DevExpress.XtraBars.ItemClickEventArgs e)
{
    if (!CardControl.Save())
        return;

    EdiScriptHelper ediScriptHelper = new EdiScriptHelper(CardControl);
    ediScriptHelper.SendDocumentFiles();
}
```

- Для команды *Отправка документов с подписанием*:

```
private void signAndSendDocumentButton_ItemClick(System.Object sender,
DevExpress.XtraBars.ItemClickEventArgs e)
{
    if (!CardControl.Save())
        return;

    EdiScriptHelper ediScriptHelper = new EdiScriptHelper(CardControl);
    ediScriptHelper.SignAndSendDocument();
}
```

```
}
```

- Для команды *Отправка документов с подписанием и выбором файлов*:

```
private void signAndSendDocumentFilesButton_ItemClick(System.Object sender,
DevExpress.XtraBars.ItemClickEventArgs e)
{
    if (!CardControl.Save())
        return;

    EdiScriptHelper ediScriptHelper = new EdiScriptHelper(CardControl);
    ediScriptHelper.SignAndSendDocumentFiles();
}
```

- Для команд *Запросить аннулирование, Подтвердить аннулирование и Отказать в аннулировании* скрипты аналогичны скриптам вида *Документ/Входящий* (см. [выше](#)).

6. Добавьте в разметку карточки элемент управления *Журнал обмена сообщениями*. Рекомендуется размещать элемент управления на новой вкладке (например, *Журнал обмена ЮЗДО*). Настраивать элемент управления не требуется.

Отображение типа неформализованного документа в карточке

Если требуется, в разметку карточки можно добавить текстовое поле для отображения типа неформализованного документа. Тип неформализованного документа содержится в расширенном поле *Данные УПД/Тип документа* карточки *Документ*.

Дополнительные функции документа

- Если модуль устанавливался ранее и версия модуля в БД была обновлена до последней, в конструкторе разметок в поле карточки документа *Данные УПД / Функция документа* (*UniversalDocumentData / DocumentFunction*) можно добавить значения *СвРК = 3, СвЗК = 4*, это будет соответствовать новым значениям атрибута *Функция УПД*.
- Если модуль устанавливается впервые или после обновления была создана новая БД, дополнительных действий не требуется.

Настройки web-разметок

1. Откройте в *Конструкторе разметок* настройки желаемого вида документа.
2. Добавьте для вида карточки новые кнопки команд.

Для карточек вида *Документ/Входящий*:

- **Подписание документа.**
- **Отказ в подписании документа.**
- **Запросить аннулирование.**
- **Подтвердить аннулирование.**
- **Отказать в аннулировании.**

Для настройки вида *Документ/Исходящий* добавьте следующие команды:

- **Отправка документов.**
- **Отправка документов с подписанием.**
- **Запросить аннулирование.**
- **Подтвердить аннулирование.**
- **Отказать в аннулировании.**

3. Укажите операции доступа для добавленных команд.
4. Добавьте для новых команд в события **При щелчке** обработчики команд:

Обработчик	Команда	Описание команды
<code>signIncoming</code>	Подписать	Используется в неформализованных входящих документах.
<code>signAndSendOutgoing</code>	Подписать и отправить	При нажатии будет вызвана функция подписания документа с отправкой контрагенту. Используется в неформализованных исходящих документах.

Обработчик	Команда	Описание команды
<code>sendIncomingWithExistingSign</code>	Отправка документов с имеющейся подписью	При нажатии будет вызвана функция отправки входящего документа с существующей подписью.
<code>sendOutgoingWithExistingSign</code>	Отправка документов с имеющейся подписью	При нажатии будет вызвана функция отправки исходящего документа с существующей подписью.
<code>revoke</code>	Аннулировать	При нажатии будет вызвана функция аннулирования формализованного документа. Используется в любых неформализованных входящих документах.
<code>approveRevocation</code>	Подтвердить аннулирование	Используется для подтверждения аннулирования. При этом потребуется формирование ответной подписи при помощи сертификата квалифицированной подписи. Используется в любых неформализованных документах.

Обработчик	Команда	Описание команды
<code>rejectRevocation</code>	Отказать в аннулировании	Для отказа в аннулировании. При этом потребуется указать причину отказа. Используется в любых неформализованных документах.
<code>rejectAndSendInvoice</code>	Отказать	При нажатии будет вызвана функция отказа в подписании с указанием причины отказа. Используется в неформализованных входящих документах.

5. При настройке разметок следует учитывать, что данные о контрагенте для отправки берутся из секции *Получатели — контрагенты*, *Получатель — организация контрагента* — поле в этой секции.

Настройки "Журнала обмена ЭДО"

При инсталляции модуля добавляется *Журнал обмена ЭДО* для Web-клиента. Журнал представлен плагином `EDIMessagesLog` для ЭУ *Контроль табличных данных*. Плагин преобразует элемент управления в *Журнал обмена ЮЗДО* для Web-клиент.

Чтобы отобразить журнал в разметке документов:

1. Добавьте ЭУ
`undefined/webclient/5.5.17/layouts/ctrl/table/dataGridControl/["Контроль табличных данных"]` в разметку неформализованного документа.
2. В свойствах **Название** и **Название плагина** укажите `EDIMessagesLog`.
3. Добавьте скрипт `edi_loadMessageLogSignatureStatus` для события **После открытия карточки** для элемента **root**, чтобы загрузка статуса подписей и подписантов выполнялась асинхронно.

Настройка функций обмена формализованными документами

Для работы с формализованными документами предназначены карточки подвидов *Документ/УПД*:

1. Исходящий УПД.
 - a. Исходящий УКД.
 - i. Исходящий УКД (736) в формате приказа ФНС 189.
 - ii. Исходящий УКД (736) в формате приказа ФНС 736.
 - b. Исходящее исправление УПД.
 - i. Исходящее исправление УПД (820).
 - c. Исходящий УПД (820).
2. Входящий УПД.
 - a. Входящий УПД (820).
 - b. Формализованный акт.
 - i. Формализованный акт (820).
 - c. Входящий УКД.
 - i. Входящий УКД (736) в формате приказа ФНС 189.
 - ii. Входящий УКД (736) в формате приказа ФНС 736.

Указанные виды карточек предоставляются с настроенными разметками, автоматом состояний и ролевой моделью.

Администратору необходимо:

1. Добавить [печатные формы для печати УПД](#).
2. Добавить виртуальные папки для работы с УПД.

Настройка печатных форм для формализованных документов

1. Откройте *Рабочее место администратора*.
2. Вызовите команду *Свойства* из контекстного меню элемента **Карточки** > **Базовые объекты** > **Документ**.

Будет открыто окно со свойствами типа карточки *Документ*.

3. Перейдите на страницу *Преобразования* и создайте преобразования:

a. *UniInvoiceRu* с параметрами:

- Название: *UniInvoiceRu*.
- Псевдоним: *UniInvoiceRu*.
- Тип преобразование: **Пользовательское**.
- Тип содержимого: **XML**.
- Данные преобразования (выберите файл): *C:\Program Files (x86)\Docsvision\Edi\5.5\Xslt\UniInvoice.ru.xslt*.

b. Выполните аналогичные настройки со следующими преобразованиями:

- *UniInvoiceActRu*
- *UniInvoiceCorrectionRu*
- *UniInvoiceCorrFields*
- *UniInvoiceFields*
- *UniInvoiceSF*
- *UniInvoiceTorgRu*

Настройка папок для формализованных документов

Для удобства работы с исходящими формализованными документами можно добавить пользователям модуля виртуальные папки со следующими поисковыми запросами:

- *Журнал Все УПД* — возвращает все виды УПД во всех статусах.
- *Журнал Исправление УПД* — возвращает только исправления УПД во всех статусах.
- *Журнал Исходящий УПД* — возвращает только исходящие УПД во всех статусах.
- *Журнал Корректировочный УПД* — только корректировочные УПД во всех статусах.
- *Поиск УПД* — выполняет поиск УПД по различным полям.

В настройках данных папок рекомендуется установить представление по умолчанию — *УПД (Все)*.

Поисковые запросы и представления для папок входящих формализованных документов, если требуется, могут быть разработаны самостоятельно.

Настройка ленты карточек для нового вида formalизованных документов

Администратор может настроить собственные виды карточек для работы с formalизованными документами. При этом добавляемые виды должны быть дочерними видами видов *Входящий УПД* или *Исходящий УПД*.

Ленты карточек видов *Входящий УПД* и *Исходящий УПД* содержат команды для вызова стандартных функций обработки formalизованных документов. В собственных видах могут использоваться функции, унаследованные от родительских видов, или разработаны собственные, например, дополнительная функция проверки данных.



Ниже приведен упрощенный код функций, вызываемых командами ленты карточек. Полный код смотрите в скриптах карточек соответствующих родительских видов.

1. Входящий УПД и подвиды.

а. Подписать документ:

```
private void SignDocument_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    EdiScriptHelper.SendInvoiceReplySignature();
}
```

б. Отказать в подписании:

```
private void RejectSingning_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    EdiScriptHelper.SendInvoiceReplyRejection(); ①
}
```

① `ediScriptHelper.SendReplySignatureRejection();` следует использовать только, если точно известно, что ответ делается внутри Диадок. Для надежности проще всегда использовать `EdiScriptHelper.SendInvoiceReplyRejection();` для неформализованных документов по аналогии с formalизованными. Web-клиент работает похожим образом: `SendInvoiceReplyRejection` через `boxesService.GenerateSignatureRejection`.

c. *Запросить аннулирование*

```
private void RequestRevocation_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    EdiScriptHelper.SendRevocation();
}
```

d. *Подтвердить аннулирование*

```
private void ApproveRevocation_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    EdiScriptHelper.SendReplyRevocationSignature();
}
```

e. *Отказать в аннулировании*

```
private void RejectRevocation_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    EdiScriptHelper.SendReplyRevocationRejection();
}
```

f. *Запросить уточнение:*

```
private void RequestForCorrection_ItemClick(System.Object sender, DevExpress
.XtraBars.ItemClickEventArgs e)
{
    EdiScriptHelper.SendCorrection()
}
```

9. *Печать* — выводить на печать печатную форму формализованного документа:

```
private void Print_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    EdiScriptHelper.PrintInvoice("UniInvoiceRU"); ①
}
```

- ① В параметре метода указывается название шаблона печати:
- **UniInvoiceRU** — стандартный шаблон для видов *Входящий УПД* и *Формализованный акт*.
 - **UniInvoiceCorrectionRU** — стандартный шаблон для вида *Входящий УКД*.

h. Обновление печатной формы (вкладка *Печатная форма*). Данная функция должна вызываться при активации карточки (*CardActivated*).

```
public virtual void FillInvoiceControl()
{
    EdiScriptHelper.FillInvoiceHtmlControl("HTMLBrowser", "UniInvoiceRU"); ①
}
```

- ① В первом параметре (**HTMLBrowser**) должно быть указано название обновляемого элемента управления с типом HTML браузер. Во втором параметре (**UniInvoiceRU**) должно быть указано название шаблона печати для обновления.

Названия стандартных шаблонов аналогичны приведенным для **Печати**.

2. Исходящий УПД и подвиды.

a. *Подписать и отправить* — выдаёт запрос на подписание документа квалифицированной ЭП (электронной подписью) и формирует электронное сообщение с документом для отправки оператору ЭДО.

```
private void SignedAndSent_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    CardControl.Save();
    if (EdiScriptHelper.SignAndSendInvoice())
    {
        ChangeCardState("SignedAndSent");
    }
}
```

b. *Заполнить данные из файла* — загружает в карточку данные из приложенного основного файла — XML-файл формализованного документа, сформированный у оператора ЭДО.

```
private void FillingOutData_ItemClick(System.Object sender, DevExpress.XtraBars
```

```
.ItemClickEventArgs e)
{
    CardControl.Save();
    EdiScriptHelper.UpdateDocumentDataFromInvoice();
    RefreshControls();
}
```

В Web-клиенте за заполнение данных карточки отвечает обработчик `edi_updateDocumentFormInvoice`. Обработчик может быть привязан к кнопке по аналогии с Windows-клиентом или быть назначен на события добавления главного файла и добавления версии файла.

Обработчик проверяет расширение файла. Если формат файла не соответствует ожидаемому (`.xml`) или в файле содержатся ошибки, будет сообщение об ошибке. Проверка содержимого выполняется по аналогии с Windows-клиентом.

Через клиентский API обработчик можно вызвать обращением к сервису `$EdiFileSync` с помощью метода:

```
updateDocumentDataFromInvoice(documentId: string, fileId?: string)
```

На сервере можно обратиться к следующему сервису:

```
IEdiFileSyncService.UpdateDocumentDataFromInvoice
```

c. Запросить аннулирование:

```
private void RequestRevocation_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    EdiScriptHelper.SendRevocation();
}
```

d. Подтвердить аннулирование:

```
private void ApproveRevocation_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
```



```

        EdiScriptHelper.SendReplyRevocationSignature();
    }

```

е. Отказать в аннулировании:

```

private void RejectRevocation_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    EdiScriptHelper.SendReplyRevocationRejection();
}

```

ф. Печать:

```

private void PrintUPD_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    EdiScriptHelper.PrintInvoice("UniInvoiceRU"); ①
}

```

① В параметре метода указывается название шаблона печати:

- **UniInvoiceRU** — стандартный шаблон для видов *Исходящий УПД* и *Исправление УПД*.
- **UniInvoiceCorrectionRu** — стандартный шаблон для вида *Исходящий УКД*.

г. Обновление печатной формы (вкладка *Печатная форма*). Данная функция должна вызываться при активации карточки (*CardActivated*).

```

public virtual void FillInvoiceControl()
{
    EdiScriptHelper.FillInvoiceHtmlControl("HTMLBrowser", "UniInvoiceRU"); ①
}

```

① В первом параметре ("HTMLBrowser") должно быть указано название обновляемого элемента управления с типом HTML браузер.

Во втором параметре (**UniInvoiceRU**) должно быть указано название шаблона печати для обновления.

Названия стандартных шаблонов аналогичны приведенным для [Печати](#).

В приведенном коде **EdiScriptHelper** объявлен в корневом виде *УПД*:

```

public EdiScriptHelper EdiScriptHelper
{
    get
    {
        if (ediScriptHelper == null)
            ediScriptHelper = new EdiScriptHelper(CardControl);

        return ediScriptHelper;
    }
}

```

Для работы указанных методов к скрипту карточки должны подключены сборки `DocsVision.Edi.DocumentScript.dll` и `Docsvision.DocumentsManagement.dll`. Сборки располагаются в каталоге клиента Docsvision.

В объявление пространств имен нужно добавить строки:

```

using DocsVision.Edi.DocumentScript;
using Docsvision.DocumentsManagement;

```

Для добавленных на ленту кнопок можно добавить методы их скрытия/отображения в зависимости от статуса обмена — проверяется с помощью методов:

- `IsSignatureReplyAvailable` — проверяет возможность подписания документа или отказа в подписании, принимает значение `TRUE`, если документ в статусе `Получен на подпись от контрагента`.
- `IsRevocationReplyAvailable` — проверяет возможность аннулирования документа или отказа в аннулировании; принимает значение `TRUE`, если документ в статусе `Получен запрос на аннулирование`.
- `IsRevocationRequestAvailable` — проверяет возможность создания запроса на аннулирование документа, принимает значение `TRUE`, если документ в статусе `Получен от контрагента`, `Отправлена ответная подпись контрагенту` или `Отправлен контрагенту`.
- `IsReceiptReplyAvailable` — проверяет возможность создания ответной квитанции.

Методы могут вызываться при открытии карточки. Методы не изменяют состояние видимости кнопок ленты — необходимо реализовать самостоятельно.

Собственный вид, не унаследованный от Входящий УПД или Исходящий УПД

Если требуется создать вид, не наследующий состояния от *Входящий УПД* или *Исходящий УПД*, необходимо самостоятельно добавить в его автомат состояний следующие состояния:

- Получена подпись от контрагента с идентификатором встроенного состояния (поле `BuildInState`) `FF1346D7-93F0-4CA3-93F8-5AE47BCD41DD`.
- Получен отказ от контрагента с идентификатором встроенного состояния `D136A114-7C59-4C6E-AB12-5348B4883AB9`.
- Требуется уточнение с идентификатором встроенного состояния `9F8A156B-68EE-4543-9527-BF47B613330E`.
- Запрошено аннулирование с идентификатором встроенного состояния `E3ED8CD1-8D9E-4332-B2D3-0EE5B9FE5A10`.
- Аннулирование подтверждено с идентификатором встроенного состояния `D5CC4959-5E45-4F31-B630-F2C3613EDCB7`.
- В аннулировании отказано с идентификатором встроенного состояния `ED9B6929-BD9C-496B-B8A7-D9597B1BA5CA`.

Также потребуется добавить в разметку элементы управления с типами и названиями, которые используются в видах *Входящий УПД* или *Исходящий УПД*.

Настройка формата "Акт (552)"

Модуль интеграции с операторами ЭДО поддерживает работу с актами формата 552. Для работы с актами данного формата необходимо выполнить настройки:

1. Создать виды:
 - *Входящий акт (552)*, унаследованный от *Формализованный акт*.
 - *Исходящий акт (552)*, унаследованный от *Исходящий УПД*.
2. В Справочнике настроек операторов ЮЗДО настроить:

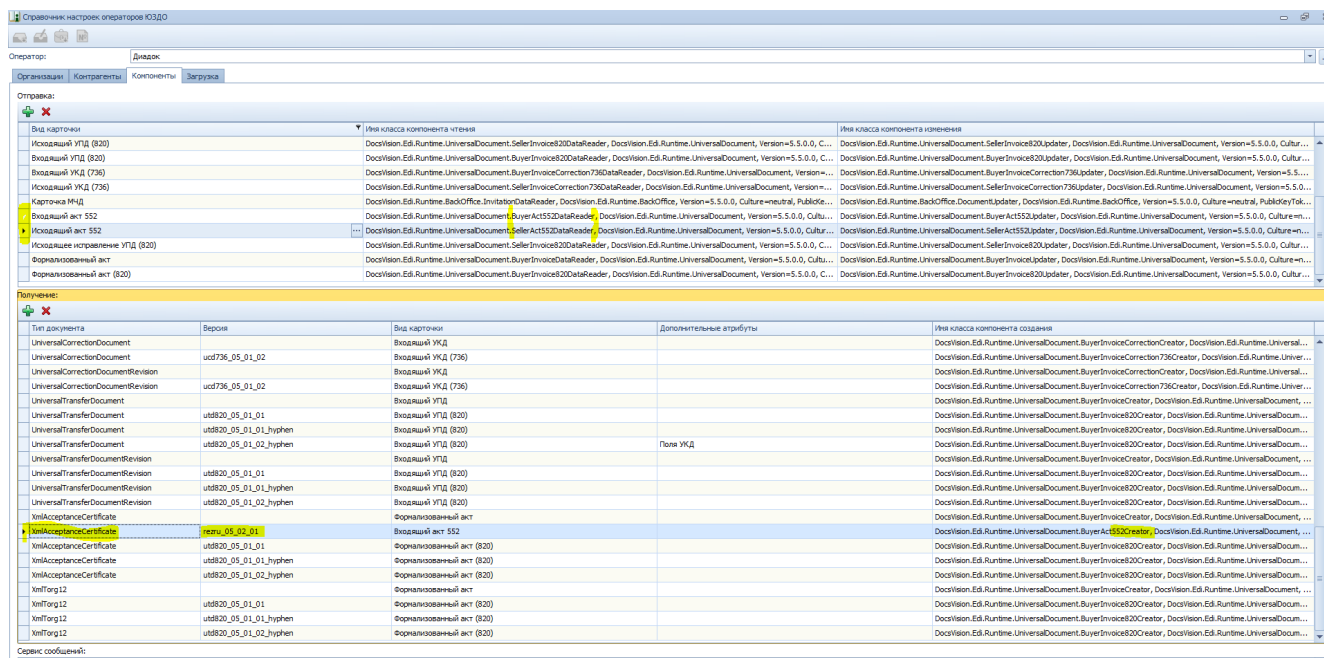


Рисунок 30. Настройки для видов Исходящий акт (552) и Входящий акт (552)

- В блоке настроек **Получение** для **Типа документа** **XmlAcceptanceCertificate**, укажите **Версию** **rezru_05_02_01**.
- Загрузите шаблон **CardDefs\Xslt\UniInvoiceAct552.ru.xslt** для карточки документа с псевдонимом **UniInvoiceAct552Ru** через РМА.
- Для созданных видов добавьте скрипты:

```
namespace BackOffice
{
    #region Properties
    #endregion
    #region Methods

    public override void FillInvoiceControl()
    {
        EdiScriptHelper.FillInvoiceHtmlControl("HTMLBrowser", "UniInvoiceAct552Ru");
    }

    #endregion
    #region Event Handlers

    private void Print_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
    {
        EdiScriptHelper.PrintInvoice("UniInvoiceAct552Ru");
    }

    #endregion
}
```

6. В разметке для входящего не используются поля получателя, кроме поля *Область полномочий покупателя*.

Поле *Область полномочий подписанта* можно заменить другим полем: *Область полномочий покупателя*. На рисунке ниже представлен минимальный набор данных для подписания:

Входящий Акт № 132846-17-H222000660 от 07.10.2017 - Документ

Документ

Сохранить Печать Запросить аннулирование Журнал подписей

Общие Действия Подписи и шифр...

Номер документа: 132846-17-H222000660 Дата документа: 16.03.2023 Тип документа: Формализованный акт

Статус УПД: Документ подписан с обеих сторон. Функция УПД:

Основная информация Подписание Журнал обмена ЮЗДО Печатная форма

Вид операции: Вид Содержание операции: Принято без претензий

Текст разногласий: Дата принятия: 16.03.2023

Основание полномочий организации: Основание 1

Сведения о подписанте

Организация подписанта: Доксвижн консалт Основание полномочий организации: Основание 2

Основание доверия к организации:

Сведения о гос. регистрации:

Подписант: Сабитов А. М. Область полномочий подписанта:

Тип подписанта: Юридическое лицо Статус подписанта: Работник организации покупателя

Иные сведения о подписанте:

Сведения о получателе

Основания полномочий получателя:

Область полномочий получателя: Лицо, совершившее сделку, операцию Статус получателя:

Уполномоченная организация: Уполномоченное лицо:

Основание доверия получателя

Иные сведения о получателе:

Рисунок 31. Минимальный набор данных для подписания

Настройка обмена пакетами с Диадок

Данное расширение модуля позволяет послать в одном пакете Диадок не только файл УПД или другой структурированный документ, но и связанные с ним неструктурированные документы.



Чтобы неструктурированный документ оказался добавлен в пакет, он должен быть связан с УПД ссылкой типа *Связано с (ЮЗДО)* (см. ["Особенности формирования связей"](#) ниже).

Отправка пакета

При нажатии кнопки **Подписать** и отправить на форме УПД производятся следующие действия:

1. У связанного с УПД неформализованного документа проверяется наличие в поле Контрагент записи соответствующей контрагенту-получателю УПД. Если запись не соответствует или отсутствует, будет выведено сообщение об ошибке и предложено указать контрагента.
2. При помощи ЭП подписывается УПД и связанные с ним документы.
3. Формируется диалоговое окно с соответствующими изменениями. В окне отображается информация о том, что был сформирован пакет документов.
4. По нажатию **ОК** формируется пакет и передается на отправку.
5. Также изменяются записи в журнале обмена ЮЗДО карточки УПД. В журнале фиксируется информация о ходе обработки документа УПД и связанных файлов.

Получение Пакета

Если контрагент переслал пакет включающий УПД и дополнительные неформализованные документы, при получении пакета будут сформированы карточки УПД и приложенных документов с соответствующими связями.

Чтобы подписать полученный УПД со связанными документами, необходимо подписать документ, к которому привязаны остальные ссылками *Ссылается на (ЮЗДО)*. В этом случае пакет будет автоматически сформирован, подписан и отправлен.

При формировании запроса на уточнение, корректировку и исправление УПД, неформализованные документы, входящие в пакет, остаются неизменными.

Особенности формирования связей

При получении пакетов обрабатываются все возможные варианты связей со следующим приоритетом:

1. Если был найден родительский документ, в нём создаётся ссылка на текущий, а в текущий добавляется ссылка на родительский.
2. Если был найден документ, подчиненный текущему, в подчинённый добавляется ссылка на родительский (текущий), а в текущем создаётся ссылка на подчинённый.

3. Если был найден документ с тем же ID, первый формализованный документ в пакете будет определён как родительский и будут сформированы соответствующие ссылки.

Связанные документы — это всегда ссылка *Ссылается на (ЮЗДО)*. Никакие другие ссылки, включая обратную *Связано с (ЮЗДО)*, аналогичным образом не обрабатываются.

Если требуется работа со ссылками *Ссылается на (ЮЗДО)* или *Связано с (ЮЗДО)* в других карточках, необходимо разрешить ее выбор и показывать этот тип ссылки в списке ссылок. Глубина связей для любых обработок составляет всегда только один уровень вниз. Рекурсивный поиск связей не производится.

При подписании пакета учитывается наличие связей, но при этом остаётся возможность подписать любой связанный документ. Можно подписать пакет, в котором уже есть подписанные связанные документы, но родительский документ не подписан. В таком случае будет подписан родительский документ, а также связанные документы, у которых ещё нет подписи. Неподписанные документы будут отображены в сообщении.

Настройка функций отправки и получения пакетов

В рамках модуля реализована возможность добавить новые функции самостоятельно.

Для отправки пакета из родительского документа и всех связанных требуется вызвать метод:

```
EdiScriptHelper.SignAndSendWithLinked();
```

Вызов метода можно привязать к новой кнопке или заменить прежний вызов метода на требуемой кнопке отправки.

Для подписания пакета, включающего родительский документ и связанные требуется вызвать метод:

```
EdiScriptHelper.SendCardWithLinkedReplySignature();
```

Вызов метода можно привязать к новой кнопке или заменить прежний вызов метода на требуемой кнопке отправки.

Приложения

Модуль интеграции с операторами ЭДО

Приложения

Дополнительная информация для администратора Docsvision.

- [Процедура обмена неформализованными документами](#)
- [Процедура обмена формализованными документами](#)
- [Кнопки для работы с документами ЮЗДО в Web-клиенте](#)

Процедура обмена неформализованными документами

1. Пользователь инициирует функцию отправки неформализованного документа, воспользовавшись командой в карточке *Документ*.



Специалист, осуществляющий настройку модуля, с помощью API модуля и предоставляемой возможности разработки компонентов модуля может реализовать собственный сценарий отправки.

2. Пользователь определяет параметры отправки: отправляемые файлы, ящик организации и др., затем отправляет документ. При этом будет сформирована *карточка обмена сообщениями*.

До подтверждения пользователем отправки *карточка обмена сообщениями* находится в статусе **Создаётся**, после — **Готово к отправке**.



Все возможные статусы *Карточки обмена сообщениями* приведены в [описании схемы карточки](#).

Получателями данного неформализованного документа будут являться контрагенты, указанные в карточке *Документ*, для которых в *Справочнике настроек операторов ЮДЗО* определен ящик контрагента. Если данная связь не выявлена, например, у контрагента нет ящика, будет выдано предупреждение.

3. *Карточки обмена сообщениями* со статусом **Готово к отправке** обрабатываются бизнес-процессом **Отправка сообщений ЮЗДО**, который формирует по карточкам

сообщения электронного обмена и отправляет их оператору ЭДО. Статус отправленных карточек изменяется на **Отправлен контрагенту**.

4. Оператор ЭДО получает сообщение электронного обмена и передает его контрагенту.
5. Контрагент обрабатывает полученный документ и формирует на него ответ (положительный или отрицательный). Также контрагент может передать собственный неформализованный документ.
6. Бизнес-процесс *Получение сообщений ЮЗДО* получает электронные сообщения от оператора ЭДО.
7. Если входящее сообщение содержит ответ на ранее отправленный неформализованный документ, то статус соответствующей ему *Карточки обмена сообщениями* будет изменен на одно из значений:
 - *Получен отказ от контрагента* — если от контрагента получен отказ в подписании.
 - *Получена подпись от контрагента* — если от контрагента получено положительное решение по подписанию.
 - *Получен запрос на аннулирование* — если от контрагента получен запрос на аннулирование документа.
 - *Аннулирован* — если от контрагента получено подтверждение запроса на аннулирование документа.
 - *Ошибка* — если при обмене электронными сообщениями с данным неформализованным документом произошли ошибки на стороне Docsvision, оператора ЭДО или контрагента.
8. Если входящее сообщение содержит новый документ, будет создана карточка входящего документа и соответствующая ей *Карточка обмена сообщениями*.
 - Если контрагент запросил подпись — документ будет создан со статусом **Получен на подпись от контрагента**. Пользователь должен будет отправить подпись документа или отказ в подписании.
 - Если контрагент не запросил подпись — документ будет создан со статусом **Получен от контрагента**.

Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или

контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

+ . Если входящее сообщение содержит запрос на аннулирование ранее полученного неформализованного документа, статус документа будет изменен на *Получен запрос на аннулирование*.

Процедура обмена формализованными документами

Отправка

1. Пользователь инициирует функцию отправки формализованного документа, воспользовавшись командой в карточке исходящего формализованного документа.
2. Пользователь выбирает ящик организации и отправляет документ. Если в документе нет ошибок и заполнены все обязательные атрибуты, формируется *Карточка обмена сообщениями*.

После подтверждения пользователем отправки *Карточка обмена сообщениями* переходит в статус **Готово к отправке**. После отправки карточка УПД и статус обмена в журнале обмена карточки ЮЗДО переходят в состояние **Подписан и отправлен**.

3. *Карточки обмена сообщениями* с состоянием **Готово к отправке** обрабатываются бизнес-процессом *Отправка сообщений ЮЗДО*, который формирует по карточкам сообщения электронного обмена и отправляет их оператору ЭДО. Отправленные карточки переводятся в состояние **Отправлен контрагенту**.
4. Оператор ЭДО получает сообщение электронного обмена и передает контрагенту формализованный документ. По факту получения титула продавца контрагенту и отправителю формализованного документа направляется квитанция.

Статус обмена в журнале обмена карточки УПД переходит в **Требуется подтверждение получения**.

5. Пользователь подтверждает получение квитанции о доставке титула продавца.

Статус обмена в журнале обмена карточки УПД переходит в **Отправлен контрагенту**.

6. Контрагент обрабатывает полученный формализованный документ и формирует ответ, который может содержать:

- Титул покупателя.
- Отказ в подписи.
- Запрос на уточнение.

7. Бизнес-процесс *Получение сообщений ЮЗДО* получает электронные сообщения от оператора ЭДО.

8. В зависимости от ответа контрагента:

- Если был передан титул покупателя:
 - а. В карточку *Исходящего УПД* добавляется файл, содержащий титул покупателя.
 - б. Карточка УПД и статус обмена в журнале обмена карточки УПД переходят в состояние **Получена подпись от контрагента**.
- Если был передан отказ в подписи — карточка УПД и статус обмена в журнале обмена карточки УПД переходят в состояние **Получен отказ от контрагента**.
- Если был передан запрос на уточнение — статус обмена в журнале обмена карточки УПД переходит в состояние **Требуется подтверждение**, после отправки — в **Требуется уточнение**.

9. Если был получен отказ в подписи, работа с данным документом прекращается.

10. Если был передан запрос на уточнение — пользователь формирует *Корректировочный УПД* или *Исправление УПД*.

Получение

1. Бизнес-процесс *Получение сообщений ЮЗДО* получает электронные сообщения от оператора ЭДО.

По формализованным документам формируются карточки соответствующих видов со статусом обмена **Получен от контрагента** или **Получен на подпись от**

контрагента.

2. По полученным УПД, исправление УПД, корректировка УПД, счёта-фактура, исправление счета-фактуры и корректировка счета фактуры со статусом **Требуется подтверждение получения** пользователь должен отправить подтверждение получения извещения.

После получения подтверждения оператор ЭДО отправит извещение о получении подтверждения, получение которого также должно быть подтверждено пользователем.

3. Для полученных накладных и акта пользователь может запросить аннулирование — статус обмена сменится на **Отправлен запрос на аннулирование**.
4. Для полученных счёта-фактуры, исправления счета-фактуры и корректировки счета фактуры пользователь может:
 - **Запросить аннулирование** — статус обмена сменится на **Отправлен запрос на аннулирование**.
 - **Запросить уточнение** — статус обмена сменится на **Отправлен запрос на уточнение**.
5. Для полученных УПД, исправление УПД и корректировка УПД пользователь может:
 - Подписать документ — статус обмена сменится на **Отправлена ответная подпись контрагенту**, работа с данным документом прекращается.

Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

+ * Отказать в подписании — статус обмена сменится на **Отправлен отказ от подписи контрагенту**. * Запросить уточнение — статус обмена сменится на **Отправлен**

запрос на уточнение.

+ . После подписания документа, пользователь может отправить запрос на аннулирование. Статус обмена сменится на **Отправлен запрос на аннулирование**, возможность выполнения других операций с документом — заблокирована.

+ Контрагент может аннулировать документ в одностороннем порядке:

+ * Для Акта и Накладной — до подписания. * Для остальных формализованных видов — до отправки первого извещения.

+ При одностороннем аннулировании документ сразу переходит в статус **Аннулирован**, работа с данным документом прекращается.

+ . Если был отправлен запрос на аннулирование:

+ * При отказе в аннулировании со стороны контрагента — статус обмена вернётся в: .. **Получен от контрагента** — для счёта-фактуры, исправления счета-фактуры и корректировки счета фактуры. .. **Отправлена ответная подпись контрагенту** — для УПД, исправление УПД, корректировка УПД, накладная и акт.

+ * При подтверждении аннулирования — статус обмена сменится на **Аннулирован**, работа с данным документом прекращается.

+ . Если был отправлен запрос на уточнение:

+ * Если документ был подписан или подписание не требуется — пользователь может отправить запрос на аннулирование. * Если документ не был подписан — работа с данным документом прекращается.

+ . Если был получен запрос на аннулирование от контрагента, статус обмена сменится на **Получен запрос на аннулирование**.

+ . Пользователь может: .. Подтвердить аннулирование — статус обмена сменится на **Аннулирован**, работа с данным документом прекращается. .. Отказать в аннулировании — статус обмена вернётся в предыдущее значение: **Получен от контрагента**, **Получен на подпись от контрагента** или **Отправлена ответная подпись контрагенту**.

Кнопки для работы с документами ЮЗДО в Web-клиенте

В web-разметках для работы с документами ЮЗДО используются кнопки, переданные в виде расширений.

Таблица 1. Кнопки в карточках документов вида "Исходящий УПД":

Название	Название ЭУ	Текст метки	События
approveRevocation	Кнопка (CustomButton)	Подтвердить аннулирование	При щелчке: approveRevocation
rejectRevocation	Кнопка (CustomButton)	Отказать в аннулировании	При щелчке: rejectRevocation
customPrintButton	Кнопка (CustomButton)	Печать	При щелчке: printInvoiceUniversalTransferDocument
signAndSend	Кнопка (CustomButton)	Подписать	При щелчке: signAndSendOutgoing
revoke	Кнопка (CustomButton)	Аннулировать	При щелчке: revoke

Таблица 2. Кнопки в карточках документов вида "Входящий УПД":

Название	Название ЭУ	Текст метки	События
customPrintButton	Кнопка (CustomButton)	Печать	При щелчке: printInvoiceUniversalTransferDocument
approveRevocation	Кнопка (CustomButton)	Подтвердить аннулирование	При щелчке: approveRevocation
rejectRevocation	Кнопка (CustomButton)	Отказать в аннулировании	При щелчке: rejectRevocation
signAndSend	Кнопка (CustomButton)	Подписать	При щелчке: signAndSendInvoice
rejectAndSend	Кнопка (CustomButton)	Отказать	При щелчке: rejectAndSendInvoice
sendReceipt	Кнопка (CustomButton)	Отправить извещение	При щелчке: receipt
revoke	Кнопка (CustomButton)	Аннулировать	При щелчке: revoke

Управление видимостью кнопок в зависимости от состояния обмена

Видимостью кнопок для работы с документами ЮЗДО в Web-клиенте можно

управлять с помощью обработчиков событий.

Обработчики для видимости кнопок:

- `edi_loadDocumentState` — на событие перед открытием карточки.
- `edi_updateButtonsVisibility` — на событие после открытия карточки.

Чтобы в карточке неформализованного и формализованного документа отображался статус обмена, добавьте в разметку ЭУ *Метка* с именем `ediStatus`. Чтобы добавить к нему заголовок, можно поместить его в ЭУ *Контейнер строка*, в котором необходимо задать метку. Заполнение выполняется тем же скриптом `edi_updateButtonsVisibility`.

Статусы документов:

В документе отсутствует заполненный формализованный файл

Если файл не прошел валидацию или его нет.

Документ не отправлен

Файл добавлен, но нет связанных с ним сообщений.

Документ подписан отправителем

Для входящего документа есть подпись контрагента, для исходящего — есть подпись с нашей стороны, документ отправлен.

Документ подписан с обеих сторон

Есть ответная подпись.

[1] Выбор режима доступен при повторном запуске инсталлятора

[2] Должен быть установлен коннектор к оператору ЭДО.

Руководство пользователя модуля при работе в Windows-клиенте

Аннотация

Данный раздел описывает работу с модулем через интерфейс Windows-клиента.

Раздел содержит следующие подразделы:

- **Работа с неформализованными документами** — операции с неформализованными документами:
 - **"Отправка с подписанием"** — создание нового неформализованного документа с отправкой и подписанием,
 - **"Подписание полученных"** — подписание ЭП документа, полученного от оператора ЭДО,
 - **"Аннулирование"** — признание документа недействительным,
 - **"Подтверждение аннулирования"** — подтверждение запроса на аннулирование, полученного от документа ЭДО.
- **Работа с формализованными документами:**
 - **Карточки формализованных документов** — описание системных карточек формализованных документов,
 - **Исходящие** — работа с исходящими формализованными документами,
 - Работа с входящими формализованными документами: **Подписание входящего** и **Запрос на уточнение**.
 - **Подтверждение извещения** — работа с входящими и исходящими извещениями,
 - **Аннулирование** — процесс аннулирования документов,
 - **Подтверждение аннулирования** — подтверждение или отказ от аннулирования документа,
 - **Печатная форма** — использование печатной формы формализованного документа.
- **Групповые операции** с неформализованными документами.
- **Журнал обмена ЮЗДО** — системный журнал, содержащий историю электронного обмена по данному документу с оператором ЭДО.

- **Работа с приглашениями к обмену** — приглашения контрагента к началу обмена электронными документами.

Работа с неформализованными документами

В данном разделе рассмотрены основные этапы работы пользователя с неформализованными входящими и исходящими документами.

Для всех документов поддерживаются подтверждения оператора и извещения о получении участников обмена формата 1.03, в соответствии с приказом ФНС России от 21.02.2023 № ЕД-7-26/133@.

Изменения затрагивают следующие технологические документы:

- Извещения о получении документа (DP_IZVPOL).
- Уведомления об уточнении (DP_UVUTOCH).
- Подтверждения даты отправки (DP_PDOTPR).
- Подтверждения даты получения (DP_PDPOL).

Для работы с неформализованными документами не предусмотрено специальных видов карточек. Настройка карточек для работы с неформализованными документами осуществляется администратором, поэтому интерфейс карточек и названия используемых элементов управления могут отличаться от приведенных в данном разделе.

Отправка подписанного неформализованного документа

Следующая инструкция описывает отправку неформализованного документа, подписанного на предыдущих этапах работы с документом (например, при согласовании). Если документ не подписан, он может быть подписан текущим сотрудником при отправке.

Для отправки неформализованного документа без подписания:

1. Откройте отправляемый документ.

В документе должны быть указаны *Получатели* — контрагенты, которым будет отправлен документ.

2. Выберите способ отправки документа^[1]:
 - **С выбором отправляемых файлов.**

- а. Нажмите кнопку **Отправка документов с выбором файлов** на ленте карточки.

Будет открыто окно отправки документа с выбором файлов.

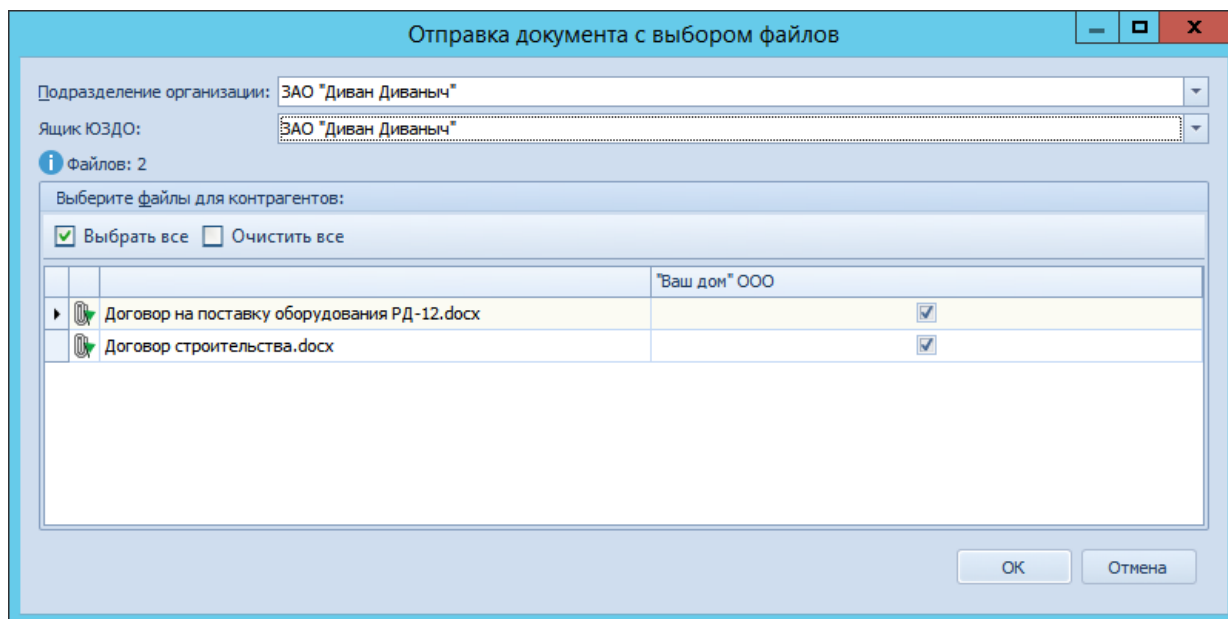


Рисунок 32. Окно отправки документа с выбором файлов

- б. В поле *Подразделение организации* выберите организацию (или подразделение), от имени которой отправляется документ.

В списке приведены организации, загруженные в *Справочник настроек операторов ЮЗДО* из системы оператора ЭДО.

- с. В поле *Ящик ЮЗДО* выберите ящик электронного обмена организации, через который будет отправлен электронный документ.

В списке приведены ящики электронного обмена, зарегистрированные для выбранного подразделения организации.

- д. Выберите отправляемые контрагентам файлы, установив или сняв флажки в списке файлов для каждого получателя (контрагента). Файлы будут отправлены только контрагентам, для которых они выбраны.

Если отправка документов контрагенту от выбранного подразделения организации невозможна, колонка выбора отправляемых файлов для данного контрагента будет заблокирована (флажки сняты со всех файлов).

Если в списке есть файлы, неподписанные квалифицированной ЭП, они

будут выделены красным цветом.

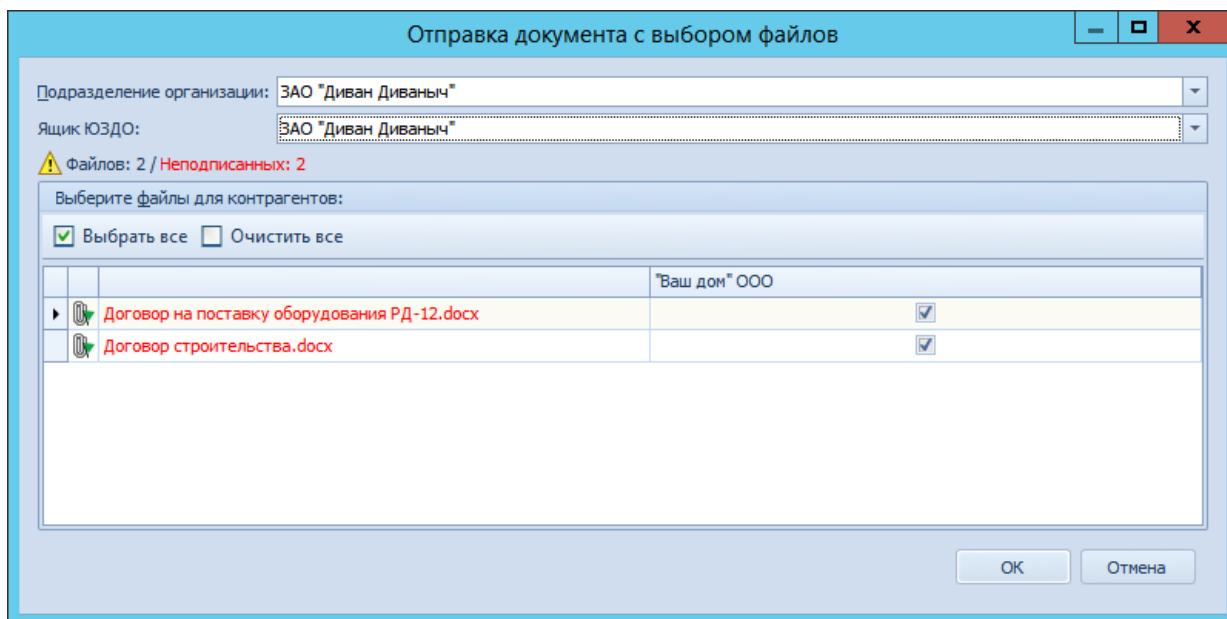


Рисунок 33. Окно отправки документа с выбором файлов с неподписанными файлами

- **Без выбора отправляемых файлов.**



В данном режиме отправляются только основные файлы документа. Дополнительные файлы не отправляются.

- а. Нажмите кнопку **Отправка документов** на ленте карточки. Будет открыто окно подтверждения отправки документа.

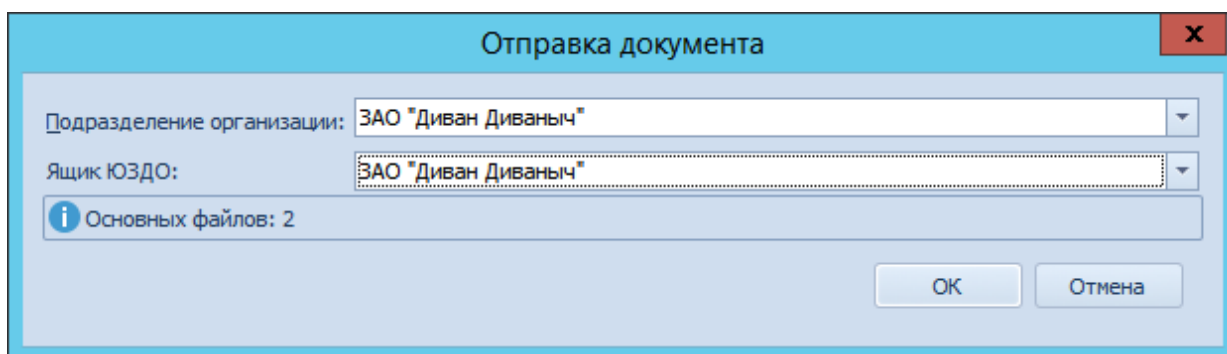


Рисунок 34. Окно отправки документа

- б. В поле *Подразделение организации* выберите организацию (или подразделение), от имени которой отправляется документ.
- с. В поле *Ящик ЮЗДО* выберите ящик электронного обмена организации, через который будет отправлен электронный документ.

Если в списке есть файлы, неподписанные квалифицированной ЭП,

данная информация будет указана в окне отправки.

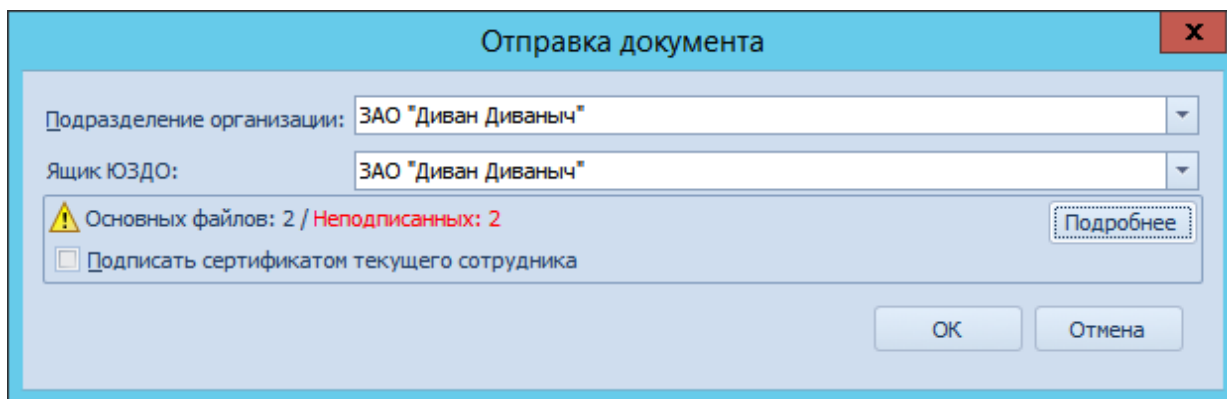


Рисунок 35. Окно отправки документа с неподписанными файлами

Для просмотра списка неподписанных файлов нажмите кнопку **Подробнее**. Для подписания файлов сертификатом текущего сотрудника установите флаг **Подписать сертификатом текущего сотрудника**.

3. Нажмите кнопку **ОК** в окне отправки. После формирования документа для отправки оператору ЭДО будет показано сообщение **Отправка успешно выполнена**.

Если в списке отправляемых файлов есть неподписанные файлы будет предложено выбрать сертификат для их подписания. Выберите сертификат квалифицированной подписи и нажмите кнопку **ОК**. Подписи **всех** отправляемых файлов будут переформированы.

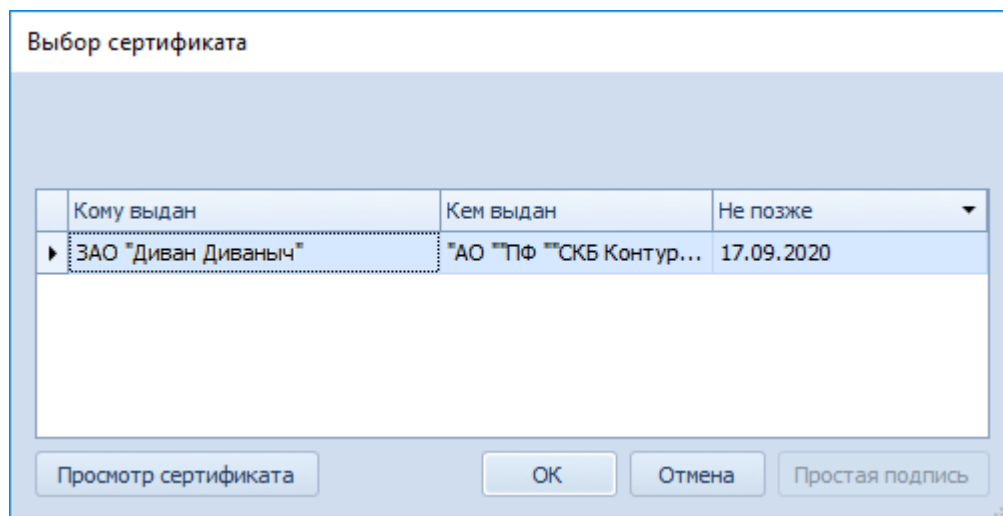


Рисунок 36. Выбор сертификата электронной подписи

Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или

контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

+ Документы будут отправлены контрагентам на подпись в течении некоторого времени. Состояние отправки документов можно отслеживать с помощью [Журнала обмена ЮЗДО](#).

Отправка неформализованного документа с подписанием

Следующая инструкция описывает отправку неформализованного документа с наложением ЭП на отправляемые файлы документа.

Если отправляемые документы были подписаны квалифицированной подписью на предыдущих этапах работы с документом (например, при согласовании), при отправке с подписанием существующая ЭП будет перезаписана. Для отправки документа с сохранением существующей ЭП следуйте инструкции [Отправка подписанного неформализованного документа](#).

Чтобы отправить неформализованный документ с подписанием:

1. Откройте отправляемый документ.

В документе должны быть указаны *Получатели* — контрагенты, которым будет отправлен документ.

2. Выберите способ отправки документа^[2]:

- **С выбором отправляемых файлов.**

- а. Нажмите кнопку **Отправка документов с подписанием и выбором файлов** на ленте карточки. Будет открыто окно отправки документа с подписанием и выбором файлов.

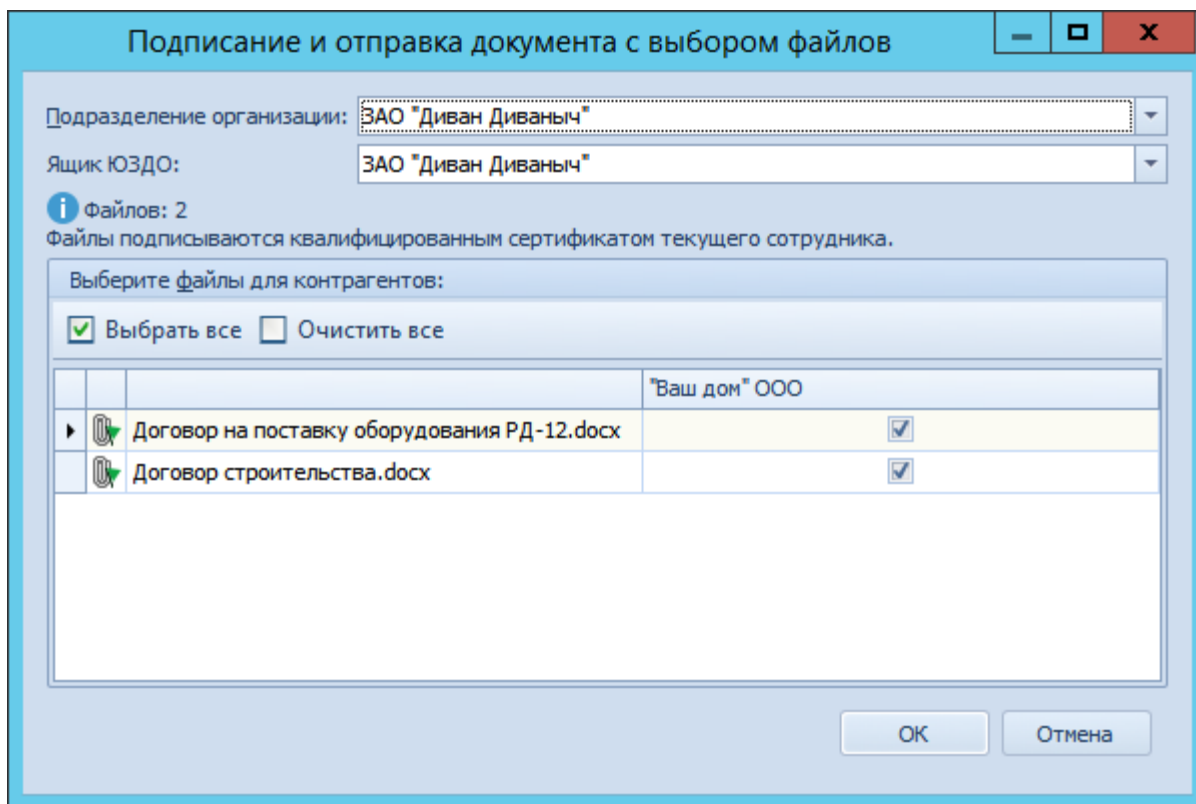


Рисунок 37. Окно подписания и отправки документа с выбором файлов



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- b. В поле *Подразделение организации* выберите организацию (или подразделение), от имени которой отправляется документ.
- c. В поле *Ящик ЮЗДО* выберите ящик электронного обмена организации, через который будет отправлен электронный документ.
- d. Выберите отправляемые контрагентам файлы, установив или сняв флажки в списке файлов для каждого получателя (контрагента).

См. дополнительную информация по указанным настройкам в пункте [Отправка подписанного неформализованного документа](#).

- **Без выбора отправляемых файлов.**



В данном режиме отправляются только основные файлы документа. Дополнительные файлы не отправляются.

- а. Нажмите кнопку **Отправка документов с подписанием** на ленте карточки.

Будет открыто окно отправки документа с подписанием.

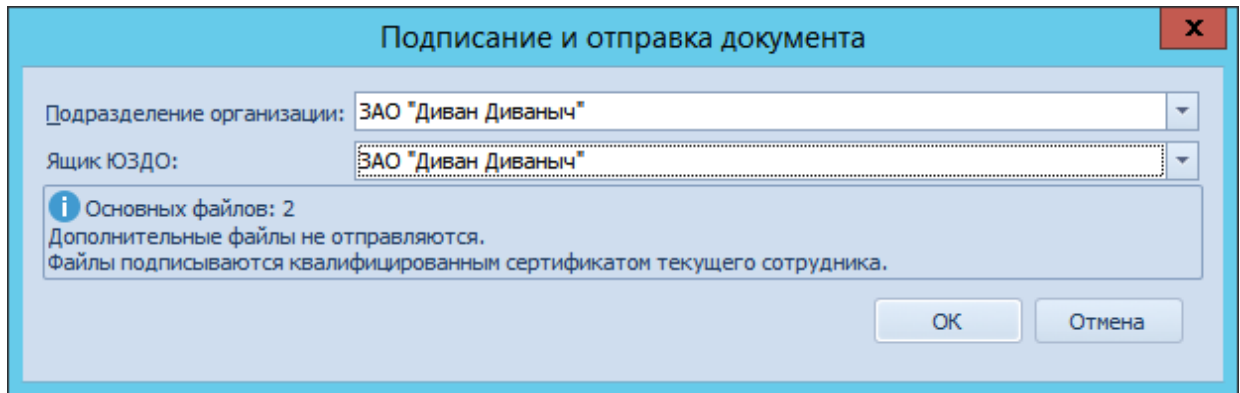


Рисунок 38. Окно подписания и отправки документа

- б. В поле *Подразделение организации* выберите организацию или подразделение, от имени которой отправляется документ.
 - с. В поле *Ящик ЮЗДО* выберите ящик электронного обмена организации, через который будет отправлен электронный документ.
3. Нажмите кнопку **ОК** в окне подписания и отправки. Будет открыто окно выбора сертификата для подписания отправляемых файлов.

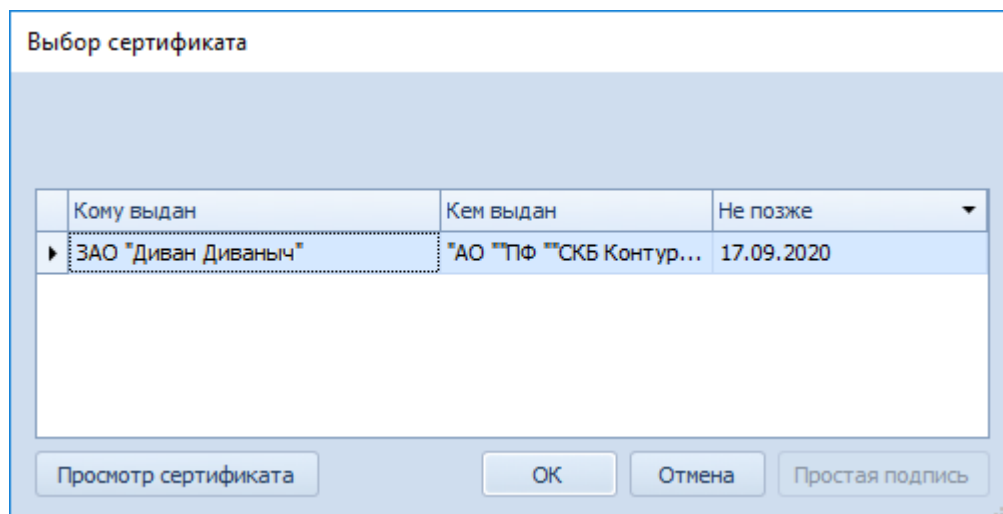


Рисунок 39. Выбор сертификата электронной подписи

4. Выберите сертификат квалифицированной подписи и нажмите кнопку **ОК**. После выполнения подписания и формирования документа для отправки оператору ЭДО будет показано сообщение **Отправка успешно выполнена**.

Документ будет отправлен контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). После этого статус документа сменится на **Отправлен контрагенту**.

Контрагент может подписать документ или отказать в подписании. При подписании статус документа сменится на **Получена подпись от контрагента**, при отказе — на **Получен отказ от контрагента**. В случае отказа, причина будет указана в *Журнале обмена ЮЗДО*. Обмен информацией по данному документу можно отслеживать с помощью **Журнала обмена ЮЗДО**.

Подписание полученного неформализованного документа

Входящий неформализованный документ должен быть подписан, если данное требование установлено контрагентом. Организация может подписать документ или отказать в подписании.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

Чтобы подписать входящий неформализованный документ:

1. Откройте документ со статусом **Получен на подпись от контрагента**.
2. Нажмите кнопку на ленте карточки для принятия решения по документу:
 - **Подписание документа** — для подписания документа. При этом нужно будет выбрать сертификат квалифицированной подписи, которым будут подписаны файлы документа.

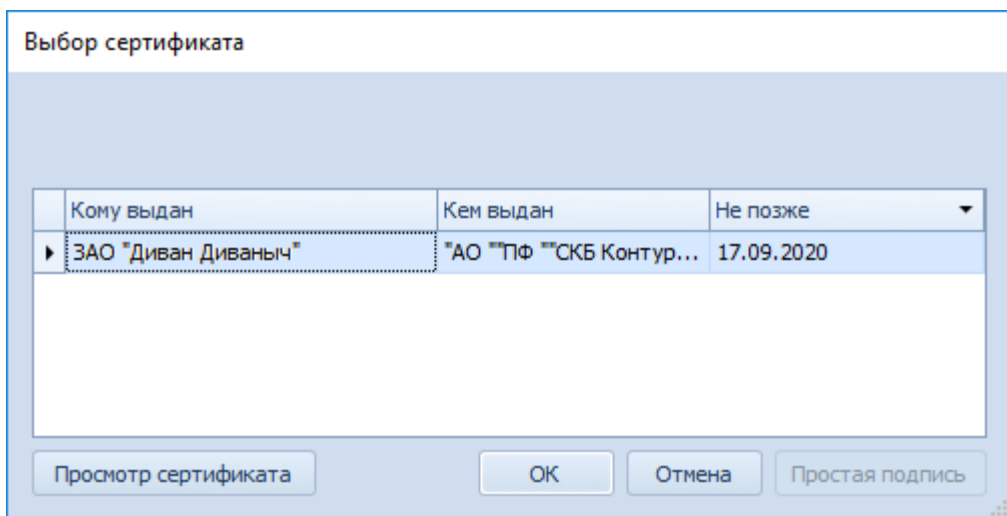


Рисунок 40. Выбор сертификата электронной подписи

- **Отказ в подписании документа** — для отказа от подписания полученного документа. При этом нужно будет указать причину отказа.

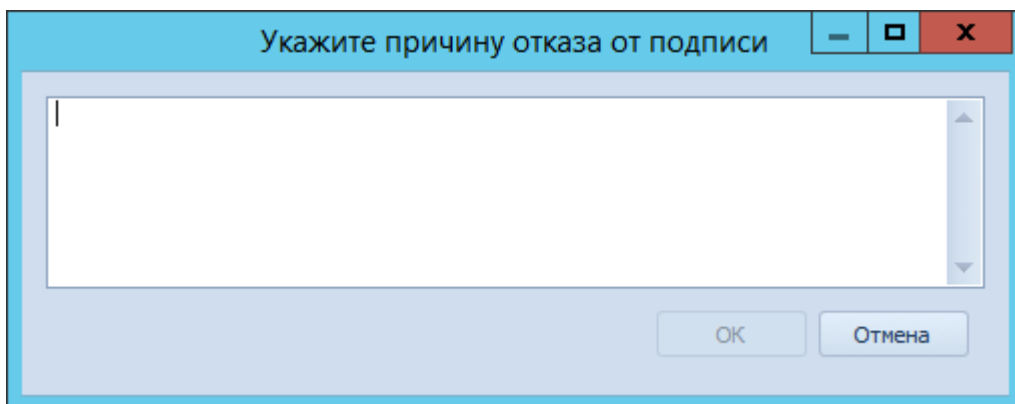


Рисунок 41. Указание причины отказа от подписания документа

Ответная подпись (или отказ) будет отправлена контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). После этого статус документа сменится на **Отправлена ответная подпись контрагенту** или **Отправлен отказ от подписи контрагенту** соответственно.

Аннулирование неформализованного документа

Аннулирование документа может быть инициировано любой из сторон ЭДО.

Для аннулирования неформализованного документа по инициативе организации:

1. Откройте аннулируемый документ.
2. Нажмите кнопку **Запросить аннулирование** на ленте карточки. Будет предложено указать причину аннулирования документа.

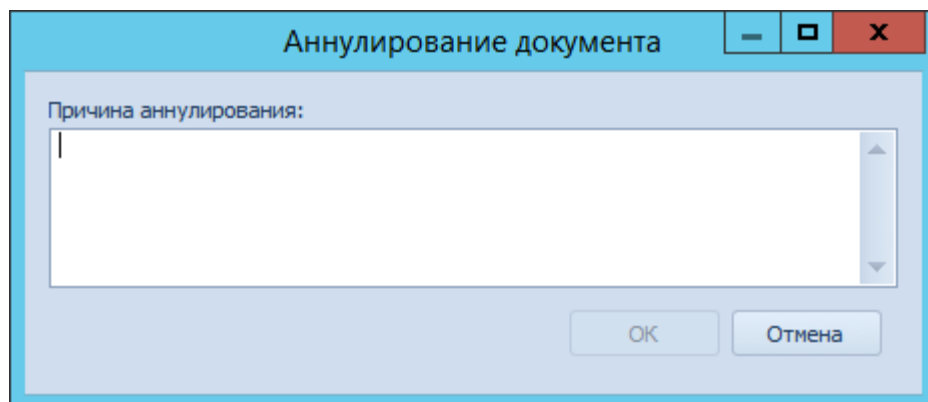


Рисунок 42. Указание причины аннулирования документа

3. Укажите причину аннулирования и нажмите кнопку **ОК**. Будет предложено выбрать сертификат для подписания сообщения об аннулировании.
4. Выберите квалифицированный сертификат и нажмите **ОК** для отправки запроса на аннулирование.

Запрос аннулирования будет отправлен контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). После этого статус документа сменится на **Отправлен запрос на аннулирование**.

Исходящий документ, который не был подписан контрагентом, будет аннулирован в одностороннем порядке. В других случаях контрагент должен подтвердить аннулирование.

Статус документа, аннулирование которого подтверждено, сменится на **Аннулирован**. Если в аннулировании было отказано (причина отказа будет указана в *Журнале обмена ЮЗДО*), статус обмена сменится на:

- a. **Получена подпись от контрагента** или **Отправлен контрагенту** (если для данного документа не требовалось подписание) — для исходящего документа.
- b. **Отправлена ответная подпись контрагенту** или **Получен от контрагента** (если для данного документа не требовалось подписание) — для входящего документа.

Подтверждение аннулирования неформализованного документа

Если аннулирование инициировано контрагентом, организация может подтвердить аннулирование или отклонить.

1. Откройте документ со статусом обмена **Получен запрос на аннулирование** (в *Журнале обмена ЮЗДО*).

2. Нажмите кнопку на ленте карточки для принятия решения по документу:

- **Подтвердить аннулирование** — для подтверждения аннулирования. При этом нужно будет выбрать сертификат квалифицированной подписи для формирования ответной подписи.

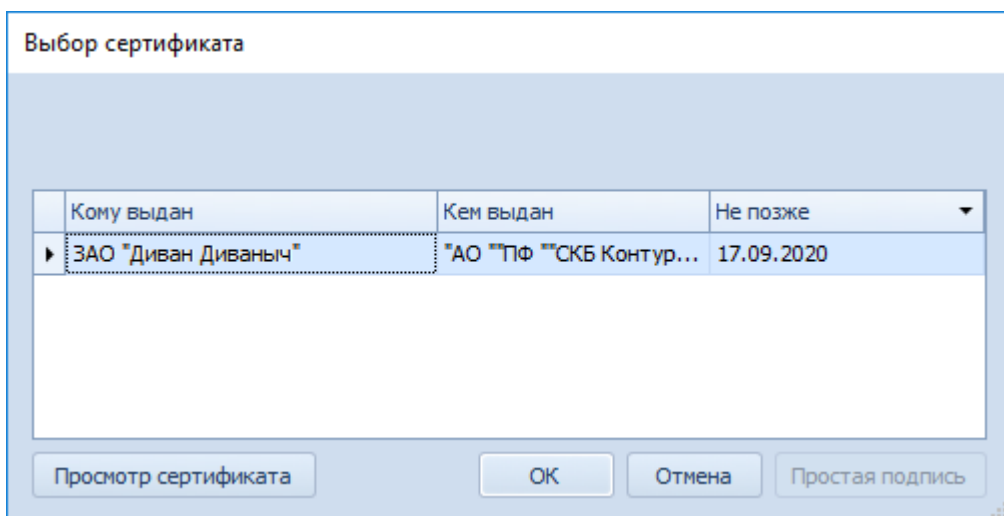


Рисунок 43. Выбор сертификата электронной подписи

- **Отказать в аннулировании** — для отказа в аннулировании. При этом нужно будет указать причину отказа.

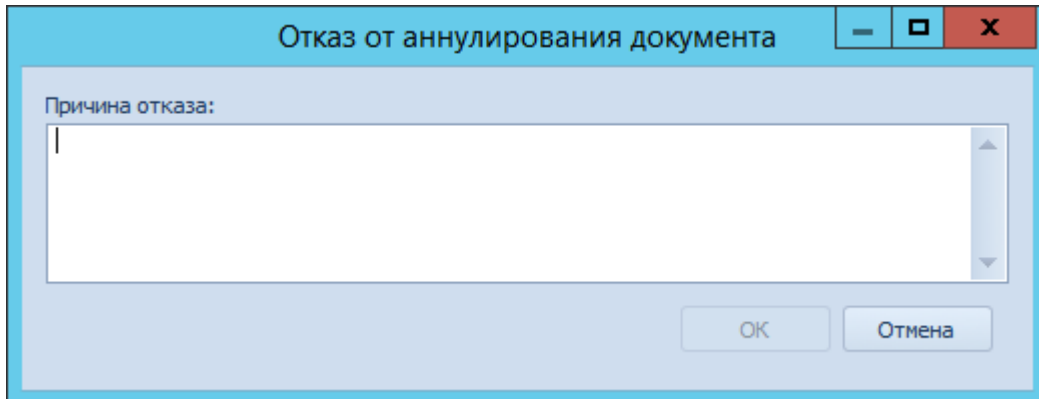


Рисунок 44. Указание причины отказа в аннулировании документа

Подтверждение (или отказ) аннулирования будет отправлено контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). После этого, при подтверждении аннулирования, статус документа сменится на **Аннулирован**. При отказе в аннулировании статус документа сменится на:

- **Отправлен контрагенту** — для исходящего без запроса подписи.
- **Получена подпись от контрагента** — для исходящего с запросом подписи.
- **Получен от контрагента** — для входящего без запроса подписи.

- **Отправлена ответная подпись контрагенту** — для входящего с запросом подписи.

Работа с формализованными документами

В данном разделе рассмотрена работа пользователя с формализованными входящими и исходящими документами, а также приведено описание карточек для работы с формализованными документами.

Карточки Docsvision для работы с формализованными документами

Для работы с формализованными документами в модуле предусмотрено несколько специальных видов карточек, производных от вида *Документ*:

1. *Исходящий УПД* — предназначен для создания исходящих УПД.
 - a. *Исходящий УКД* — предназначен для создания корректировок к отправленным УПД.
 - b. *Исходящее исправление УПД* — предназначен для формирования исправлений отправленных УПД.
 - i. *Исходящее исправление УПД (820)* — предназначен для формирования исправлений для отправленных УПД в формате, вступившем в силу в 2020 году.
 - c. *Исходящий УПД (820)* — предназначен для создания исходящих УПД в формате, вступившем в силу в 2020 году.
 - d. *Исходящий акт (552)* — предназначен для создания исходящих актов в формате 552.
2. *Входящий УПД* — предназначен для работы с входящими УПД. Карточки данного вида создаются модулем при получении формализованных документов.
 - a. *Входящий УКД* — предназначен для работы с корректировками полученных УПД.
 - b. *Входящий УПД (820)* — предназначен для работы с корректировками полученных УПД в формате, вступившем в силу в 2020 году.
 - c. *Формализованный акт* — предназначен для работы с входящими актами и формализованными накладными в рекомендованном ФНС формате.
 - i. *Формализованный акт (820)* — предназначен для работы с входящими актами и формализованными накладными в рекомендованном ФНС

формате, вступившем в силу в 2020 году.

- ii. *Входящий акт (552)* — предназначен для работы с входящими актами в формате 552.

Для всех документов поддерживаются подтверждения оператора и извещения о получении участников обмена формата 1.03, в соответствии с приказом ФНС России от 21.02.2023 № ЕД-7-26/133@.

Изменения затрагивают следующие технологические документы:

- Извещения о получении документа (**DP_IZVPOL**).
- Уведомления об уточнении (**DP_UVUTOCH**).
- Подтверждения даты отправки (**DP_PDOTPR**).
- Подтверждения даты получения (**DP_PDPOL**).

Исходящий УПД

Карточки вида *Исходящий УПД* и *Исходящий УПД (820)* предназначены для передачи контрагентам счетов-фактур, накладных, актов и УПД. Пример карточки приведен на следующем рисунке.

Исходящий УПД - Документ

Документ

Сохранить Подписать и отправить Заполнить данные из файла Печать Журнал подписей

Общие Подписи и шифр...

ON_SCHFDOPPR_xxxx_2BM-9660026731-966001000-201909240135490337231_20191021_e1656df1-f998-45f0-a5e9-e9e2ea5f83f6.xml

Состояние документа: На подготовке Функция УПД: ДОП

Статус УПД: В документе отсутствует подписанная форма исходящего УПД.

Основная информация Печатная форма Журнал обмена ЮЗДО Ссылки

Номер документа: 1212 № Дата документа: 10.10.2019

Стоимость с НДС: 696 000, Сумма НДС: 116 000,

Контрагент: Кравчук Владимир Владимирович Организация контрагента: "Солнечный берег" ООО

Описание:

Данные о подписанте

Подписант: Быков М. А. Сведения о подписанте: Юридическое лицо

Организация подписанта: "Диван Диваныч" ЗАО

Область полномочий: Лицо, ответственное за подписани... Статус: Работник организации продавца товаров (работ, услуг, имущест...

Основание полномочий: Должностные обязанности Основание полномочий организации: Доверенность №121

Иные сведения:

Папки\ЮЗДО

Рисунок 45. Пример исходящего УПД

Карточка содержит следующие элементы:

1. Лента карточки — может содержать кнопки (в зависимости от статуса обмена):
 - **Сохранить.**
 - **Подписать и отправить.** При нажатии будет вызвана функция подписания документа с отправкой контрагенту. Кнопка доступна, если документ не был отправлен.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- **Заполнить данные из файла.** При нажатии поля карточки с информацией о документе будут заполнены данными из приложенного файла XML. Кнопка доступна, если документ не был отправлен.
 - **Печать.** При нажатии будет открыто окно экспорта и печати, из которого можно выполнить печать формализованного документа.
 - **Запросить аннулирование.** При нажатии будет вызвана функция аннулирования формализованного документа.
 - **Подтвердить аннулирование.** При нажатии будет подтвержден поступивший запрос на аннулирование.
2. Список основных файлов — предназначен для добавления XML-файла счета-фактуры, акта, накладной или УПД.
- Кнопка **Журнал подписей.** При нажатии будет открыт *Журнал подписей* с информацией об электронных подписях документа.
3. О документе:
- *Состояние документа* — текущее состояние карточки *Документ*.
 - *Статус УПД* — текущее состояние подписания УПД:
 - **УПД подписан отправителем** — если сформирован и подписан только титул продавца.
 - **УПД подписан с обеих сторон** — если сформирован и подписан титул продавца и получен ответный подписанный титул покупателя.
- Результат проверки подписи отображается справа от состояния подписания.
- *Функция УПД* — **формат УПД**, может принимать значения: **СЧФ**, **ДОП** и **СЧФДОП**.
4. Вкладки:
- *Основная информация.* В данном разделе карточки указываются данные покупателя, а также заполняется *титул продавца*.
 - *Печатная форма*, см. пункт **Печатная форма формализованного документа**.

- Журнал обмена ЮЗДО, см. пункт [Журнал обмена ЮЗДО](#).
- Ссылки — приводятся ссылки на полученные документы, являющиеся для данного документа исправлением или корректировкой, а также ссылка на исправляемый или корректируемый документ, ссылки на другие документы, полученные в одном пакете с данным документом.

Исходящий УКД

Карточки вида *Исходящий УКД* предназначены для корректировки сумм и количества товаров в переданном УПД. Пример карточки приведен на следующем рисунке.

The screenshot shows a software window titled "Корректировочный УПД - Документ". The interface includes a top toolbar with icons for saving, signing, filling data, printing, and a journal. Below the toolbar, there's a status bar showing the document's state as "На подготовке" and the function as "КСЧФ". The main area is divided into tabs: "Основная информация", "Печатная форма", "Журнал обмена ЮЗДО", and "Ссылки". The "Основная информация" tab is active, displaying fields for document number (1212), correction number (1213), dates (10.10.2019 and 22.10.2019), amounts (816 000 and 136 000), and parties (Кравчук Владимир Владимирович and "Солнечный берег" ООО). A section for the signatory details is also present, showing the signatory as "Быков М. А." and the organization as "Диван Диваныч" ЗАО, with a status of "Работник организации продавца товаров (работ, услуг, имущест...)".

Рисунок 46. Пример исходящего УКД

Список элементов карточки *Исходящий УКД* аналогичен списку элементов карточки *Исходящий УПД*. Функция УПД может принимать значения: **КСЧФ, ДИС**

и **КСЧФДИС**.

Исходящее исправление УПД

Карточки вида *Исходящее исправление УПД* и *Исходящее исправление УПД (820)* предназначены для замены переданного УПД. Пример карточки приведен на следующем рисунке.

Рисунок 47. Пример исходящего исправления УПД

Список элементов карточки *Исходящее исправление УПД* аналогичен списку элементов карточки *Исходящий УПД*.

Входящий УПД

Карточки вида *Входящий УПД* и *Входящий УПД (820)* предназначены для загрузки в Docsvision счетов-фактур, УПД и исправлений к счетам-фактурам и

УПД. Пример карточки приведен на следующем рисунке.

УПД № 2234 от 24.10.2019 - Документ

Документ

Сохранить Печать Подписать документ Оказать в подписании Запросить уточнение Журнал подписей

Общие Действия Подписи и шифр...

Номер документа: 2234 Дата документа: 24.10.2019 Тип документа: УПД

Статус УПД: УПД подписан отправителем. Функция УПД: доп

Основная информация Подписание Журнал обмена ЮЗДО Печатная форма

Контрагент: "Солнечный берег" ООО Ответственное подразделение: "Диван Диваныч" ЗАО

Сумма НДС: 200 000, Сумма документа с НДС: 2 200 000,

Описание:

Ссылки:

Тип ссылки	Автор	Дата создания	Ссылка

Рисунок 48. Пример входящего УПД

Карточка содержит следующие элементы:

1. Лента карточки — может содержать кнопки (в зависимости от статуса обмена):
 - **Сохранить.**
 - **Печать.** При нажатии будет открыто окно экспорта и печати, из которого можно выполнить печать печатной формы формализованного документа.
 - **Подписать документ.** При нажатии будет вызвана функция подписания и отправки титула покупателя.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть

задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- **Отказать в подписании.** При нажатии будет вызвана функция отказа в подписании с указанием причины отказа.
- **Запросить аннулирование.** При нажатии будет вызвана функция аннулирования формализованного документа.
- **Подтвердить аннулирование.** При нажатии будет подтвержден поступивший запрос на аннулирование.
- **Отказать в аннулировании.** При нажатии будет вызвана функция отказа от аннулирования по поступившему запросу.
- **Запросить уточнение.** При нажатии будет вызвана функция запроса исправления документа с указанием комментария.
- **Журнал подписей.** При нажатии будет открыт *Журнал подписей* с информацией об электронных подписях документа.

2. Раздел с информацией об УПД:

- *Номер документа.*
- *Дата документа.*
- *Тип документа* — название типа формализованного документа, указанного в полученном электронном документе.
- *Статус УПД* — текущее состояние подписания УПД:
 - **УПД подписан отправителем** — если сформирован и подписан только титул продавца.
 - **УПД подписан с обеих сторон** — если сформирован и подписан титул продавца и получен ответный подписанный титул покупателя.

Результат проверки подписи отображается справа от состояния подписания.

- *Функция УПД* — **формат УПД**, может принимать значения: **СЧФ**, **ДОП** и **СЧФДОП**.

3. Вкладки:

- *Основная информация.* В данном разделе карточки указываются продавец и сумма документа.
- *Подписание.* В данном разделе заполняется титул покупателя и данные получателя.
- *Журнал обмена ЮЗДО,* см. пункт [Журнал обмена ЮЗДО](#).
- *Печатная форма,* см. пункт [Печатная форма формализованного документа](#).
- Ссылки — приводятся ссылки на полученные документы, являющиеся для данного документа исправлением или корректировкой, а также ссылка на исправляемый или корректируемый документ, ссылки на другие документы, полученные в одном пакете с данным документом.

Входящий УКД

Карточки вида *Входящий УКД* предназначены для загрузки в Docsvision исправлений к входящим УПД. Пример карточки приведен на следующем рисунке.

УКД № 1000 от 05.11.2019 - Документ

Документ

Сохранить Печать Подписать документ Оказать в подписании Запросить уточнение Журнал подписей

Общие Действия Подписи и шифр...

Номер документа: 1000 Дата документа: 05.11.2019 Тип документа: УКД

Статус УПД: УПД подписан отправителем. Функция УПД: КСЧФ

Основная информация Подписание Журнал обмена ЮЗДО Печатная форма

Контрагент: "Солнечный берег" ООО Ответственное подразделение: "Диван Диваныч" ЗАО

Сумма НДС: 1 800 000, Сумма документа с НДС: 10 800 000,

Описание:

Ссылки:

Тип ссылки	Автор	Дата создания	Ссылка
Связано с (ЮЗДО) (Обратная: Ссылается на (ЮЗДО))	WorkflowService	05.11.2019 16:01:45	Карточка УПД № 999 от 05.11.2019'

Рисунок 49. Пример входящего УКД

Список элементов карточки *Исходящий УКД* аналогичен списку элементов карточки *Входящий УПД*. Функция УПД может принимать значения: **КСЧФ**, **ДИС** и **КСЧФДИС**. В разделе *Ссылки* содержится ссылка на *Входящий УПД* или на исправление, по которому поступила данная корректировка.

Формализованный акт

Карточки вида *Формализованный акт* и *Формализованный акт (820)* предназначены для загрузки в Docsvision актов и накладных. Пример карточки приведен на следующем рисунке.

УПД № 8889 от 25.10.2019 - Документ

Документ

Сохранить Печать Подписать документ Оказать в подписании Журнал подписей

Общие Действия Подписи и шифр...

Номер документа: 8889 Дата документа: 25.10.2019 Тип документа: Формализованный акт

Статус УПД: УПД подписан отправителем. Функция УПД: ДОП

Основная информация Подписание Журнал обмена ЮЗДО Печатная форма

Контрагент: "Солнечный берег" ООО Ответственное подразделение: "Диван Диваныч" ЗАО

Сумма НДС: 640 000, Сумма документа с НДС: 3 840 000,

Описание:

Ссылки:

Тип ссылки	Автор	Дата создания	Ссылка

Рисунок 50. Пример входящего формализованного акта

Карточка содержит следующие элементы:

1. Лента карточки — в зависимости от статуса обмена может содержать кнопки:
 - а. **Сохранить**.
 - б. **Печать**. При нажатии будет вызвана функция экспорта и печати карточки.
 - с. **Подписать документ**. При нажатии будет вызвана функция подписания и отправки титула покупателя.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов

проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- d. **Отказать в подписании.** При нажатии будет вызвана функция отказа в подписании с указанием причины отказа.
 - e. **Запросить аннулирование.** При нажатии будет вызвана функция аннулирования формализованного документа.
 - f. **Подтвердить аннулирование.** При нажатии будет подтвержден поступивший запрос на аннулирование.
 - g. **Отказать в аннулировании.** При нажатии будет вызвана функция отказа от аннулирования по поступившему запросу.
 - h. **Журнал подписей.** При нажатии будет открыт *Журнал подписей* с информацией об электронных подписях документа.
2. Раздел с информацией об УПД:
- a. *Номер документа.*
 - b. *Дата документа.*
 - c. *Тип документа* — название типа формализованного документа, указанного в полученном электронном документе.
 - d. *Статус УПД* — текущее состояние подписания УПД:
 - i. **УПД подписан отправителем** — если сформирован и подписан только титул продавца.
 - ii. **УПД подписан с обеих сторон** — если сформирован и подписан титул продавца и получен ответный подписанный титул покупателя.

Результат проверки подписи отображается справа от состояния подписания.
 - e. *Функция УПД — формат УПД.* Для *Формализованного акта* принимает значение: **доп.**
3. Вкладки:
- a. *Основная информация.* В данном разделе карточки указываются продавец и сумма документа.
 - b. *Подписание.* В данном разделе заполняется титул покупателя и данные получателя.

- с. *Журнал обмена ЮЗДО*, см. пункт [Журнал обмена ЮЗДО](#).
- d. *Печатная форма*, см. пункт [Печатная форма формализованного документа](#).
- е. Ссылки — приводятся ссылки на полученные документы, являющиеся для данного документа исправлением или корректировкой, а также ссылка на исправляемый или корректируемый документ, ссылки на другие документы, полученные в одном пакете с данным документом.

Работа с исходящими документами

Работа с исходящими документами включает следующие этапы:

1. Пользователь модуля (далее — *Продавец*) создаёт исходящий документ вида (или подвида) *Исходящий УПД* и отправляет его контрагенту (далее — *Покупатель*).
2. Оператор ЭДО принимает документ от *Продавца* и отправляет ему Извещение о получении, а также передаёт документ Покупателю.
3. Продавец получает Извещение о получении от оператора ЭДО и подтверждает его получение. Подтверждение не требуется для актов и накладных.
4. После получения документа Покупатель может:
 - Подписать или отказать в подписании документа — не требуется для счёт-фактуры.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- Запросить аннулирование — для счёт-фактуры.
 - Запросить уточнение у *Продавца*.
5. *Продавец* получает ответ от *Покупателя*.

Если "Покупатель" подписал документ (был сформирован титул покупателя):

- a. Титул покупателя добавляется в карточку *Исходящего УПД* (или подвида).

б. Карточка и статус обмена переходят в состояние **Получена подпись от контрагента**.

с. Документооборот завершен.

- Если *Покупатель* отказал в подписании документа, карточка и статус обмена переходят в состояние "Получен отказ от контрагента".
- Если *Покупатель* запросил уточнение, статус обмена переходит в состояние **Требуется подтверждение получения**, а далее в **Требуется уточнение**. После этого работа с данным документом завершается. Пользователь модуля может создать корректировочный УПД или исправление УПД.

6. *Покупатель* или *Продавец* могут запросить аннулирование подписанного документа. Вторая сторона ЭДО может подтвердить или отказать в аннулировании.

Если *Продавец* запросил аннулирование до получения извещения от оператора ЭДО (о получении документа от *Продавца*), документ будет аннулирован в одностороннем порядке (без подтверждения со стороны *Покупателя*).

Отправка УПД

Отправка УПД осуществляется с помощью карточки вида *Исходящий УПД*. Для создания исходящего УПД нужен XML файл сформированной **счет-фактуры, акта, накладной** или **УПД**.

1. Создайте новую карточку вида **Документ/УПД/Исходящий УПД**.

Рисунок 51. Карточка Исходящий УПД



С 1 января 2020 года согласно [приказу ФНС РФ от 19.12.2018 N ММВ-7-15/820@](#) в силу вступает новый формат УПД. С указанной даты вместо карточек *Исходящий УПД* нужно создавать карточки вида *Документ/УПД/Исходящий УПД/Исходящий УПД (820)*.

2. Добавьте в карточку файл отправляемого универсального передаточного документа в формате XML:
 - Перетащите файл с файловой системы в область *Добавьте файл...* карточки.
 - Вызовите команду **Добавить файл с файловой системы > Основной** из контекстного меню области *Добавьте файл...*, и выберите файл из файловой системы.

3. Нажмите кнопку **Заполнить данные из файла** на ленте карточки. Поля *Номер документа*, *Дата документа*, *Стоимость с НДС* и *Сумма НДС* будут заполнены автоматически. Если указанный в файле контрагент зарегистрирован в *Справочнике контрагентов* (должен совпадать КПП/ИНН), поле *Организация контрагента* также будет заполнено.
4. Выберите сотрудника контрагента в поле *Контрагент*. Убедитесь, что в поле *Организация контрагента* указан корректный получатель данного электронного документа.
5. Заполните *титул продавца* в разделе *Данные о подписанте*.
6. Нажмите кнопку **Подписать и отправить** на ленте карточки. Будет открыто окно *Подписание и отправка*.

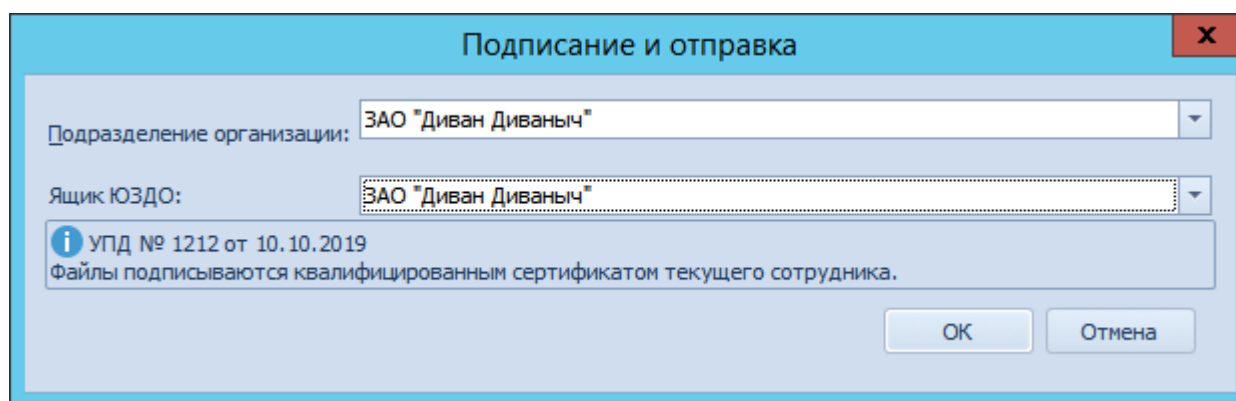


Рисунок 52. Подписание и отправка



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

7. При необходимости выберите другое подразделение/организацию и ящик ЮЗДО, через который будет отправлен электронный документ. Нажмите кнопку **ОК**. Будет открыто окно выбора сертификата для подписания документа.
8. Выберите сертификат квалифицированной подписи и нажмите кнопку **ОК**. Будет показано сообщение **Отправка успешно выполнена**. Состояние документа изменится на **Подписан и отправлен**.

После получения документа оператором ЭДО нужно будет [подтвердить получение извещения](#) о приёме документа.

После подписания документа контрагентом (формируется *титул покупателя*) документ перейдет в статус **УПД подписан с обеих сторон**. Статус обмена изменится на **Получена подпись от контрагента**. На этом документооборот считается завершенным.

Для счёт-фактуры подписание контрагентом не требуется. Документооборот будет завершен после [подтверждения получения извещения](#). Статус обмена: **Отправлен контрагенту**.

Обмен информацией по данному документу можно отслеживать с помощью [Журнала обмена ЮЗДО](#)

Корректировка УПД (создание УКД)

Корректировка цены или количества товара в отправленной УПД выполняется с помощью карточки вида *Исходящий УКД*. Для создания корректировочного документа в Docsvision нужен файл отправляемого универсального корректировочного документа в [формате XML](#).

1. Создайте новую карточку вида **Документ/УПД/Исходящий УПД/Исходящий УКД**.

Рисунок 53. Карточка Исходящий УКД

2. Добавьте в карточку файл отправляемого универсального корректировочного документа в формате XML:
 - Перетащите файл с файловой системы в область **Добавьте файл....**
 - Вызовите команду **Добавить файл с файловой системы > Основной** из контекстного меню области **Добавьте файл...**, и выберите файл из файловой системы.
3. Нажмите кнопку **Заполнить данные из файла** на ленте карточки.

Поля *Номер документа*, *Дата документа*, *Номер корректировки*, *Дата корректировки*, *Стоимость с НДС* и *Сумма НДС* будут заполнены автоматически. Если указанный в файле контрагент зарегистрирован в *Справочнике контрагентов* (должен совпадать КПП/ИНН), поле *Организация контрагента* также будет заполнено.

4. Выберите сотрудника контрагента в поле *Контрагент*. Убедитесь, что в поле *Организация контрагента* указан корректный получатель данного электронного документа.
5. Заполните *титул продавца* в разделе *Данные о подписанте*.
6. Нажмите кнопку **Подписать и отправить** на ленте карточки. Будет открыто окно **Подписание и отправка**.

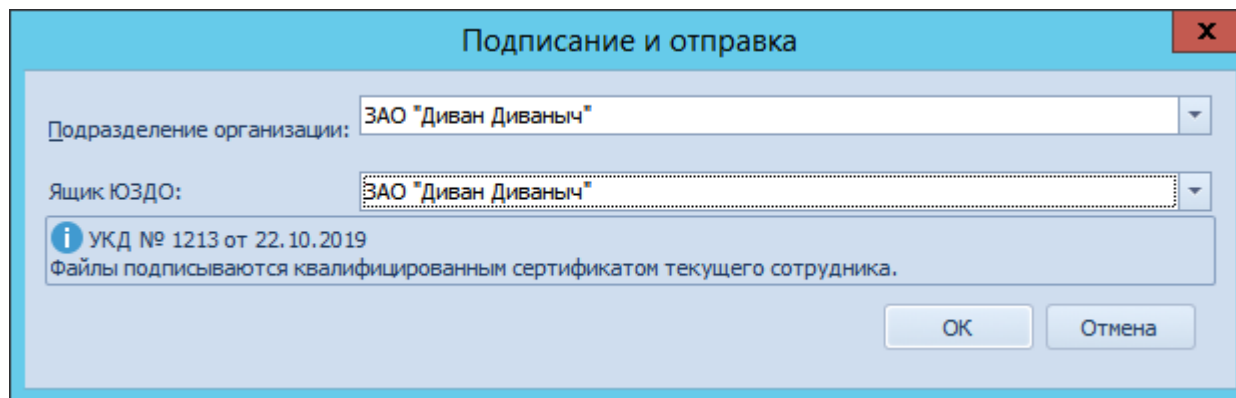


Рисунок 54. Подписание и отправка



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

7. При необходимости выберите другое подразделение/организацию и ящик ЮЗДО, через который будет отправлен электронный документ. Нажмите кнопку **ОК**. Будет открыто окно выбора сертификата для подписания документа.
8. Выберите сертификат квалифицированной подписи и нажмите кнопку **ОК**. Будет показано сообщение **Отправка успешно выполнена**. Состояние документа изменится на **Подписан и отправлен**.

После получения документа оператором обмена нужно будет подтвердить получение извещения о приёме документа. См. подробнее в пункте [Подтверждение получения извещения](#).

Исправление УПД

Если в отправленном УПД были выявлены ошибки (например, неверно указан адрес), нужно сформировать и отправить исправление УПД. Исправление выполняется с помощью карточки вида *Исходящее исправление УПД*. Для создания исправленного документа в Docsvision нужен файл отправляемого исправленного универсального передаточного документа в формате XML.

1. Создайте новую карточку вида *Документ/УПД/Исходящий УПД/Исходящее исправление УПД*.



С 1 января 2020 года согласно [приказу ФНС РФ от 19.12.2018 N ММВ-7-15/820@](#) в силу вступает новый формат УПД. С указанной даты вместо карточек *Исходящий УПД* нужно создавать карточки вида *Документ/УПД/Исходящий УПД/Исходящий УПД (820)*.

Рисунок 55. Карточка "Исходящее исправление УПД"

2. Добавьте в карточку файл отправляемого исправления универсального передаточного документа в формате XML:
 - Перетащите файл с файловой системы в область **Добавьте файл....**
 - Вызовите команду **Добавить файл с файловой системы > Основной** из контекстного меню области *Добавьте файл...*, и выберите файл из файловой системы.
3. Нажмите кнопку **Заполнить данные из файла** на ленте карточки. Поля *Номер документа*, *Дата документа*, *Номер исправления*, *Дата исправления*, *Стоимость с НДС* и *Сумма НДС* будут заполнены автоматически. Если указанный в файле контрагент зарегистрирован в *Справочнике контрагентов* (должен совпадать КПП/ИНН), поле *Организация контрагента* также будет заполнено.

4. Выберите сотрудника контрагента в поле *Контрагент*. Убедитесь, что в поле *Организация контрагента* указан корректный получатель данного электронного документа.
5. Заполните *титул продавца* в разделе *Данные о подписанте*.
6. Нажмите кнопку **Подписать и отправить** на ленте карточки. Будет открыто окно *Подписание и отправка*.

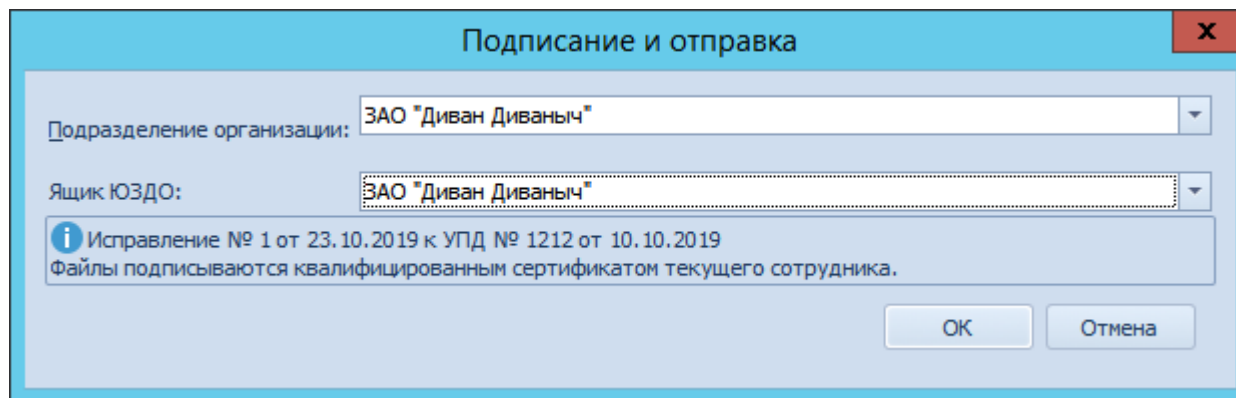


Рисунок 56. Подписание и отправка



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

7. При необходимости выберите другое подразделение/организацию и ящик ЮЗДО, через который будет отправлен электронный документ. Нажмите кнопку **ОК**. Будет открыто окно выбора сертификата для подписания документа.
8. Выберите сертификат квалифицированной подписи и нажмите кнопку **ОК**. Будет показано сообщение **Отправка успешно выполнена**. Состояние документа изменится на **Подписан и отправлен**.

После получения документа оператор обмена нужно будет подтвердить получение извещения о приёме документа.

После подписания документа контрагентом (формируется *титул покупателя*) документ перейдет в статус **УПД подписан с обеих сторон**. На этом документооборот

считается завершенным.

Для счёт-фактуры подписание контрагентом не требуется. Документооборот будет завершен после [подтверждения получения извещения](#). Статус обмена: [Отправлен контрагенту](#).

Обмен информацией по данному документу можно отслеживать с помощью [Журнала обмена ЮЗДО](#).

Работа с входящими документами

Модуль интеграции с операторами ЭДО

Работа с входящими документами через интерфейс Windows-клиента

Дополнительная информация для администратора Docsvision.

- [Подписание полученного формализованного документа](#)
- [Подтверждение получения извещения](#)
- [Аннулирование формализованного документа](#)
- [Больше подробностей в навигационном меню слева...](#)

Подписание полученного формализованного документа

Пользователь модуля должен принять решение по поступившему на подпись документу: подписать или отказать.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

1. Откройте документ со статусом обмена в [Журнале обмена ЮЗДО](#): [Получен на подпись от контрагента](#).

2. Заполните информацию о покупателе на странице *Подписание*. Не требуется для отказа в подписании.
3. Нажмите кнопку на ленте карточки для принятия решения по документу:
 - *Подписать документ* — для подписания документа. При этом нужно будет выбрать сертификат квалифицированной подписи, которым будут подписаны файлы документа.

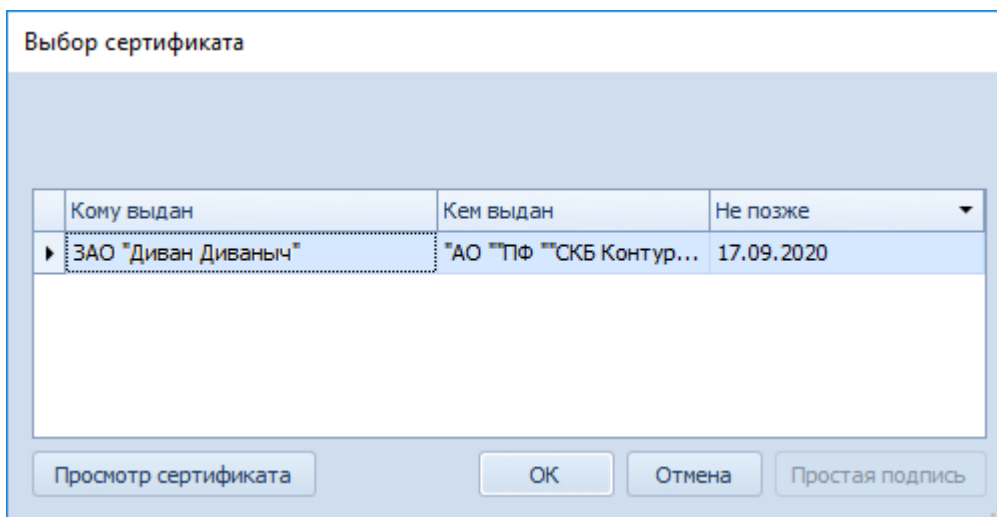


Рисунок 57. Выбор сертификата электронной подписи

- *Отказать в подписании* — для отказа от подписания полученного документа. При этом нужно будет указать причину отказа.

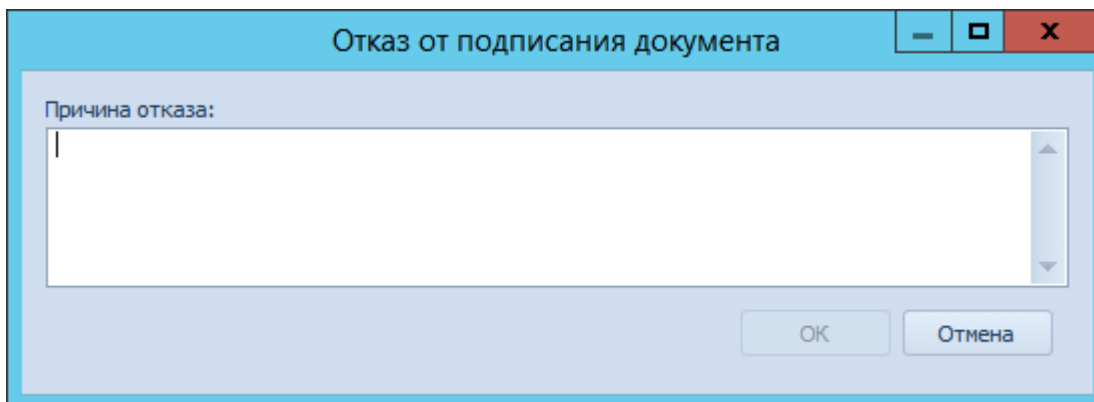


Рисунок 58. Указание причины отказа от подписания документа

Ответная подпись (или отказ) будет отправлена контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). После этого статус документа сменится на *Отправлена ответная подпись контрагенту* или *Отправлен отказ от подписи контрагенту* соответственно.

Запрос на уточнение

Покупатель может запросить у продавца уточнение полученного документа, если в документе были обнаружены ошибки. Запрос может быть выполнен после [подтверждения получения извещения](#).

После отправки запроса на уточнение возможность подписания и аннулирования документа будет заблокирована — дальнейший документооборот с использованием данного документа будет приостановлен. Если документ подписан или подписание не требуется, возможность аннулирования после отправки запроса не блокируется.

Чтобы отправить запрос на уточнение:

1. Откройте полученный документ.
2. Нажмите кнопку **Запросить уточнение** на ленте карточки. Будет предложено указать комментарий к уточнению.

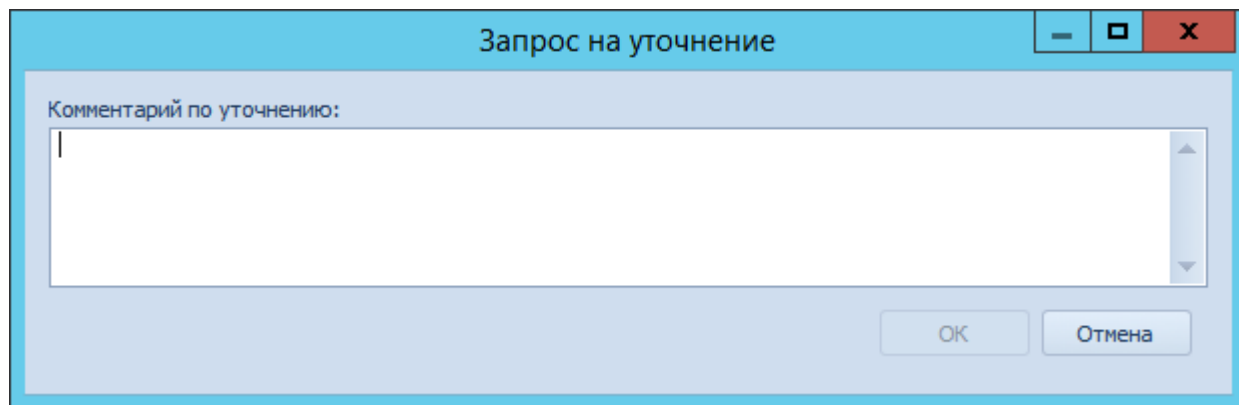


Рисунок 59. Запрос уточнения

3. Укажите комментарий и нажмите кнопку **ОК**. Будет предложено выбрать сертификат для подписания запроса на уточнение.
4. Выберите квалифицированный сертификат и нажмите **ОК**.

Запрос будет отправлен контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). После этого статус документа сменится на [Отправлен запрос на уточнение](#).

Контрагент может создать исправление или корректировку документа. Исправление или корректировка поступит в виде отдельного входящего документа, ссылка на который будет добавлена в документ, по которому было запрошено уточнение.

УПД № 12 от 23.09.2019 - Документ

Документ

Сохранить Печать Журнал подписей

Общие Подписи и шифр...

Номер документа: 12 Дата документа: 24.10.2019 Тип документа: Счет-фактура

Статус УПД: УПД подписан отправителем. Функция УПД: Счет-фактура

Основная информация Подписание Журнал обмена ЮЗДО Печатная форма

Контрагент: "Солнечный берег" ООО Ответственное подразделение: "Диван Диваныч" ЗАО

Сумма НДС: 20 000, Сумма документа с НДС: 1 000 000,

Описание:

Ссылки:

Тип ссылки	Автор	Дата создания	Ссылка
Ссылается на (ЮЗДО) (Обратная: Св...	WorkflowService	24.10.2019 15:32:43	Карточка 'УКД № 1 от 24.10.2019'

Рисунок 60. Ссылка на корректирующий документ в УПД

Подтверждение получения извещения

Оператор ЭДО формирует извещение, когда передаёт организации электронный документ или получает электронный документ от организации. При загрузке данного извещения в систему Docsvision статус обмена (на вкладке *Журнал обмена ЮЗДО*) изменится на **Требуется подтверждение получения**. Пользователь должен отправить подтверждение получения данного извещения.

Основная информация Печатная форма Журнал обмена ЮЗДО Ссылки

Отправить извещение Экспортировать файлы

Контрагент: ООО "Солнечный берег"

Статус: Требуется подтверждение получения

Ящик ЮЗДО: ЗАО "Диван Диваныч" Ящик контрагента: ООО "Солнечный берег"

Рисунок 61. Статус обмена электронными документами

В зависимости от этапа жизненного цикла документа определенные операции на

ленте карточки станут доступны только после отправки подтверждения получения извещения.

Чтобы подтвердить получение извещения:

1. Откройте карточку формализованного документа, по которому нужно отправить подтверждение.
2. Перейдите на вкладку *Журнал обмена ЮЗДО*.
3. Нажмите кнопку **Отправить извещение**.

Состояние документа: Подписан и отправлен Функция УПД: СЧФДОП

Статус УПД: УПД подписан отправителем.

Основная информация Печатная форма Журнал обмена ЮЗДО Ссылки

Отправить извещение Экспортировать файлы ▾

Контрагент: ООО "Солнечный берег"

Статус: Требуется подтверждение получения

Ящик ЮЗДО: ЗАО "Диван Диваныч" Ящик контрагента: ООО "Солнечный берег"

Тип собы...	Дата	Файлы	Подписант	Пр...	Описание	Комментарий
► Из Docsvi...	05.11.2...	УПД № 1222 от 10.10.2019			Документ успешно отправлен контрагенту.	

Рисунок 62. Кнопка "Отправить извещение" на вкладке "Журнал обмена ЮЗДО"

Будет открыто окно выбора сертификата для подписания извещения.

4. Выберите сертификат квалифицированной подписи и нажмите кнопку **ОК**.

Подтверждение будет отправлено контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). В *Журнале обмена ЮЗДО* появится запись о создании и (позже) отправке подтверждения получения извещения.

Аннулирование формализованного документа

Отправленный/полученный формализованный документ может быть аннулирован, если было **подтверждено получение извещения** от оператора ЭДО.

Отправленный документ может быть аннулирован в одностороннем порядке до получения извещения от оператора ЭДО. В данном случае подтверждение получения извещения не требуется.

Полученный документ может быть аннулирован продавцом, до подтверждения получения извещения или до подписания Акта/Накладной.

Чтобы отправить запрос на аннулирование:

1. Откройте аннулируемый документ.
2. Нажмите кнопку **Запросить аннулирование** на ленте карточки. Будет предложено указать причину аннулирования документа.

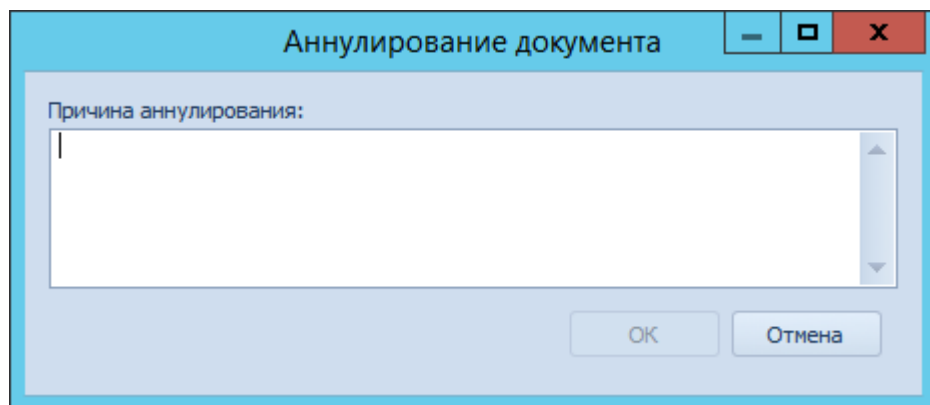


Рисунок 63. Указание причины аннулирования документа

3. Укажите причину аннулирования и нажмите кнопку **ОК**. Будет предложено выбрать сертификат для подписания сообщения об аннулировании.
4. Выберите квалифицированный сертификат и нажмите **ОК**.

Запрос на аннулирование будет отправлен после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*).

После подтверждения аннулирования или при одностороннем аннулировании статус документа сменится на **Аннулирован**. В случае отказа в аннулировании со стороны контрагента статус вернётся в предыдущее значение, за исключением случая, когда осуществляется аннулирование документа, по которому был отправлен запрос на уточнение — в данном случае статус сменится на **Получен от контрагента** или **Отправлена ответная подпись контрагенту**. Причина отказа будет указана в *Журнале обмена ЮЗДО*.

Подтверждение аннулирования формализованного документа

Если аннулирование инициировано контрагентом, организация может подтвердить аннулирование или отклонить.

1. Откройте документ со статусом обмена в **Журнале обмена ЮЗДО**: **Получен запрос на аннулирование**.
2. Нажмите кнопку на ленте карточки для принятия решения по документу:
 - **Подтвердить аннулирование** — для подтверждения аннулирования. При этом нужно будет выбрать сертификат квалифицированной подписи для

формирования ответной подписи.

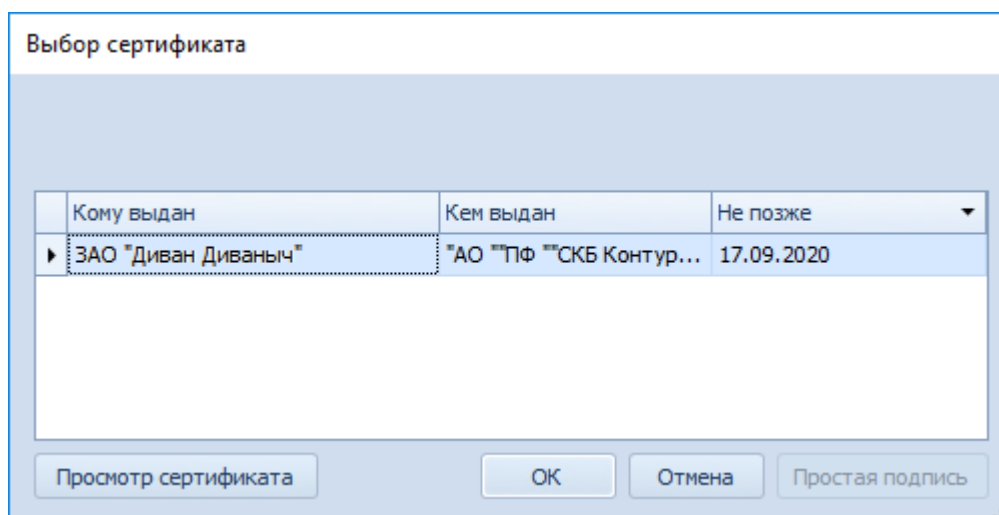


Рисунок 64. Выбор сертификата электронной подписи

- **Отказать в аннулировании** — для отказа от аннулирования. При этом нужно будет указать причину отказа.

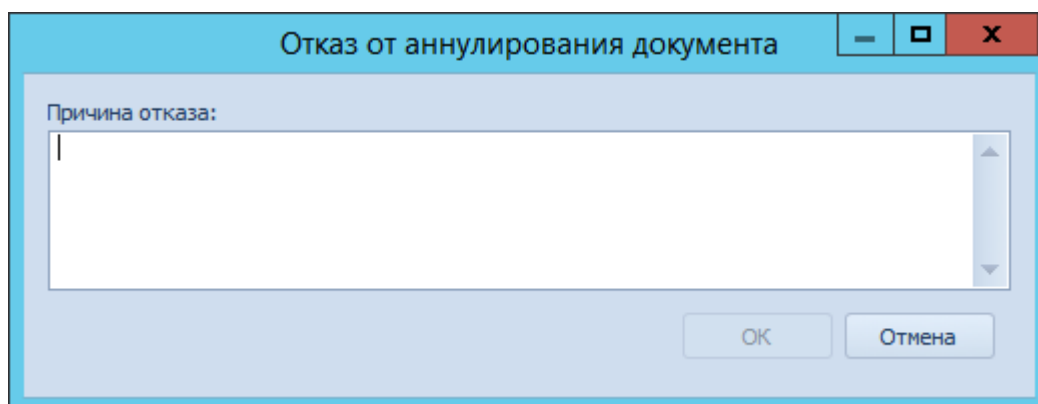


Рисунок 65. Указание причины отказа от аннулирования документа

Подтверждение (или отказ) аннулирования будет отправлено контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). При подтверждении аннулирования статус документа сменится на **Аннулирован**. При отказе в аннулировании статус документа сменится на **Получен от контрагента, Отправлена ответная подпись контрагенту, Отправлен контрагенту, Получена подпись от контрагента**. В случае отказа в аннулировании со стороны контрагента статус вернётся в предыдущее значение.

Печатная форма формализованного документа

Элемент разметки "Печатная форма" предназначен для просмотра электронного документа в печатном виде.

Корректировочный УПД - Документ

Документ

Сохранить Печать Журнал подписей

Общие Подписи и шифр...

ON_KORSCHFDOPPR_2BM-9660026731-966001000-201909240135490337231_2BM-9619136959-961901000-201909240750372227921_20191022_d4028ac1-b0fd-4b1c-9fff-dbf1ea

Состояние документа: Подписан и отправлен Функция УПД: УПД

Статус УПД: УПД подписан отправителем.

Основная информация Печатная форма Журнал обмена ЮЗДО Ссылки

Приложение № 1 к письму ФНС России от _____ № _____

Приложение № 2 к постановлению Правительства Российской Федерации от 26 декабря 2011 г. № 1137

Корректировочный счет-фактура № 1213 от 22.10.2019 (1), исправление корректировочного счета-фактуры № _____ от _____ (1a)

Статус: ☐ 1 - счет-фактура и передаточный документ (акт) 2 - передаточный документ (акт)

Продавец Диван Диваныч ЗАО (2)

Адрес Санкт-Петербург, (2a)

ИНН/КПП продавца 9619136959 / 961901000 (26)

Покупатель Солнечный берег ООО (3)

Адрес Красноярский край, (3a)

ИНН/КПП покупателя 9660026731 / 966001000 (36)

Валюта: наименование, код Российский рубль (4)

№ п/п	Код товара/ работ, услуг	Наименование товара (описание выполненных работ, оказанных услуг), имущественного права	Показатели в связи с изменением стоимости отгруженных товаров (выполненных работ, оказанных услуг), переданных имущественных прав	Единица измерения		Количество (объем)	Цена (тариф) за единицу измерения	Стоимость товаров (работ, услуг), имущественных прав без налога - всего	В том числе сумма акциза	Налоговая ставка	Сумма налога	Стоимость товаров (работ, услуг), имущественных прав с налогом - всего
				код	условное обозначение (национальное)							
A	B	1	1a	2a	26	3	4	5	6	7	8	9
1		Круглый лес	A (до изменения)			100	5000	500000	без	20%	100000	600000

Папки\ЮЗДО

Рисунок 66. Пример печатной формы

Для печати формы нажмите кнопку **Печать** на ленте карточки или используйте одноименную команду из контекстного меню *Печатной формы*.

Групповые операции с неформализованными документами

Пользователь может отправить контрагенту сразу несколько подготовленных неформализованных документов (подписанных или не подписанных), подписать или отказать в подписании неформализованных документов, по которым контрагентами запрошена подпись.

Данные групповые операции выполняются с помощью команд, расположенных на ленте Windows-клиента.

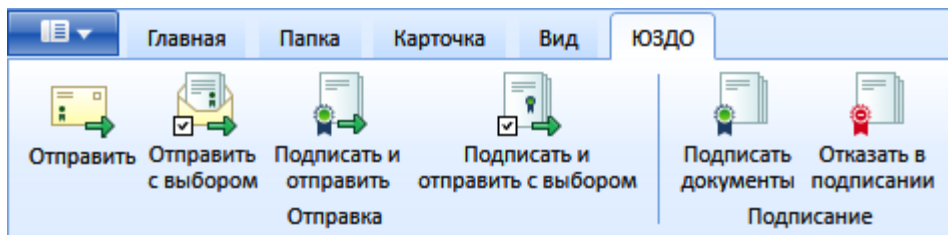


Рисунок 67. Команды выполнения групповых операций с неформализованными документами

- **Отправить** — отправляет все основные файлы выбранных карточек.
- **Отправка с выбором** — отправляет выбранные файлы выбранных карточек.
- **Подписание и отправка** — подписывает и отправляет все основные файлы выбранных карточек.
- **Подписание и отправка с выбором** — подписывает и отправляет файлы выбранных карточек.
- **Подписать документы** — подписывает выбранные документы, для которых требуется подписание.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- **Отказать в подписании** — отказывает в подписании выбранных документов, для которых требуется подписание.

Групповая отправка подписанных документов

Данная групповая операция предназначена для отправки нескольких неформализованных документов, подписанных на предыдущих этапах работы с документами (например, при согласовании). Если документ не подписан, он может быть подписан текущим сотрудником при отправке.

1. Выберите в папке карточки для отправки контрагентам.

В выбранных карточках должны быть указаны получатели.

2. Выберите способ отправки документов:

а. **С выбором отправляемых файлов.**

- i. Нажмите кнопку **Отправить с выбором** на ленте Windows-клиента. Будет открыто окно отправки документов с выбором файлов.

Отправка документов с выбором файлов

Подразделение организации: ЗАО "Диван Диваныч"

Ящик ЮЗДО: ЗАО "Диван Диваныч"

Карточек: 2 / Файлов: 2

Выберите файлы для контрагентов:

Документ	Под...
Исходящий Договор на поставк...	1
Исходящий Дополнительное сог...	1

☒ Выбрать все ☐ Очистить все

	"Солнечный берег" ООО
Договор на поставку оборудования....	☒

Выберите подпись:

	Дата	Подписант	Метка
☒	18.10.2019	Быков М. А.	

ОК Отмена

Рисунок 68. Окно отправки документов с выбором файлов

- ii. В поле *Подразделение организации* выберите организацию (или подразделение), от имени которой отправляются документы.

В списке приведены организации, загруженные в *Справочник настроек операторов ЮЗДО* из системы оператора ЭДО.

- iii. В поле *Ящик ЮЗДО* выберите ящик электронного обмена организации, через который будут отправлены документы.

В списке приведены ящики электронного обмена, зарегистрированные для выбранного подразделения организации.

- iv. В левой таблице выберите документ, для которого нужно изменить список отправляемых файлов.

- v. В правой таблице выберите отправляемые контрагентам файлы, установив или сняв флажки в списке файлов для каждого получателя (контрагента).

Если отправка документов контрагенту от выбранного подразделения организации невозможна, колонка выбора отправляемых файлов для данного контрагента будет заблокирована (флажки сняты со всех файлов).

Если в списке есть файлы, неподписанные квалифицированной ЭП, они будут выделены красным цветом.

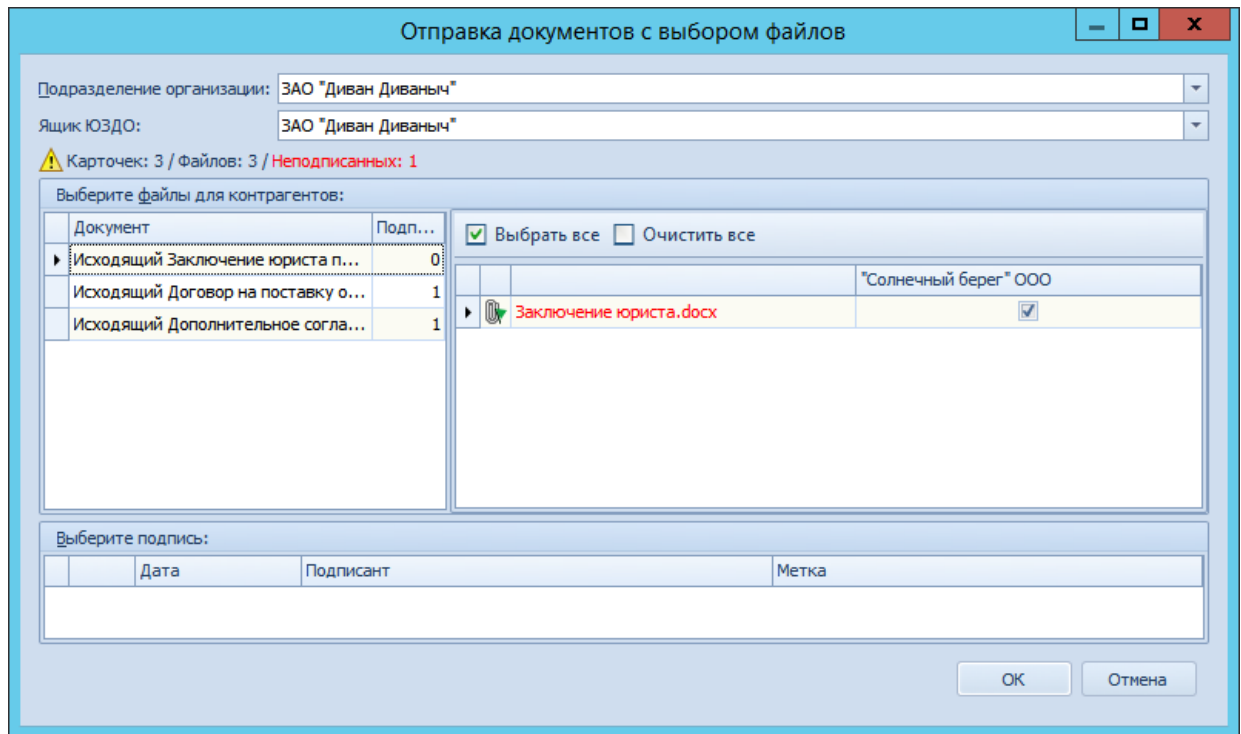


Рисунок 69. Окно отправки документов с выбором файлов с неподписанными файлами

- vi. В нижней таблице выберите подпись, с которой документ будет отправлен контрагенту.
- b. **Без выбора отправляемых файлов.**
 - i. Нажмите кнопку **Отправить** на ленте Windows-клиента. Будет открыто окно отправки документов.

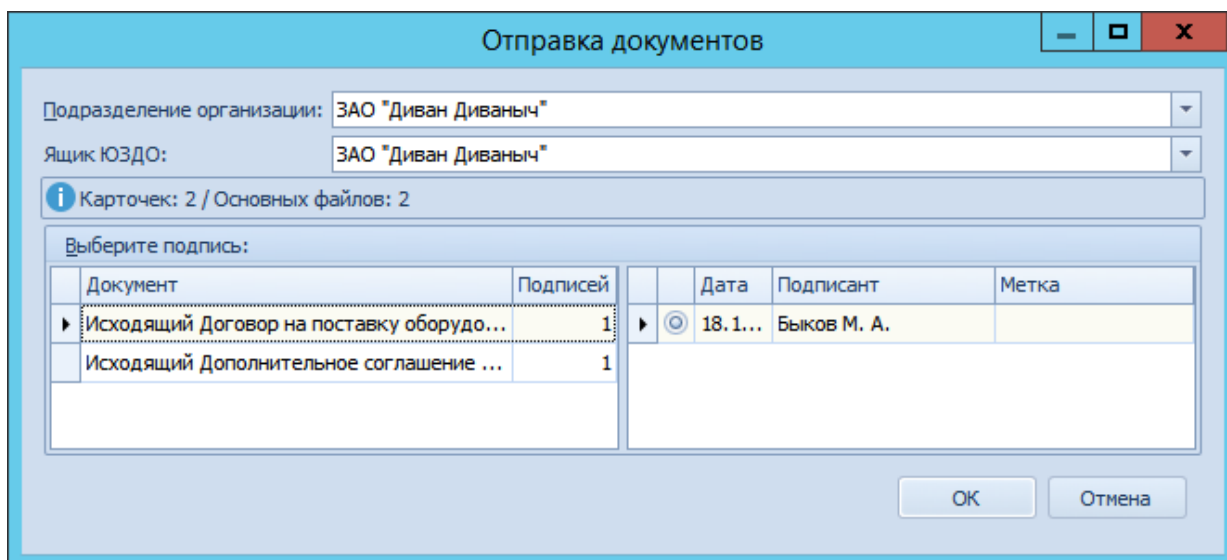


Рисунок 70. Окно отправки документа

- с. В поле *Подразделение организации* выберите организацию (или подразделение), от имени которой отправляется документ.
- і. В поле *Ящик ЮЗДО* выберите ящик электронного обмена организации, через который будет отправлен электронный документ.

Если в списке есть файлы, неподписанные квалифицированной ЭП, данная информация будет указана в окне отправки.

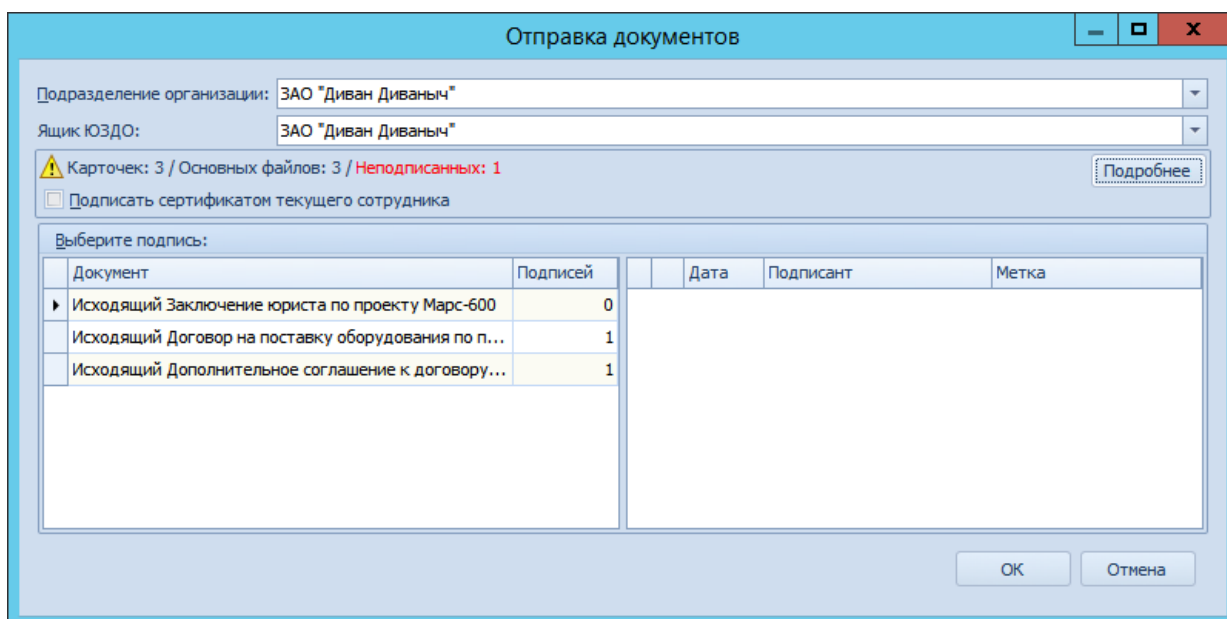


Рисунок 71. Окно отправки документа с неподписанными файлами

Чтобы посмотреть список неподписанных файлов, нажмите кнопку **Подробнее**. Чтобы подписать файлы сертификатом текущего сотрудника, установите флаг **Подписать сертификатом текущего сотрудника**.

3. Нажмите кнопку **ОК** в окне отправки. После формирования документов для отправки оператору ЭДО будет показано сообщение **Сообщения отправки документов успешно созданы для всех получателей**.

Если в списке отправляемых файлов есть неподписанные файлы будет предложено выбрать сертификат для их подписания. Выберите сертификат квалифицированной подписи и нажмите кнопку **ОК**. Подписи **всех** отправляемых файлов будут переформированы.

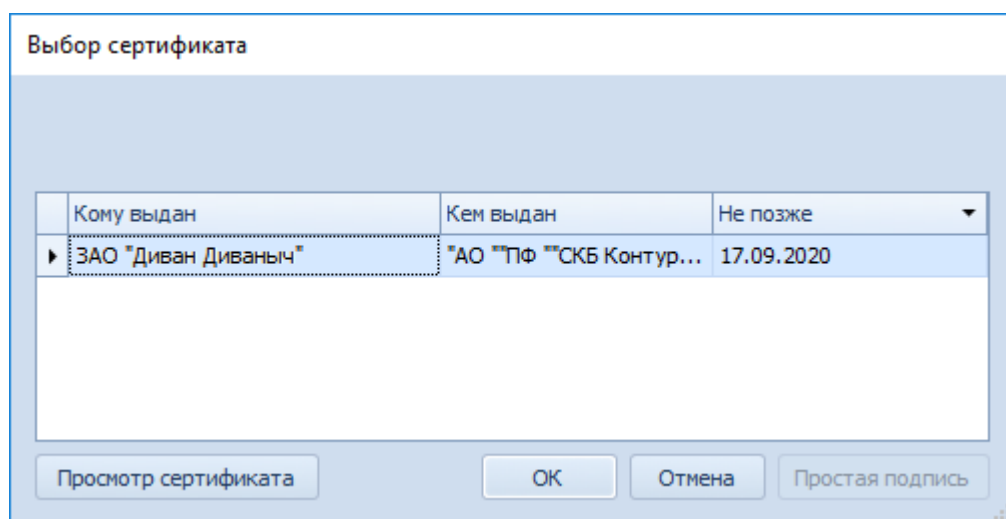


Рисунок 72. Выбор сертификата электронной подписи

Документы будут отправлены контрагентам на подпись в течении некоторого времени. Состояние отправки документов можно отслеживать с помощью [Журнала обмена ЮЗДО](#).

Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

Групповая отправка документов с подписанием

Данная групповая операция предназначена для отправки нескольких

неформализованных документов с предварительным подписанием квалифицированной подписью.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

1. Выберите в папке карточки для отправки контрагентам.

В выбранных карточках должны быть указаны получатели.

2. Выберите способ отправки документов:

а. **С выбором отправляемых файлов.**

i. Нажмите кнопку **Подписать и отправить с выбором** на ленте Windows-клиента. Будет открыто окно отправки документов с выбором файлов.

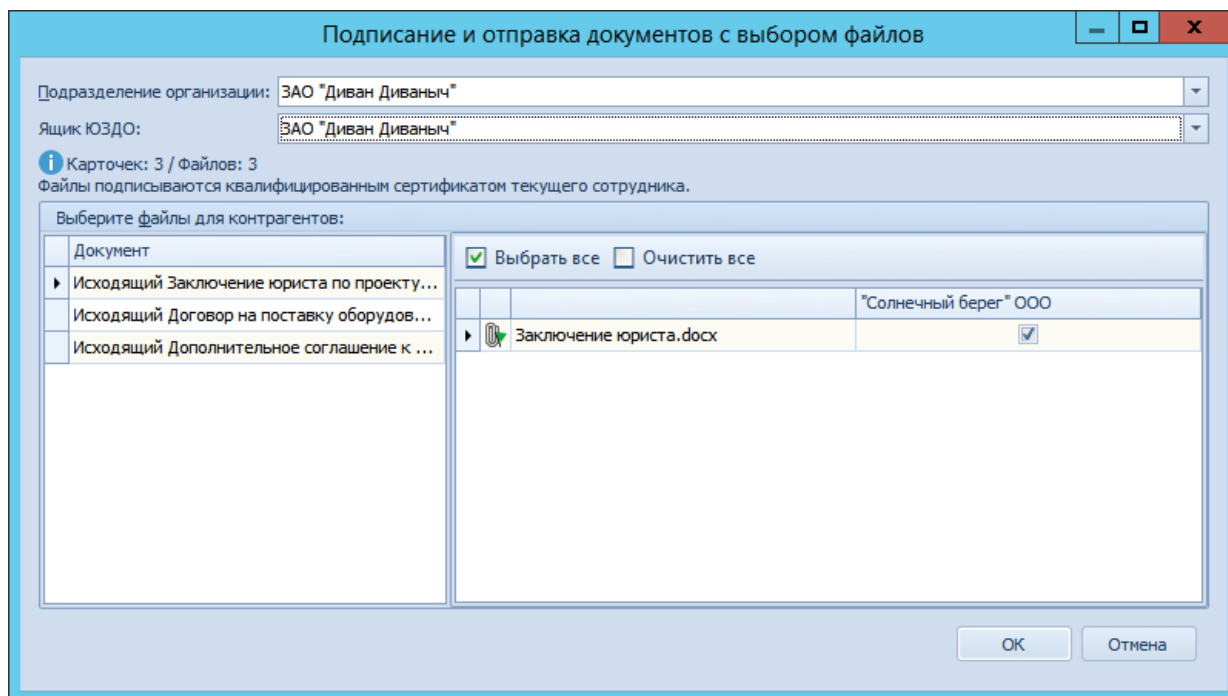


Рисунок 73. Окно отправки документов с выбором файлов

ii. В поле *Подразделение организации* выберите организацию или подразделение, от имени которых отправляются документы.

- iii. В поле *Ящик ЮЗДО* выберите ящик электронного обмена организации, через который будут отправлены документы.
- iv. В левой таблице выберите документ, для которого нужно изменить список отправляемых файлов.
- v. В правой таблице выберите отправляемые контрагентам файлы, установив или сняв флажки в списке файлов для каждого получателя (контрагента).

б. Без выбора отправляемых файлов.

- i. Нажмите кнопку **Подписать и отправить** на ленте Windows-клиента. Будет открыто окно отправки документов.

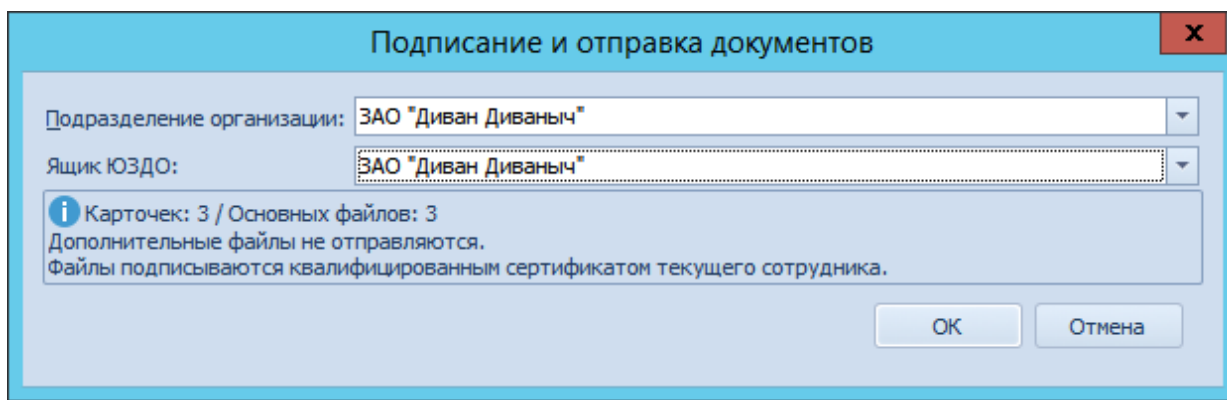


Рисунок 74. Окно отправки документа

- ii. В поле *Подразделение организации* выберите организацию (или подразделение), от имени которой отправляются документы.
 - iii. В поле *Ящик ЮЗДО* выберите ящик электронного обмена организации, через который будут отправлены документы.
3. Нажмите кнопку **ОК** в окне отправки. Будет открыто окно выбора сертификата для подписания отправляемых файлов.

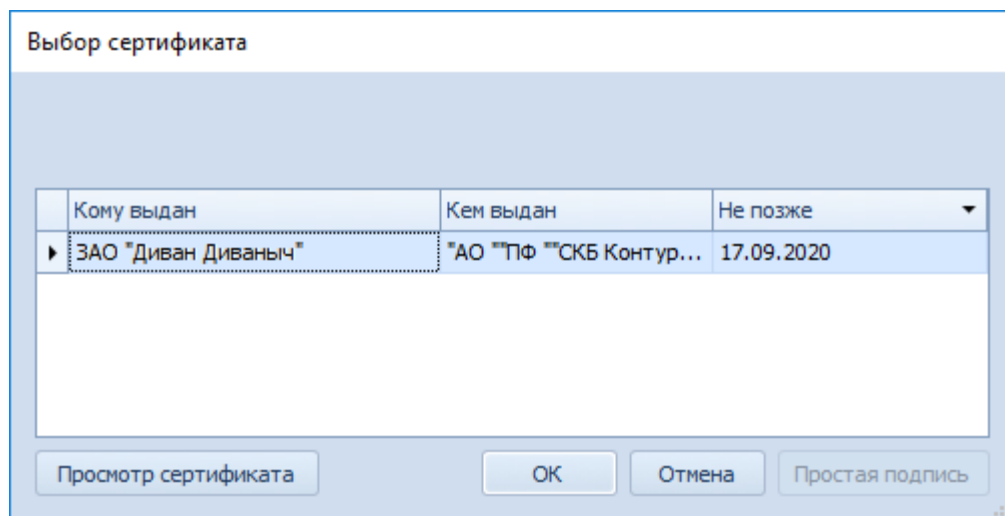


Рисунок 75. Выбор сертификата электронной подписи

4. Выберите сертификат квалифицированной подписи и нажмите кнопку **ОК**. После подписания и формирования документов для отправки оператору ЭДО будет показано сообщение **Сообщения отправки документов успешно созданы для всех получателей**.

Документы будут отправлены контрагентам на подпись в течении некоторого времени. Состояние отправки документов можно отслеживать с помощью [Журнала обмена ЮЗДО](#).

Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

Групповое подписание полученного неформализованного документа

Данная групповая операция предназначена для подписания или отказа в подписании нескольких неформализованных документов, если необходимость подписания определена отправившими документы контрагентами.

1. Выберите в папке карточки входящих неформализованных документов, для которых нужно принять решение по подписанию.
2. Нажмите кнопку на ленте Windows-клиента для принятия решения по документу:
 - **Подписать документы** — для подписания выбранных документов. При этом нужно будет выбрать сертификат квалифицированной подписи, которым будут подписаны файлы документов.

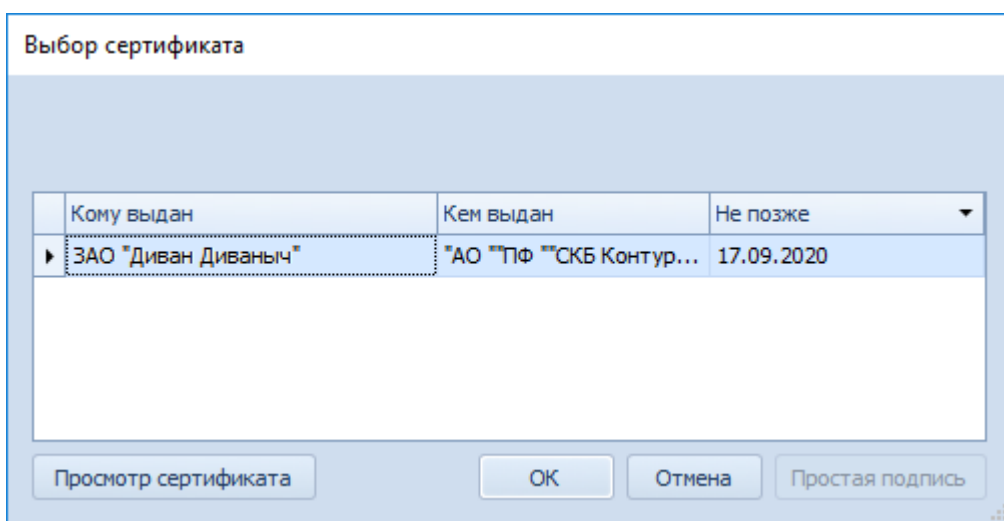


Рисунок 76. Выбор сертификата электронной подписи



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- **Отказать в подписании** — для отказа от подписания полученных документов. При этом нужно будет указать причину отказа.

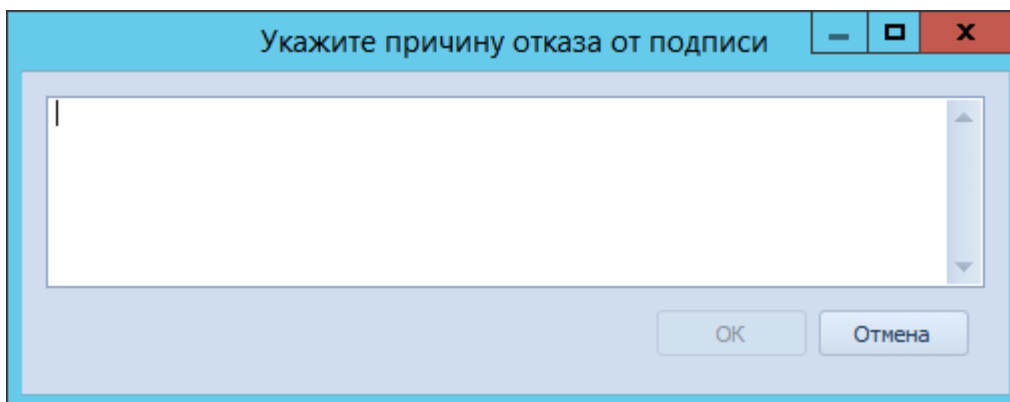


Рисунок 77. Указание причины отказа в подписании документов

В течении некоторого времени контрагентам будут отправлены ответные подписи или отказы в подписании. Статусы документов сменятся на **Отправлена ответная подпись контрагенту** или **Отправлен отказ от подписи контрагенту** соответственно.

Если в списке выбранных документов, есть документы не требующие ответную подпись, будет показано сообщение "Часть отказов от подписи по выбранным документам отправить не удалось". Список документов, по которым не были отправлены ответные подписи, приведён в разделе *Дополнительная информация* информационного сообщения.

Журнал обмена ЮЗДО

Журнал обмена ЮЗДО содержит историю электронного обмена по данному документу с оператором ЭДО.

Для просмотра журнала откройте страницу *Журнал обмена ЮЗДО* в карточке электронного документа. Пример журнала приведен на рисунке.

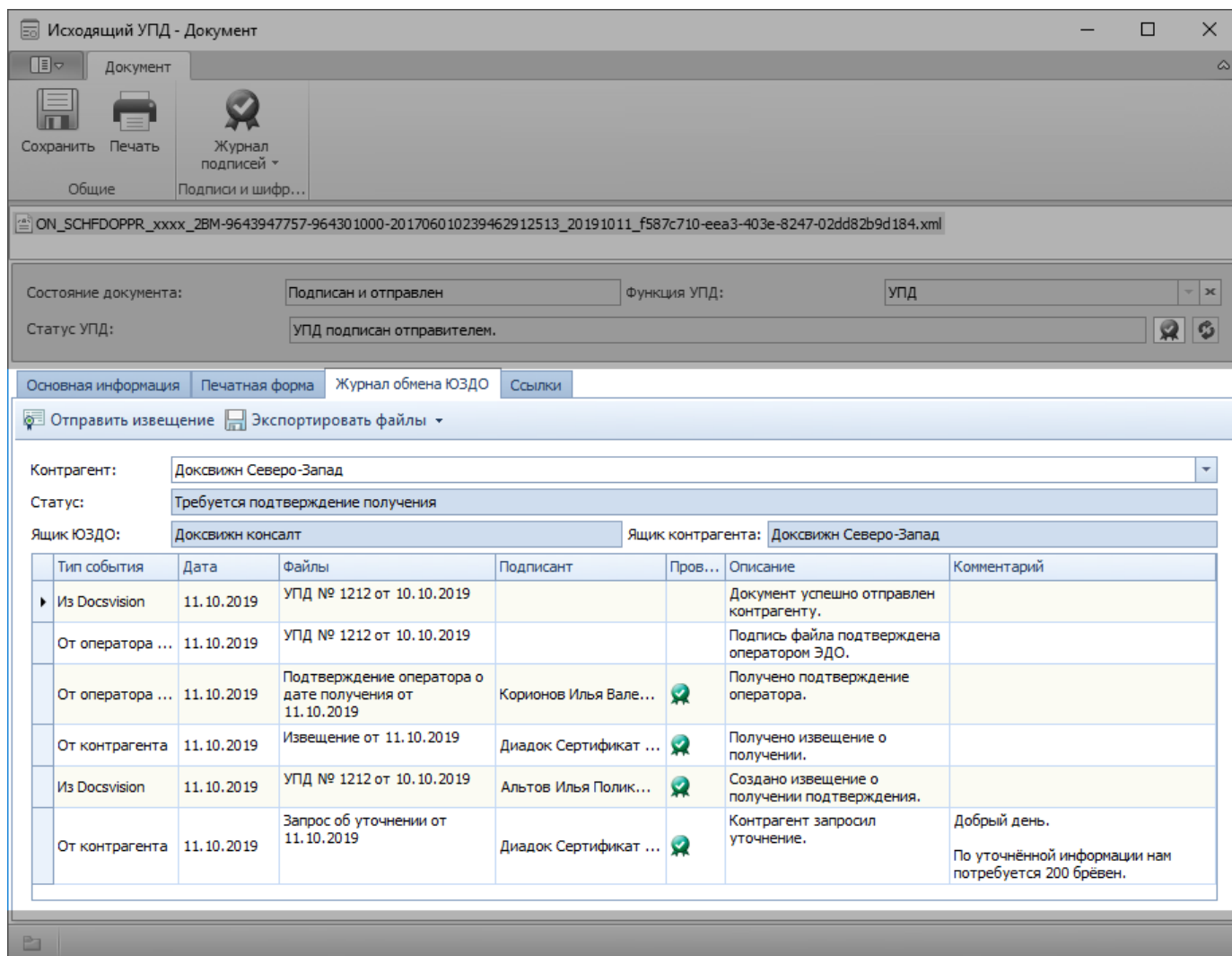


Рисунок 78. Журнал обмена ЮЗДО в карточке исходящего УПД

Журнал содержит следующие элементы:

- Кнопки:
 - **Отправить извещение** — отправляет извещение-подтверждение о получении оператора ЭДО. Кнопка доступна только при статусе обмена: **Требуется подтверждение получения**.
 - **Экспортировать файлы** — сохраняет выбранный или все файлы и подписи на диск.
- **Контрагент** — название контрагента, которому отправлен или от которого получен документ. В исходящих документах в данном поле можно выбрать контрагента, историю обмена с которым нужно открыть.
- **Статус** — текущий статус обмена документом с контрагентом. Может принимать следующие значения:
 - **Получен на подпись от контрагента** — электронный документ получен от

контрагента и нуждается в подписании.

- **Получен от контрагента** — получен электронный документ, не требующий подписания (например, счет-фактура или неформализованный документ без необходимости подписания).
- **Отправлен контрагенту** — электронный документ отправлен контрагенту.
- **Получен отказ от контрагента** — контрагент отказал в подписании отправленного электронного документа.
- **Получена подпись от контрагента** — контрагент подписал отправленный электронный документ.
- **Ошибка** — возникала ошибка при обмене данным электронным документом на стороне Docsvision, оператора ЭДО или контрагента.
- **Готово к отправке** — электронный документ готов, но не отправлен контрагенту — не был обработан процессами модуля (БП *Отправка сообщений ЮЗДО*).
- **Отправлена ответная подпись контрагенту** — организация подписала полученный от контрагента электронный документ.
- **Требуется подтверждение получения** — организация получила документ, для которого нужно отправить подтверждение о получении.
- **Требуется уточнение** — от контрагента получен запрос на уточнение.
- **Получен запрос на аннулирование** — контрагент отправил запрос на аннулирование документа.
- **Отправлен запрос на аннулирование** — отправлен запрос на аннулирование документа.
- **Аннулирован** — контрагент или организация подтвердили аннулирование документа.
- **Отправлен отказ от подписи контрагенту** — организация отказала в подписании полученного от контрагента электронного документа.
- **Отправлен запрос на уточнение** — контрагенту отправлен запрос на уточнение реквизитов документа.
- **Ящик ЮЗДО** — ящик организации, через который отправлен/получен документ.
- **Ящик контрагента** — ящик контрагента, на который или с которого отправлен документ.

- *Журнал событий* с полями:

- *Тип события* — указывает на источник/тип события. Возможные типы событий: Из Docsvision, От оператора обмена, От контрагента, Ответ, Ошибка, Информация.
- *Дата* — дата и время выполнения события.
- *Файлы* — названия отправленных/полученных файлов.
- *Подписант* — владелец квалифицированного сертификата, которым подписан отправленный файл.
- *Проверка в подписи* — результат проверки электронной подписи. При двойном щелчке по значку результата проверки будет открыто окно с информацией о сертификате.
- *Описание* — описание события.
- *Комментарий* — комментарий, поступивший или сформированный для контрагента (например, при отказе в подписании).

Для обновления содержимого журнала вызовите команду **Обновить** из контекстного меню журнала событий.

Работа с приглашениями к обмену



Работа с приглашениями по умолчанию производится только из Web-клиента. Для работы с приглашениями через Windows-клиент администратор должен самостоятельно настроить разметки.

Для Windows-клиента был реализован *Журналом работы с приглашениями ЭДО*. Администратор самостоятельно должен настроить разметки.

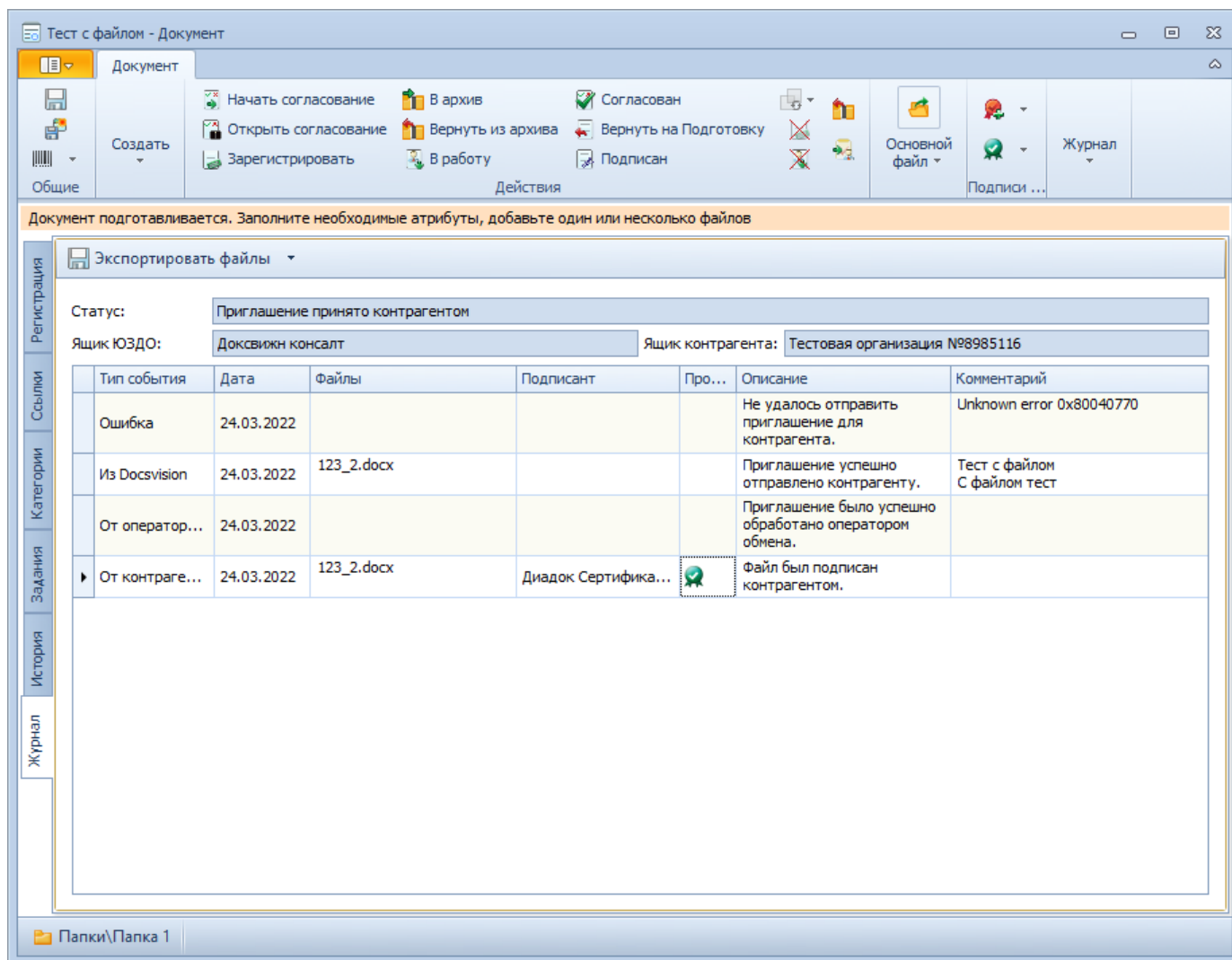


Рисунок 79. Журнал работы с приглашениями ЭДО

[1] Указываются рекомендуемые названия элементов управления, которые могут отличаться от фактических.

[2] Указываются рекомендуемые названия элементов управления, которые могут отличаться от фактических.

Руководство пользователя при работе в Web-клиенте

Аннотация

Данный раздел описывает работу с модулем через интерфейс Web-клиента. В данной версии функциональность модуля при работе из Web-клиента поддерживается не полностью.

Раздел содержит следующие подразделы:

- **Работа с неформализованными документами** — операции с неформализованными документами:
 - **"Отправка с подписанием"** — создание нового неформализованного документа с отправкой и подписанием,
 - **"Подписание полученных"** — подписание ЭП документа, полученного от оператора ЭДО,
 - **"Аннулирование"** — признание документа недействительным,
 - **"Подтверждение аннулирования"** — подтверждение запроса на аннулирование, полученного от документа ЭДО.
- **Работа с формализованными документами:**
 - **Карточки формализованных документов** — описание системных карточек формализованных документов,
 - **Исходящие** — работа с исходящими формализованными документами,
 - **Входящие** — работа с входящими формализованными документами,
 - **Подтверждение извещения** — работа с входящими и исходящими извещениями,
 - **Аннулирование** — процесс аннулирования документов,
 - **Подтверждение аннулирования** — подтверждение или отказ от аннулирования документа,
 - **Печатная форма** — использование печатной формы формализованного документа.
- **Групповые операции** с неформализованными документами.
- **Журнал обмена ЮЗДО** — системный журнал, содержащий историю электронного обмена по данному документу с оператором ЭДО.

- **Работа с приглашениями к обмену** — приглашения контрагента к началу обмена электронными документами.

Работа с неформализованными документами

В данном разделе рассмотрены основные этапы работы пользователя с неформализованными входящими и исходящими документами в интерфейсе модуля *Web-клиент*.

Для всех документов поддерживаются подтверждения оператора и извещения о получении участников обмена формата 1.03, в соответствии с приказом ФНС России от 21.02.2023 № ЕД-7-26/133@.

Изменения затрагивают следующие технологические документы:

- Извещения о получении документа (**DP_IZVPOL**).
- Уведомления об уточнении (**DP_UVUTOCH**).
- Подтверждения даты отправки (**DP_PDOTPR**).
- Подтверждения даты получения (**DP_PDPOL**).

Отправка подписанного неформализованного документа

В данном разделе описана отправка неформализованного подписанного документа через Web-клиент. Документ может быть подписан на предыдущих этапах работы, например, при согласовании.

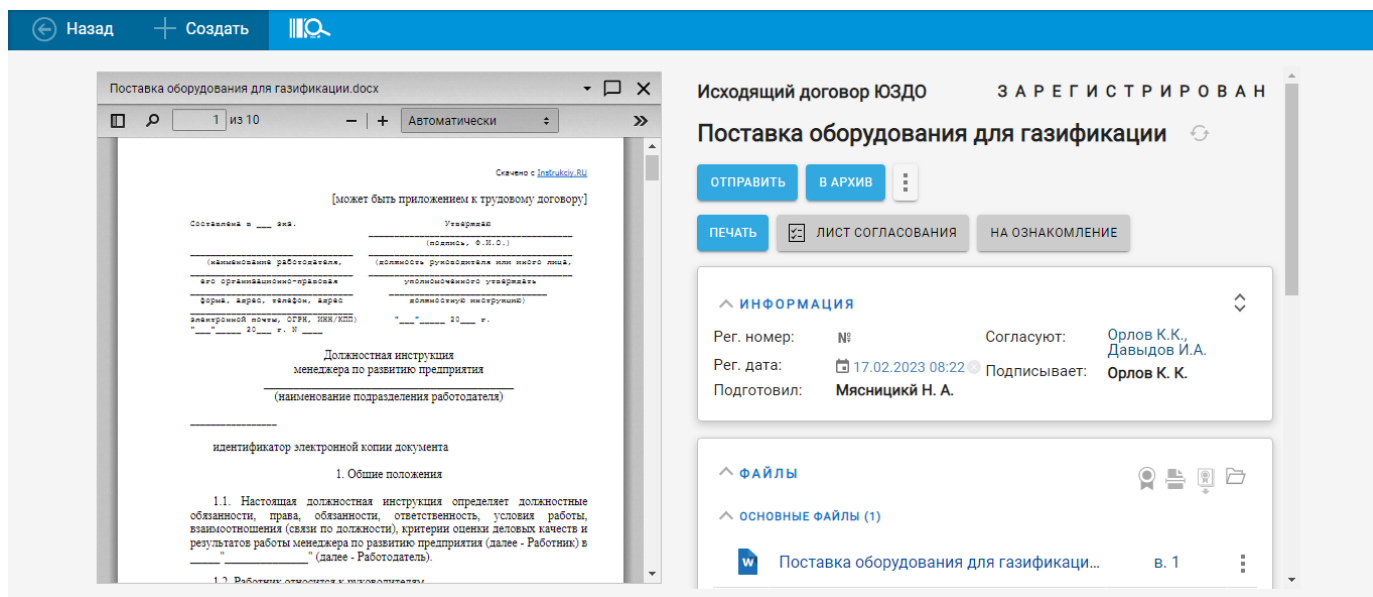


Рисунок 80. Отправка неформализованного документа

Чтобы отправить неформализованный документ с существующей ЭП через Web-клиент:

1. Откройте отправляемый документ.

В документе должны быть указаны *Получатели* — контрагенты, которым будет отправлен документ.

2. Нажмите кнопку **Отправить** на ленте карточки (кнопка добавляется администратором, название может не совпадать).

Если основные файлы не были подписаны или содержат недействительную подпись, данная информация будет выведена в предупреждении.

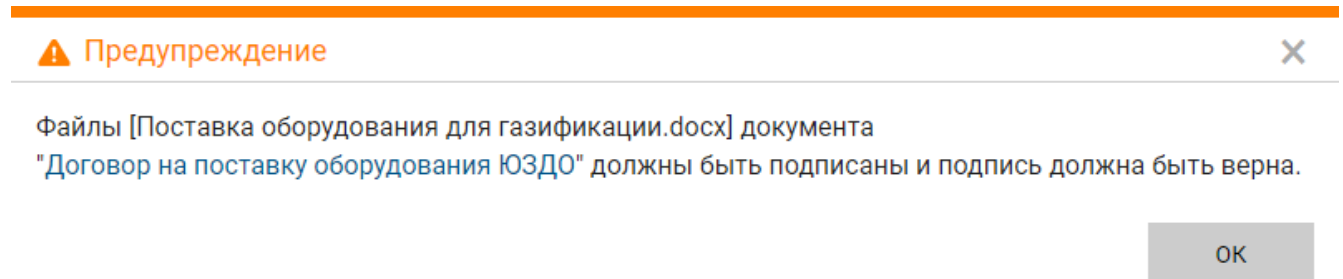


Рисунок 81. Предупреждение о неподписанных файлах



Отправка выполняется **Без выбора отправляемых файлов.**

В данном режиме отправляются только основные файлы документа. Дополнительные файлы не отправляются.

Будет открыто окно отправки документа с выбором ящика, с которого будет отправлен документ.

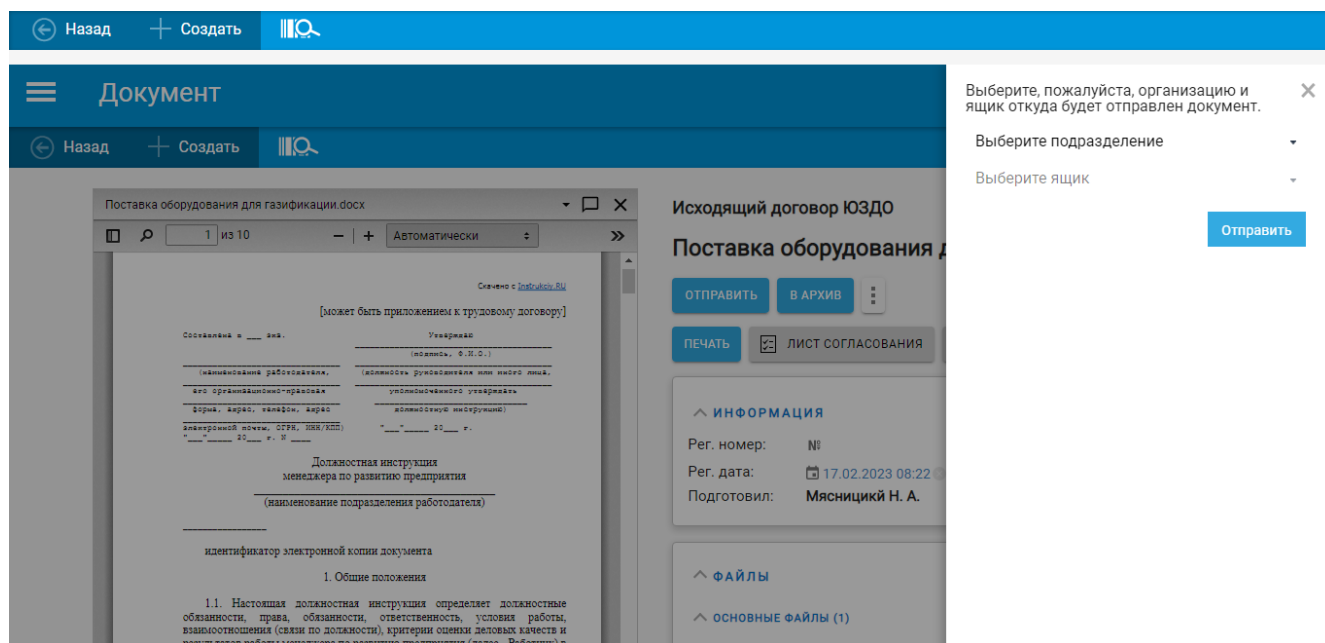


Рисунок 82. Окно отправки документа с выбором ящика

3. Выберите подразделение из списка или организацию, от имени которой отправляется документ.

В списке приведены организации, загруженные в *Справочник настроек операторов ЮЗДО* из системы оператора ЭДО.

4. Нажмите кнопку **Отправить**.

Документы будут отправлены контрагентам на подпись в течении некоторого времени. Состояние отправки документов можно отслеживать с помощью [Журнала обмена ЮЗДО](#).

Отправка неформализованного документа с подписанием

Для отправки и подписания в неформализованный документ при [настройке модуля](#) должна быть добавлена кнопка **Подписать и отправить** с обработчиком. После нажатия на нее будет запрос сертификата и выбор ящика

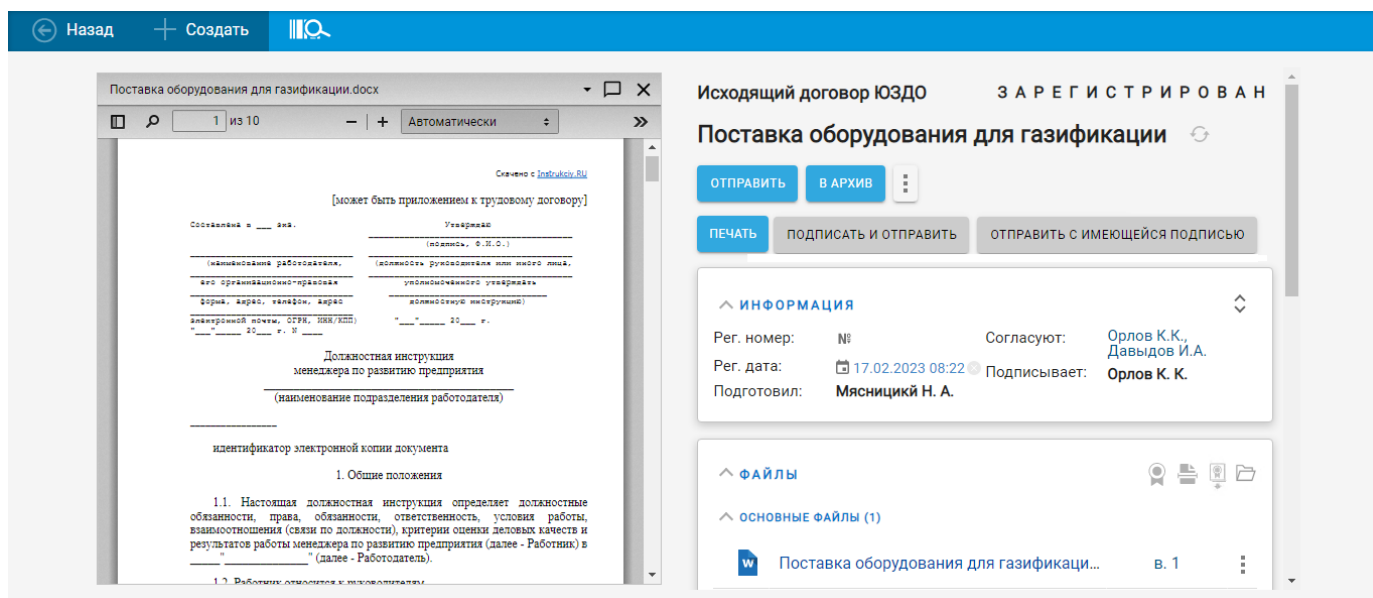


Рисунок 83. Отправка неформализованного документа

Чтобы отправить документ с подписанием через Web-клиент:

Если отправляемые документы были подписаны квалифицированной подписью на предыдущих этапах работы с документом (например, при согласовании), при отправке с подписанием существующая ЭП будет перезаписана. Для отправки документа с сохранением существующей ЭП следуйте инструкции [Отправка подписанного неформализованного документа](#).

Чтобы отправить неформализованный документ с подписанием:

1. Откройте отправляемый документ.

В документе должны быть указаны *Получатели — контрагенты*, которым будет отправлен документ.

При настройке разметок следует учитывать, что данные о контрагенте для отправки берутся из секции *Получатели — контрагенты*, *Получатель — организация контрагента* — поле в этой секции.

2. Нажмите на кнопку **Отправить с существующей подписью** и в раскрывающемся меню выберите *Подписать*.

Рисунок 84. Выберите сертификат электронной подписи

3. Выберите любой сертификат из списка, чтобы подписать документ усиленной локальной или облачной подписью.
4. Нажмите кнопку **Отправить** на ленте карточки (кнопка добавляется администратором, название может не совпадать).



Отправка выполняется **Без выбора отправляемых файлов**.

В данном режиме отправляются только основные файлы документа. Дополнительные файлы не отправляются.

Будет открыто окно отправки документа с выбором ящика, с которого будет отправлен документ.

Рисунок 85. Окно отправки документа с выбором ящика

5. Выберите подразделение из списка или организацию, от имени которой отправляется документ.

В списке приведены организации, загруженные в *Справочник настроек операторов ЮЗДО* из системы оператора ЭДО.

6. Нажмите кнопку **Отправить**.

Документы будут отправлены контрагентам на подпись в течении некоторого времени. Состояние отправки документов можно отслеживать с помощью [Журнала обмена ЮЗДО](#).

Подписание полученного неформализованного документа

Входящий неформализованный документ должен быть подписан, если данное требование установлено контрагентом. Организация может подписать документ или отказать в подписании.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

Чтобы подписать входящий неформализованный документ:

1. Откройте документ со статусом **Получен на подпись от контрагента**.
2. Нажмите кнопку на ленте карточки для принятия решения по документу:
 - **Подписать** — для подписания документа. При этом нужно будет выбрать сертификат квалифицированной подписи, которым будут подписаны файлы документа.

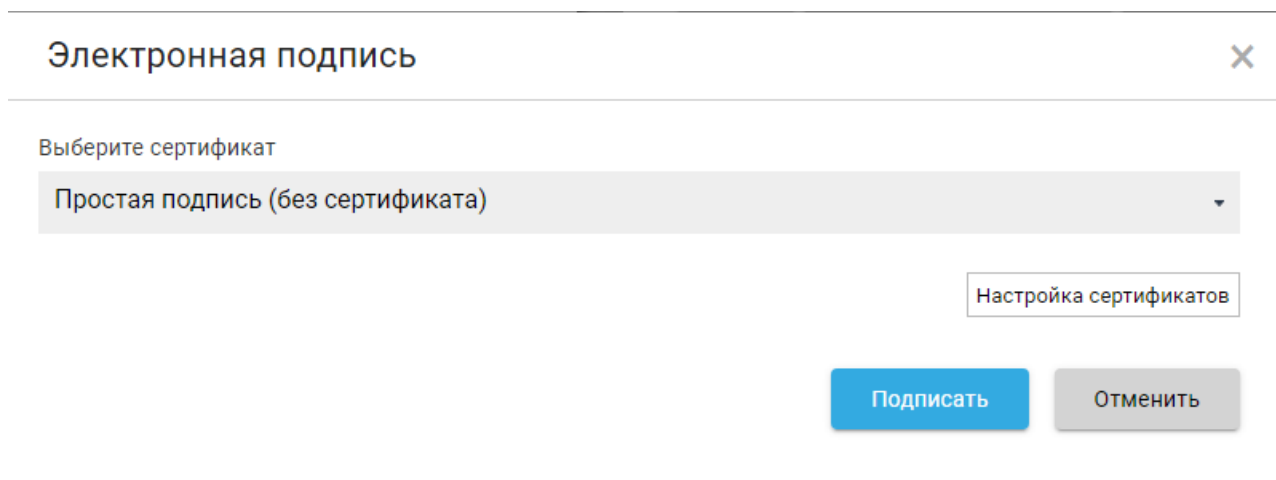


Рисунок 86. Выбор сертификата электронной подписи

- **Отказать** — для отказа от подписания полученного документа. При этом нужно будет указать причину отказа.



Рисунок 87. Указание причины отказа от подписания документа

Ответная подпись (или отказ) будет отправлена контрагенту после обработки процессами модуля (БП **Отправка сообщений ЮЗДО**). После этого статус документа сменится на **Отправлена ответная подпись контрагенту** или **Отправлен отказ от подписи контрагенту** соответственно.

Аннулирование неформализованного документа

Аннулирование документа может быть инициировано любой из сторон ЭДО.

Для аннулирования неформализованного документа по инициативе организации:

1. Откройте аннулируемый документ.
2. Нажмите кнопку **Запросить аннулирование** на ленте карточки. Будет предложено указать причину аннулирования документа.

Введите комментарий

ОК

Отмена

Рисунок 88. Указание причины аннулирования документа

3. Укажите причину аннулирования и нажмите кнопку **ОК**. Будет предложено выбрать сертификат для подписания сообщения об аннулировании.
4. Выберите квалифицированный сертификат и нажмите **ОК** для отправки запроса на аннулирование.

Запрос аннулирования будет отправлен контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). После этого статус документа сменится на **Отправлен запрос на аннулирование**.

Исходящий документ, который не был подписан контрагентом, будет аннулирован в одностороннем порядке. В других случаях контрагент должен подтвердить аннулирование.

Статус документа, аннулирование которого подтверждено, сменится на **Аннулирован**. Если в аннулировании было отказано (причина отказа будет указана в *Журнале обмена ЮЗДО*), статус обмена сменится на:

- а. **Получена подпись от контрагента** или **Получен от контрагента** (если для данного документа не требовалось подписание) — для исходящего документа.
- б. **Получен от контрагента** — для входящего документа.

Подтверждение аннулирования неформализованного документа

Если аннулирование инициировано контрагентом, организация может подтвердить аннулирование или отклонить.

1. Откройте документ со статусом обмена **Получен запрос на аннулирование** (в *Журнале обмена ЮЗДО*).
2. Нажмите кнопку на ленте карточки для принятия решения по документу:
 - **Подтвердить аннулирование** — для подтверждения аннулирования. При этом нужно будет выбрать сертификат квалифицированной подписи для

формирования ответной подписи.

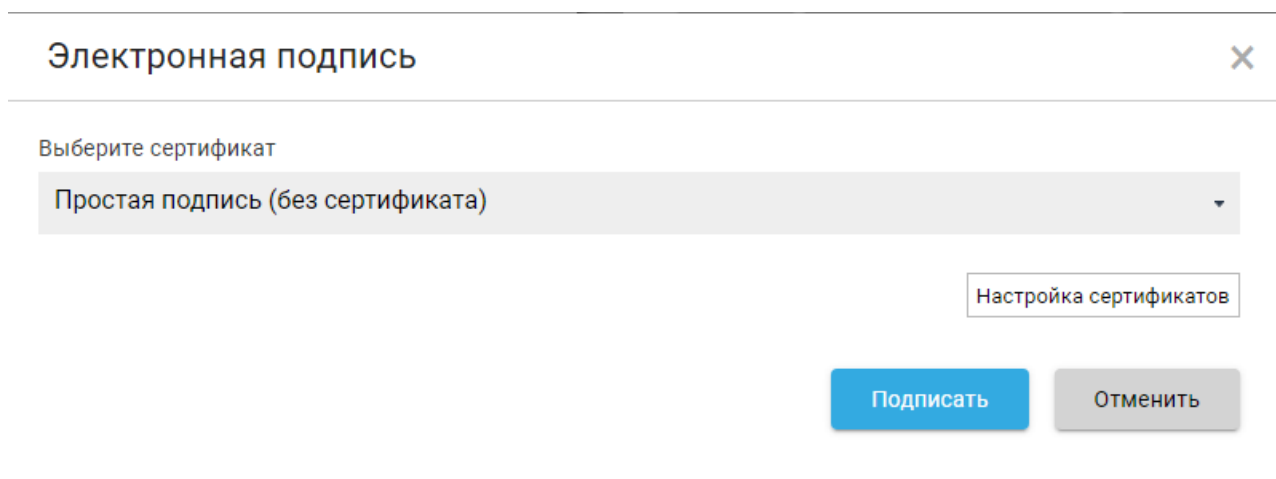


Рисунок 89. Выбор сертификата электронной подписи

- **Отказать в аннулировании** — для отказа в аннулировании. При этом нужно будет указать причину отказа.



Рисунок 90. Указание причины отказа в аннулировании документа

Подтверждение (или отказ) аннулирования будет отправлено контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). После этого статус документа сменится на **Аннулирован** или **Получен от контрагента** соответственно.

Работа с формализованными документами

В данном разделе рассмотрена работа пользователя с формализованными входящими и исходящими документами с использованием интерфейса модуля *Web-клиент*. В разделе также приводится описание карточек для работы с формализованными документами.

Карточки Docsvision для работы с формализованными документами

Для работы с формализованными документами в модуле предусмотрено

несколько специальных видов карточек, производных от вида *Документ*:

1. *Исходящий УПД* — предназначен для создания исходящих УПД.
 - *Исходящий УКД* — предназначен для создания корректировок к отправленным УПД.
 - *Исходящий УКД (736)* — предназначен для создания корректировок к отправленным УПД, вступившем в силу 27.11.2020.
 - *Исходящее исправление УПД* — предназначен для формирования исправлений отправленных УПД.
 - *Исходящее исправление УПД (820)* — предназначен для формирования исправлений для отправленных УПД в формате, вступившем в силу 1.01.2020.
 - *Исходящий УПД (820)* — предназначен для создания исходящих УПД в формате, вступившем в силу 1.01.2020.
2. *Исходящее приглашение ЭДО* — предназначен для создания исходящих приглашения к обмену ЮЗДО.
3. *Входящий УПД* — предназначен для работы с входящими УПД. Карточки данного вида создаются модулем при получении формализованных документов.
 - *Входящий УКД* — предназначен для работы с корректировками полученных УПД.
 - *Входящий УКД (736)* — предназначен для создания корректировок к отправленным УПД, вступившем в силу 27.11.2020.
 - *Входящий УПД (820)* — предназначен для работы с корректировками полученных УПД в формате, вступившем в силу 1.01.2020.
 - *Формализованный акт* — предназначен для работы с входящими актами и формализованными накладными в рекомендованном ФНС формате.
 - *Формализованный акт (820)* — предназначен для работы с входящими актами и формализованными накладными в рекомендованном ФНС формате, вступившем в силу 1.01.2020.

Для всех документов поддерживаются подтверждения оператора и извещения о получении участников обмена формата 1.03, в соответствии с приказом ФНС России от 21.02.2023 № ЕД-7-26/133@.

Изменения затрагивают следующие технологические документы:

проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- **Запросить аннулирование.** При нажатии будет вызвана функция аннулирования формализованного документа.
 - **Подтвердить аннулирование.** При нажатии будет подтвержден поступивший запрос на аннулирование.
 - **Отказать в аннулировании.** При нажатии будет вызвана функция отказа от аннулирования по поступившему запросу.
2. Список файлов — предназначен для добавления XML-файла счета-фактуры, акта, накладной или УПД.
3. О документе:
- *Состояние документа* — текущее состояние карточки *Документ*.
 - *Номер документа*.
 - *Дата документа*.
 - *Тип документа* — название типа формализованного документа, указанного в полученном электронном документе.
 - *Статус УПД* — текущее состояние подписания УПД:
 - **УПД подписан отправителем** — если сформирован и подписан только титул продавца.
 - **УПД подписан с обеих сторон** — если сформирован и подписан титул продавца и получен ответный подписанный титул покупателя.
- Результат проверки подписи отображается справа от состояния подписания.
- *Функция УПД* — **формат УПД**, заполняется вручную.
4. Области:
- *О документе.* В данном разделе карточки указываются продавец и сумма документа.
 - *Данные о подписанте.* В данном разделе заполняется титул покупателя и данные получателя.
 - *Журнал обмена ЮЗДО*, см. пункт **Журнал обмена ЮЗДО**.

- *Печатная форма*, см. пункт [Печатная форма формализованного документа](#).
- Ссылки — приводятся ссылки на полученные документы, являющиеся для данного документа исправлением или корректировкой, а также ссылка на исправляемый или корректируемый документ, ссылки на другие документы, полученные в одном пакете с данным документом.
- Файлы — содержит основные и дополнительные файлы карточки и кнопку, а также *Журнал подписей*.
 - При нажатии на кнопку **Журнал подписей** будет открыт *Журнал подписей* с информацией об электронных подписях документа.

Исходящий УКД

Карточки вида *Исходящий УКД*, *Исходящий УКД 736* предназначены для корректировки сумм и количества товаров в переданном УПД. Пример карточки приведен на следующем рисунке.

Назад + Создать

УКД
Статус: ☐

1 - корректировочный счет-фактура и соглашение (уведомление)
2 - соглашение (уведомление)

Корректировочный счет-фактура № 5 от 13.10.2022 (1), исправление
от (1a)
к счету-фактуре (счетом-фактурам) № 2 от 12.10.2022 с учетом исправления (16)

Продавец ООО Перспектива
Адрес Санкт-Петербург, Большая Пушкарская, 28
ИНН/КПП продавца 9653547757 / 965301000

Покупатель 5
Адрес Московская область, Коммунальный пр., 21, Дубна
ИНН/КПП покупателя 9736284116 / 742801201
Валюта: наименование, код Российский рубль

№ п/п	Код товара/работ, услуги	Наименование товара (описание выполненных работ, оказанных услуг), имущественного права	Показатели в связи с изменением стоимости отгруженных товаров (выполненных работ, оказанных услуг), переданных имущественных прав	Единица измерения		Количество (объем)	Цена (тариф) за единицу измерения	Стоимость товаров (работ, услуг), имущественных прав без налога - всего
				код	условное обозначение (национальное)			
A	B	1	1a	2a	26	3	4	5
1			А (до изменения)					
			Б (после изменения)					5.00

ПОЛУЧЕН

Исходящий Корректировочный счет-фактура от 13.10.2022

Тип документа: УКД Функция документа: КСЧФ
Статус УПД: Документ подписан отправителем.

ОТПРАВИТЬ ИЗВЕЩЕНИЕ

ПЕЧАТЬ ИСТОРИЯ

О ДОКУМЕНТЕ

Юридическое лицо: [Доксвижн Северо-Запад](#)
Контрагент: [ДоксВижн консалт](#)
Номер документа: 5
Дата документа: 13.10.2022
Дата исправления: 13.10.2022
НДС: 0
Сумма с НДС: 5,00
Валюта:
Комментарий:

ДАННЫЕ О ПОДПИСАНТЕ

Подписант: [kmg01 k. k.](#)
Дата принятия: 14.10.2022
Получатель: [kma01 k. k.](#)

Рисунок 92. Пример исходящего УКД

Список элементов карточки *Исходящий УКД* аналогичен списку элементов карточки [Исходящий УПД](#). Функция УПД заполняется вручную.

Исходящее исправление УПД

Карточки вида *Исходящее исправление УПД* и *Исходящее исправление УПД (820)* предназначены для замены переданного УПД. Пример карточки приведен на следующем рисунке.

Универсальный передаточный документ

Статус: ☐ 1 - счет-фактура и передаточный документ (акт) ☒ 2 - передаточный документ (акт)

Счет-фактура № 1312 от 23.09.2019 (1)
Исправление № 10 от 22.10.2019 (1а)

Г
от 26 декабря 2019
к постановлению Правительства Российской Федерации

Продавец
Адрес
ИНН КПП продавца
Грузополучатель и его адрес
Грузополучатель и его адрес
К платежно-расчетному документу № 2015 от 04.09.2019
Покупатель
Адрес
ИНН КПП покупателя
Валюта: наименование, код

ООО Перспектива
Санкт-Петербург, ул. Горького, 32 к2, офис 73
96455967757 : 964301000
Санкт-Петербург, ул. Горького, 32 к2, офис 73
ООО Пингвин, Московская область, Коммунальный пр., 21, Дубина
№ 2015 от 04.09.2019
ООО Пингвин
Коммунальный пр., Московская область, 21, Дубина
96431313085 : 945701000
Российский рубль

N п	Код товара по АТн	Наименование товара (описание выполняемых работ, оказанных услуг), инвентаризационного права	Единица измерения		Количество (объем)	Цена (тариф) за единицу измерения	Стоимость товаров (работ, услуг), инвентаризационных прав без налога - всего	В том числе сумма акцизов	Налоговая ставка	Сумма налога, подлежащая уплате покупателем	Стоимость товаров (работ, услуг), инвентаризационных прав с налогом - всего	Страна происхождения товара	
			код	условное обозначение (национальное)								цифровой код	краткое наименование
A	B	1	2a	2б	3	4	5	6	7	8	9	10a	10б
1		Паллет 1200x800	шт		27	1481.48	40000.00	без акциза	НДС	без НДС	43000.00		
2		Паллет 1000x800	шт		41	1097.56	45000.00	без акциза	НДС	без НДС	45000.00		
						Всего к оплате	\$5000.00	x	x	без НДС	\$8000.00		
Руководитель организации или иное уполномоченное лицо Индивидуальный предприниматель			электронная подпись (подпись)		тестовый сертификат (ф.и.о.)	Главный бухгалтер или иное уполномоченное лицо (подпись)		(ф.и.о.)					

Основание передачи («лав») / получения (приемки) _____ (договор, доверенность и др.)
Данные о транспортировке и грузе: ООО Неолит Северо-Запад, Санкт-Петербург, Большая Пушкарская, 28

Исходящий Исправление №10 от 22.10.2019 для УПД №1312 от 23.09.2019

Тип документа: УПД
Статус УПД: Документ не отправлен.

Функция документа: УПД

ПОДПИСАТЬ И ОТПРАВИТЬ ОТПРАВИТЬ

ПЕЧАТЬ ИСТОРИЯ

О ДОКУМЕНТЕ

Юридическое лицо:
Контрагент:
Номер документа:
Дата документа:
НДС:
Сумма с НДС:
Валюта:
Автор:
Комментарий:

ООО Перспектива
ООО Пингвин
1312
23.09.2019

88 000,00
Российский рубль
Мясницкий Н.Р.

ДАнные О ПОДПИСАНТЕ

Подписант:
Сведения о подписанте:
Область полномочий:
Организация подписанта:
Статус подписанта:
Основание полномочий подписанта:
Основание полномочий организации:
Иные сведения:

Сидоров Е.С.
Индивидуальный предприниматель
Лицо, ответственное за оформление свершившегося события
ООО Неолит
Работник иной уполномоченной организации
Доверенность №022323
Устав ООО Неолит

Рисунок 93. Пример исходящего исправления УПД

Список элементов карточки *Исходящее исправление УПД* аналогичен списку элементов карточки *Исходящий УПД*.

Входящий УПД

Карточки вида *Входящий УПД* и *Входящий УПД (820)* предназначены для загрузки в Docsvision счетов-фактур, УПД и исправлений к счетам-фактурам и УПД. Пример карточки приведен на следующем рисунке.

		Продавец №1 и клиент №2 Исправление №10		от 23.09.2022 (1) от 22.10.2022 (1а)		Г																					
Универсальный передаточный документ Статус: ☐ 1 - счет-фактура и передаточный документ (акт) 2 - передаточный документ (акт)		Счет-фактура № 1312 Исправление № 10		от 23.09.2022 (1) от 22.10.2022 (1а)		к постановлению Правительства Российской Федерации от 26 декабря 2010 года № 940																					
Продавец Адрес ИНН КПП продавца Грузоотправитель и его адрес Грузополучатель и его адрес К платежизначальному документу		ООО Пантоне Санкт-Петербург, Большая пушкарская, 28, 197198 964394757 / 964301000 ООО Pantone Spedition Co ООО Перспектива																									
Покупатель Адрес ИНН КПП покупателя Валюта: наименование, код		ООО Перспектива Санкт-Петербург, Большая морская, 12, 191186 9676313085 / 9676451000 Российский рубль																									
N п/п		Код товара по классификации товаров, работ, услуг, оказанных (услуг), имущественного права		Единица измерения		Количество (объем)		Цена (тариф) за единицу измерения		Стоимость товаров (работ, услуг), имущественных прав без налога - всего		В том числе сумма акциза		Налоговая ставка		Сумма налога, предъявленного покупателю		Стоимость товаров (работ, услуг), имущественных прав с налогом - всего		Страна происхождения товара							
A		B		1		2a		2b		3		4		5		6		7		8		9		10a		10b	
1				Оборудование для фиксации марки "STlab"		796шт		45		464.00		20880.00		без акциза		без НДС		20880.00									
										Всего к оплате		20880.00		x		x		без НДС		20880.00							
				Руководитель организации или иное уполномоченное лицо Индивидуальный предприниматель		электронная подпись (подпись)		Масинский Константин Андреевич (ф.и.о.)		Главный бухгалтер или иное уполномоченное лицо (подпись)																	
				Основание передачи (сдачи) / получения (приемки)		Без документа-основания																					
				Данные о транспортировке и грузе																							
				Товар (груз) передат / услуги, результаты работ, права сдал																							
				Товар (груз) получил / услуги, результаты работ, права принял																							

Входящий Счёт на поставку от 15.02.2023

Тип документа: УПД
Статус УПД: Документ подписан отправителем.

Функция документа: УПД

[ПОДПИСАТЬ](#) [ОТКАЗАТЬ](#) [ОТПРАВИТЬ ИЗВЕЩЕНИЕ](#)

[ПЕЧАТЬ](#) [ИСТОРИЯ](#)

Юридическое лицо:
Контрагент:
Номер документа:
Дата документа:
Дата исправления:
НДС:
Сумма с НДС:
Валюта:
Комментарий:

Доксвижн Северо-Запад
ДоксВижн консалт
520123
15.02.2023
15.02.2023
НДС:
20 880,00
RUB

Данные о подписанте

Подписант:
Дата принятия:
Получатель:
Организация о подписанте:
Область полномочий:
Организация подписант:
Статус подписант:
Основание полномочий подписант:
Основание полномочий организации:
Основание полномочий получателя:

Колесников А.Г.
15.02.2023
Колесников А.Г.
Индивидуальный предприниматель
Лицо, ответственное за подписание счетов-фактур
ООО Перспектива
Работник организации покупателя
Должностные обязанности
Устав организации от 17.03.2012
Доверенность №1 от 23.12.2022

Рисунок 94. Пример входящего УПД

Карточка содержит следующие элементы:

1. Лента карточки — в зависимости от статуса обмена может содержать кнопки:

- **Сохранить.**
- **Печать.** При нажатии будет вызвана функция экспорта и печати карточки.
- **Подписать документ.** При нажатии будет вызвана функция подписания и отправки титула покупателя.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- **Отказать в подписании.** При нажатии будет вызвана функция отказа в подписании с указанием причины отказа.
- **Запросить аннулирование.** При нажатии будет вызвана функция аннулирования формализованного документа.
- **Подтвердить аннулирование.** При нажатии будет подтвержден поступивший запрос на аннулирование.
- **Отказать в аннулировании.** При нажатии будет вызвана функция отказа от аннулирования по поступившему запросу.
- **Запросить уточнение.** При нажатии будет вызвана функция запроса исправления документа с указанием комментария.

2. О документе:

- *Состояние документа* — текущее состояние карточки *Документ*.
- *Номер документа*.
- *Дата документа*.
- *Тип документа* — название типа формализованного документа, указанного в полученном электронном документе.
- *Статус УПД* — текущее состояние подписания УПД:
 - **УПД подписан отправителем** — если сформирован и подписан только титул продавца.

- **УПД подписан с обеих сторон** — если сформирован и подписан титул продавца и получен ответный подписанный титул покупателя.

Результат проверки подписи отображается справа от состояния подписания.

- *Функция УПД* — **формат УПД**, заполняется вручную.

3. Области:

- *О документе*. В данном разделе карточки указываются продавец и сумма документа.
- *Данные о подписанте*. В данном разделе заполняется титул покупателя и данные получателя.
- *Журнал обмена ЮЗДО*, см. пункт **Журнал обмена ЮЗДО**.
- *Печатная форма*, см. пункт **Печатная форма формализованного документа**.
- *Ссылки* — приводятся ссылки на полученные документы, являющиеся для данного документа исправлением или корректировкой, а также ссылка на исправляемый или корректируемый документ, ссылки на другие документы, полученные в одном пакете с данным документом.
- *Файлы* — содержит основные и дополнительные файлы карточки и кнопку, а также *Журнал подписей*.
 - При нажатии на кнопку **Журнал подписей** будет открыт *Журнал подписей* с информацией об электронных подписях документа.

Входящий УКД

Карточки вида *Входящий УКД* и *Входящий УКД (736)* предназначены для загрузки в Docsvision исправлений к входящим УПД. Пример карточки приведен на следующем рисунке.

Рисунок 95. Пример входящего УКД

Формализованный акт

148

- **Отказать в аннулировании.** При нажатии будет вызвана функция отказа от аннулирования по поступившему запросу.

2. О документе:

- *Состояние документа* — текущее состояние карточки *Документ*.
- *Номер документа*.
- *Дата документа*.
- *Тип документа* — название типа формализованного документа, указанного в полученном электронном документе.
- *Статус УПД* — текущее состояние подписания УПД:
 - **УПД подписан отправителем** — если сформирован и подписан только титул продавца.
 - **УПД подписан с обеих сторон** — если сформирован и подписан титул продавца и получен ответный подписанный титул покупателя.

Результат проверки подписи отображается справа от состояния подписания.

- *Функция УПД* — **формат УПД**, заполняется вручную.

3. Области:

- *О документе*. В данном разделе карточки указываются продавец и сумма документа.
- *Данные о подписанте*. В данном разделе заполняется титул покупателя и данные получателя.
- *Журнал обмена ЮЗДО*, см. пункт **Журнал обмена ЮЗДО**.
- *Печатная форма*, см. пункт **Печатная форма формализованного документа**.
- *Ссылки* — приводятся ссылки на полученные документы, являющиеся для данного документа исправлением или корректировкой, а также ссылка на исправляемый или корректируемый документ, ссылки на другие документы, полученные в одном пакете с данным документом.
- *Файлы* — содержит основные и дополнительные файлы карточки и кнопку, а также *Журнал подписей*.
 - При нажатии на кнопку **Журнал подписей** будет открыт *Журнал подписей* с информацией об электронных подписях документа.

Работа с исходящими документами

Работа с исходящими документами включает следующие этапы:

1. Пользователь модуля (далее — *Продавец*) создаёт исходящий документ вида (или подвида) *Исходящий УПД* и отправляет его контрагенту (далее — *Покупатель*).
2. Оператор ЭДО принимает документ от *Продавца* и отправляет ему *Извещение* о получении, а также передаёт документ *Покупателю*.
3. *Продавец* получает *Извещение* о получении от оператора ЭДО и подтверждает его получение. Подтверждение не требуется для актов и накладных.
4. После получения документа *Покупатель* может:
 - Подписать или отказать в подписании документа — не требуется для счёт-фактуры.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- Запросить аннулирование — для счёт-фактуры.
 - Запросить уточнение у *Продавца*.
5. *Продавец* получает ответ от *Покупателя*.

Если "*Покупатель*" подписал документ (был сформирован титул покупателя):

 - a. Титул покупателя добавляется в карточку *Исходящего УПД* (или подвида).
 - b. Карточка и статус обмена переходят в состояние **Получена подпись от контрагента**.
 - c. Документооборот завершен.
 - Если *Покупатель* отказал в подписании документа, карточка и статус обмена переходят в состояние "Получен отказ от контрагента".
 - Если *Покупатель* запросил уточнение, статус обмена переходит в

состояние **Требуется подтверждение получения**, а далее в **Требуется уточнение**. После этого работа с данным документом завершается. Пользователь модуля может создать корректировочный УПД или исправление УПД.

6. *Покупатель* или *Продавец* могут запросить аннулирование подписанного документа. Вторая сторона ЭДО может подтвердить или отказать в аннулировании.

Если *Продавец* запросил аннулирование до получения извещения от оператора ЭДО (о получении документа от *Продавца*), документ будет аннулирован в одностороннем порядке (без подтверждения со стороны *Покупателя*).

Отправка УПД

Отправка УПД осуществляется с помощью карточки вида *Исходящий УПД*. Для создания исходящего УПД нужен XML файл сформированной **счет-фактуры**, **акта**, **накладной** или **УПД**.

1. Создайте новую карточку вида **Документ/УПД/Исходящий УПД**.



С 1 января 2020 года согласно **приказу ФНС РФ от 19.12.2018 N ММВ-7-15/820@** в силу вступил новый формат УПД. Вместо карточек *Исходящий УПД* нужно создавать карточки вида **Документ/УПД/Исходящий УПД/Исходящий УПД (820)**.

Рисунок 97. Карточка *Исходящий УПД*

2. Добавьте в карточку файл отправляемого универсального передаточного документа в формате XML:

- Перетащите файл из файловой системы в область *Файлы* карточки.
- Нажмите в области *Файлы* левой кнопкой мыши и выберите файл из файловой системы.
- Нажмите на элемент управления , выберите файл из файловой системы.

Если добавленный файл имеет ошибки или некорректный формат, будет выведено предупреждение.

3. При загрузке основного файла и новой версии файла в карточку *Исходящего УПД* и подвидов, следующие поля карточки будут заполнены автоматически:

Автор, Номер документа, Дата документа, НДС, Сумма с НДС, Функция документа.

4. Если указанный в файле контрагент зарегистрирован в *Справочнике контрагентов* (должен совпадать КПП/ИНН), поле *Контрагент* также будет заполнено автоматически.

Если поле не было заполнено автоматически, выберите сотрудника контрагента в поле *Контрагент*. Убедитесь, что в поле *Организация контрагента* указан корректный получатель данного электронного документа.

5. Заполните *титул продавца* в разделе *Данные о подписанте*.

6. При необходимости добавьте ссылки на другие документы.

7. Нажмите кнопку **Сохранить** на ленте карточки. Карточка будет открыта в режиме чтения.

8. Нажмите кнопку **Подписать и отправить**, будет открыто меню выбора сертификата.

Назад + Создать ИИ

Примечание: В 1-м разделе: ФНС России от 21.10.2013 N АЗСБ-20-3-06/08

Счет-фактура № 1312 от 23.09.2019 (1)
Исправление № 10 от 22.10.2019 (1a)

к постановлению Правительства Российской Федерации от 26 декабря 2011 г. N 1271-ПП

Универсальный передаточный документ
Статус: ☐ 1 - счет-фактура и передаточный документ (акт)
☐ 2 - передаточный документ (акт)

Продавец
ООО Перспектива
Адрес: Санкт-Петербург, Большая пушкарская, 28
ИНН/КПП продавца: 780401759 / 7804017400
Грузоотправитель и его адрес: ООО Пантеон, Г. Москва, в.тер. г. Муниципальный Округ Арбат, ул. Старый Арбат, дом 32, помещен. комнаты 8
К платежно-расчетному документу № 2015 от 04.09.2019

Покупатель
ООО Перспектива
Адрес: Санкт-Петербург, Большая пушкарская, 28
ИНН/КПП покупателя: 780401759 / 7804017400
Валюта: наименование, код: Российский рубль

N п	Код товара/услуг	Наименование товара (описание работ, оказанных услуг), имущественного права	Единица измерения	Количество (объем)	Цена (тариф) за единицу измерения	Стоимость товаров (работ, услуг), имущественных прав без налога - всего	В том числе сумм акциза	Налоговая ставка	Сумма налога, подлежащая уплате покупателем	Стоимость товаров (работ, услуг), имущественных прав с налогом - всего	Страна происхождения товара	
											10a	10b
1		Стулья офисные модель "Перспектива 17"	шт	8	5000.00	40000.00	без акциза	НДС	без НДС	43000.00		
2		Стулья офисные модель "Перспектива 15"	шт	5	8000.00	40000.00	без акциза	НДС	без НДС	43000.00		
Всего:						80000.00	х	х	без НДС	88000.00		

Руководитель организации или иное уполномоченное лицо: Мясницкий Н.Р. (подпись) (ф.и.о.)
Генеральный директор или иное уполномоченное лицо: Нечаев С.Д. (подпись) (ф.и.о.)
Индивидуальный предприниматель: (подпись) (ф.и.о.) (реквизиты свидетельства о государственной регистрации индивидуального предпринимателя)

Основание полномочия (ссылка) / получения (получения): Ответствен

НА ПОДГОТОВКЕ

Исходящий УПД №10 от 22.10.2019 к счѣт-фактуре №1312 от 23.09.2019

Тип документа: УПД
Статус УПД: Документ не отправлен.

Подписать и отправить
ПЕЧАТЬ ИСТОРИЯ

Юридическое лицо: Дюкович Северо-Запад
Контрагент: Тестовая организация №4650615
Номер документа: 1312
Дата документа: 23.09.2019
НДС: 88 000,00
Сумма с НДС: 88 000,00
Валюта: Мясницкий Н.Р.
Автор: Мясницкий Н.Р.
Комментарий:

ДАННЫЕ О ПОДПИСАНТЕ

Подписант: Нечаев С.Д.
Сведения о подписанте: Юридическое лицо
Область полномочий: Лицо, ответственное за оформление свершившегося события
Организация подписанта: Электрогенерация
Статус подписанта: Работник организации - составителя информации покупки
Основание полномочий подписанта: Работник организации - составителя информации покупки
Основание полномочий организации: Работник организации - составителя информации покупки
Иные сведения:

Рисунок 98. Подписание и отправка

Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

- Подпишите файлы при помощи действительного сертификата ЭП.
- При необходимости выберите другое подразделение и ящик ЮЗДО, через который будет отправлен электронный документ. Нажмите кнопку **ОК**. Будет открыто окно выбора сертификата для подписания документа.

Выберите вариант подписания

Подписать без метки

Выберите сертификат

Petrov A.F.; Panteon Neolit Certification 2; 03.05.2023

Настройки способов подписи

Просмотр доверенности

Подписать

Отменить

Рисунок 99. Диалог выбора сертификата для подписания

- Выберите сертификат квалифицированной подписи и нажмите кнопку **Отправить**. Будет показано сообщение **Отправка успешно выполнена**. Состояние документа изменится на **Подписан и отправлен**.

После получения документа оператором ЭДО нужно будет **подтвердить получение извещения** о приёме документа.

После подписания документа контрагентом (формируется *титул покупателя*) документ перейдет в статус **УПД подписан с обеих сторон**. На этом документооборот считается завершенным.

Для счёт-фактуры подписание контрагентом не требуется. Документооборот будет завершен после **подтверждения получения извещения**. Статус обмена: **Отправлен контрагенту**.

Обмен информацией по данному документу можно отслеживать с помощью **Журнала обмена ЮЗДО**.

Корректировка УПД (создание УКД)

Корректировка цены или количества товара в отправленной УПД выполняется с помощью карточки вида *Исходящий УКД*. Для создания корректировочного документа в Docsvision нужен файл отправляемого универсального корректировочного документа в **формате XML**.

- Создайте новую карточку вида **Документ/УПД/Исходящий УПД/Исходящий УКД**.

Создать

Исходный

Получен

УКД

Статус: ☐ 1 - корректировочный счет-фактура и соглашение (уведомление) 2 - соглашение (уведомление)

Корректировочный счет-фактура № 5 от 13.10.2022 (1), исправление корректировочного счета-фактуры № 1 от 12.10.2022 (1a) к счету-фактуре (счетам-фактурам) № 2 от 12.10.2022 с учетом исправления № от (16) Продавец ООО Перспектива Адрес Санкт-Петербург, ул. Горького, 32 к2, офис 73 ИНН КПП продавца 9632177757 / 915641000 Покупатель ООО Пантеон Адрес Московская область, Коммунальный пр., 21, Дубна ИНН КПП покупателя 9126527440 / 79821001 Валюта: наименование, код Российский рубль

Примечание № 1 к письму ФНС России от _____ № _____ Приложение № 2 к постановлению Правительства Российской Федерации от 26 декабря 2011 г. № 1137

Исходящий УКД №5 от 13.10.2022 Тип документа: УКД Функция документа: УПД Статус: УПД. Документ подписан отправителем.

Отправить извещение

Печать История

О документе

Юридическое лицо: ООО Перспектива
Контрагент: ООО Пантеон
Номер документа: 5
Дата документа: 13.10.2022
Дата исправления: 13.10.2022
СДС: 0
НДС: 20.00
Валюта: Российский рубль
Комментарий:

Данные о подписанте

Подписант: Мясницкий Н.Р.
Дата принятия: 14.10.2022
Получатель: Николай А.Д.
Сведения о подписанте: Юридическое лицо
Лицо, ответственное за оформление свершившегося события: Директор Северо-Запад
Статус подписанта: Работник иной уполномоченной организации
Основание полномочий подписанта: Доверенность №1 от 12.02.2023
Основание полномочий организации: Устав ООО Перспектива
Основание полномочий получателя: Доверенность №1 от 02.02.2023

A	B	1	1a	Единица измерения		3	4	5	6	7	8	9
				2a	2b							
1		Паллеты 1200x800	40 (до изменения)				1200.00		без анализа	без НДС	без НДС	без НДС
			41 (после изменения)				1200.00	1200.00	без анализа	без НДС	без НДС	1200.00
			1 (увеличение)	x	x	x	x	1200.00	x	x	x	2.98
			x	x	x	x	x	x	x	x	2.00	x
							Всего увеличение (сумма строк В):	1200.00	x	x	0.00	2.98
							Всего уменьшение (сумма строк Г):	0.00	x	x	2.00	0.00
		Руководитель организации или иное уполномоченное	электронная подпись	Мясницкий Николай Романович	Главный бухгалтер или иное уполномоченное	электронная подпись	Алексеев Сергей Леонидович					

2. Добавьте в карточку файл отправляемого универсального корректировочного документа в формате XML:

Если добавленный файл имеет ошибки или некорректный формат, будет выведено предупреждение.

- При загрузке основного файла и новой версии файла в карточку *Исходящего УПД* и подвидов, следующие поля карточки будут заполнены автоматически:
- Нажмите кнопку **Заполнить данные из файла** на ленте карточки.

Номер документа, Дата документа, Номер корректировки, Дата корректировки, Стоимость с НДС и Сумма НДС.

5. Если указанный в файле контрагент зарегистрирован в *Справочнике контрагентов* (должен совпадать КПП/ИНН), поле *Контрагент* также будет заполнено автоматически.

Если поле не было заполнено автоматически, выберите сотрудника контрагента в поле *Контрагент*. Убедитесь, что в поле *Организация*

контрагента указан корректный получатель данного электронного документа.

6. Заполните *титул продавца* в разделе *Данные о подписанте*.
7. При необходимости добавьте ссылки на другие документы.
8. Нажмите кнопку **Сохранить** на ленте карточки. Карточка будет открыта в режиме чтения.
9. Нажмите кнопку **Подписать и отправить**, будет открыто меню выбора сертификата.

Исходящий УКД №5 от 13.10.2022

Тип документа: УКД
Статус УПД: Документ не отправлен.
Функция документа: УПД

Подписать и отправить | Отправить

Печать | История

О ДОКУМЕНТЕ

Юридическое лицо: ООО Перспектива
Контрагент: ООО Пантеон
Номер документа: 5
Дата документа: 13.10.2022
Дата исправления: 13.10.2022
НДС: 0
Сумма с НДС: 1200.00
Валюта: Российский рубль
Комментарий:

ДАННЫЕ О ПОДПИСАНТЕ

Подписант: Мяснищев Н.Р.
Дата принятия: 14.10.2022
Получатель: Николай А.Д.
Сведения о подписанте: Лицо, ответственное за оформление свершившегося события
Область полномочий: Докладчик Северо-Запад
Организация подписанта: Работник иной уполномоченной организации
Статус подписанта: Доверенность №1 от 12.02.2023
Основание полномочий подписанта: Устав ООО Перспектива
Основание полномочий организации: Доверенность №1 от 02.02.2023

№ п/п	Код товара	Наименование товара (описание выполненных работ, оказанных услуг), имущественного права	Показатели в связи с изменением стоимости отгруженных товаров (выполненных работ, оказанных услуг), переданных имущественных прав	Единица измерения	Количество (объем)	Цена (тариф) за единицу измерения	Стоимость товаров (работ, услуг), имущественных прав без налога - всего	В том числе сумма акциза	Налоговая ставка	Сумма налога	Стоимость товаров (работ, услуг), имущественных прав с налогом - всего	
А	Б	1	1а	2а	2б	3	4	5	6	7	8	9
1		Паллеты 1200x800	40 (до изменения)			1200.00			без НДС	без НДС	без НДС	
			41 (после изменения)			1200.00	1200.00		без НДС	без НДС	без НДС	1200.00
		1 (увеличение)	x	x		x	x	1200.00	x	x	x	1200.00
		x	x	x		x	x		x	x	x	
			Всего увеличение (сумма строк В):				1200.00		x	x	0.00	1200.00
			Всего уменьшение (сумма строк Г):				0.00		x	x	0.00	

Руководитель организации или иное уполномоченное лицо: Мяснищев Николай Романович
Главный бухгалтер или иное уполномоченное лицо: Алексеев Сергей Леонидович

Рисунок 101. Подписание и отправка

Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.



ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

10. Подпишите файлы при помощи действительного сертификата ЭП.
11. При необходимости выберите другое подразделение и ящик ЮЗДО, через который будет отправлен электронный документ. Нажмите кнопку **ОК**. Будет открыто окно выбора сертификата для подписания документа.

Выберите вариант подписания

Подписать без метки

Выберите сертификат

Petrov A.F.; Panteon Neolit Certification 2; 03.05.2023

Настройки способов подписи

Просмотр доверенности

Подписать

Отменить

Рисунок 102. Диалог выбора сертификата для подписания

- Выберите сертификат квалифицированной подписи и нажмите кнопку **Отправить**. Будет показано сообщение **Отправка успешно выполнена**. Состояние документа изменится на **Подписан и отправлен**.

После получения документа оператором ЭДО нужно будет **подтвердить получение извещения** о приёме документа.

Исправление УПД

Если в отправленном УПД были выявлены ошибки (например, неверно указан адрес), нужно сформировать и отправить исправление УПД. Исправление выполняется с помощью карточек вида *Исходящее исправление УПД* и *Исходящее исправление УПД (820)*. Для создания исправленного документа в Docsvision нужен файл отправляемого исправленного универсального передаточного документа в формате XML.

- Создайте новую карточку вида **Документ/УПД/Исходящий УПД/Исходящее исправление УПД**.



С 1 января 2020 года согласно [приказу ФНС РФ от 19.12.2018 N ММВ-7-15/820@](#) в силу вступил новый формат УПД. Вместо карточек *Исходящий УПД* нужно создавать карточки вида **Документ/УПД/Исходящий УПД/Исходящий УПД (820)**.

Рисунок 103. Карточка "Исходящее исправление УПД"

2. Добавьте в карточку файл отправляемого исправления универсального передаточного документа в формате XML:
 - Перетащите файл из файловой системы в область *Файлы* карточки.
 - Нажмите в области *Файлы* левой кнопкой мыши и выберите файл из файловой системы.
 - Нажмите на элемент управления , выберите файл из файловой системы.

Если добавленный файл имеет ошибки или некорректный формат, будет выведено предупреждение.

3. При загрузке основного файла и новой версии файла в карточку *Исходящего УПД* и подвидов, следующие поля карточки будут заполнены автоматически:

Номер документа, Дата документа, Номер исправления, Дата исправления, Стоимость с НДС и Сумма НДС будут заполнены автоматически.

4. Если указанный в файле контрагент зарегистрирован в *Справочнике контрагентов* (должен совпадать КПП/ИНН), поле *Контрагент* также будет заполнено автоматически.

Если поле не было заполнено автоматически, выберите сотрудника контрагента в поле *Контрагент*. Убедитесь, что в поле *Организация*

контрагента указан корректный получатель данного электронного документа.

5. Заполните титул продавца в разделе *Данные о подписанте*.
6. При необходимости добавьте ссылки на другие документы.
7. Нажмите кнопку **Сохранить** на ленте карточки. Карточка будет открыта в режиме чтения.
8. Нажмите кнопку **Подписать и отправить**, будет открыто меню выбора сертификата.

Рисунок 104. Подписание и отправка



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

9. Подпишите файлы при помощи действительного сертификата ЭП.
10. При необходимости выберите другое подразделение и ящик ЮЗДО, через который будет отправлен электронный документ. Нажмите кнопку **ОК**. Будет открыто окно выбора сертификата для подписания документа.

Выберите вариант подписания

Подписать без метки

Выберите сертификат

Petrov A.F.; Panteon Neolit Certification 2; 03.05.2023

Настройки способов подписи

Просмотр доверенности

Подписать

Отменить

Рисунок 105. Диалог выбора сертификата для подписания

- Выберите сертификат квалифицированной подписи и нажмите кнопку **Отправить**. Будет показано сообщение **Отправка успешно выполнена**. Состояние документа изменится на **Подписан и отправлен**.

После получения документа оператором ЭДО нужно будет **подтвердить получение извещения** о приёме документа.

После подписания документа контрагентом (формируется *титул покупателя*) документ перейдет в статус **УПД подписан с обеих сторон**. На этом документооборот считается завершенным.

Для счёт-фактуры подписание контрагентом не требуется. Документооборот будет завершен после **подтверждения получения извещения**. Статус обмена: **Отправлен контрагенту**.

Обмен информацией по данному документу можно отслеживать с помощью **Журнала обмена ЮЗДО**.

Работа с входящими документами

Подписание полученного формализованного документа

Пользователь модуля должен принять решение по поступившему на подпись документу: подписать или отказать.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в

корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

1. Откройте документ со статусом обмена в [Журнале обмена ЮЗДО](#): **Получен на подпись от контрагента**.
2. Заполните информацию о покупателе на странице *Подписание*. Не требуется для отказа в подписании.
3. Нажмите кнопку на ленте карточки для принятия решения по документу:
 - *Подписать документ* — для подписания документа. При этом нужно будет выбрать сертификат квалифицированной подписи, которым будут подписаны файлы документа.

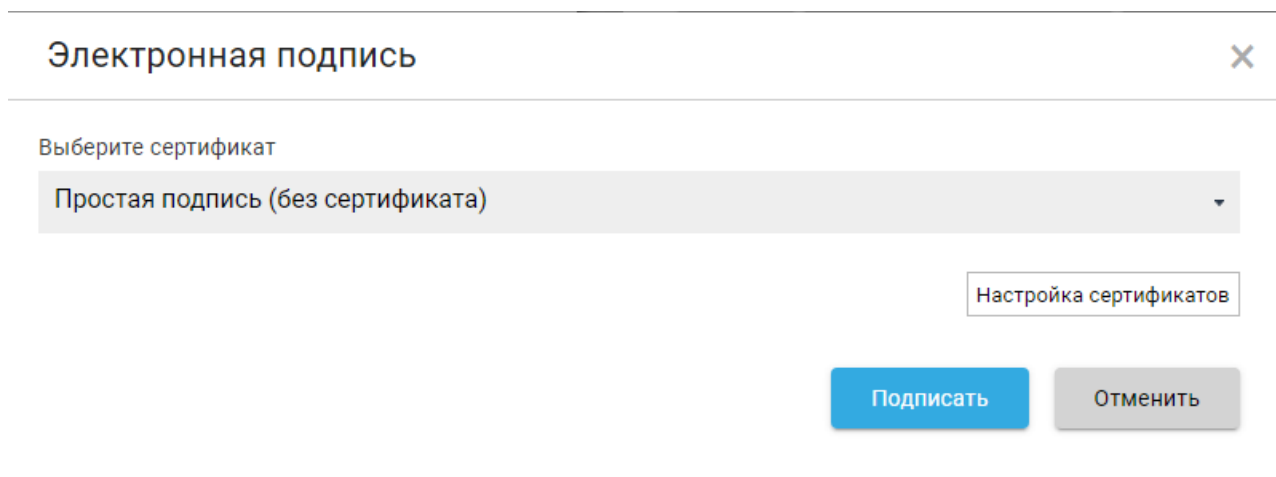


Рисунок 106. Выбор сертификата электронной подписи

- *Отказать в подписании* — для отказа от подписания полученного документа. При этом нужно будет указать причину отказа.



Рисунок 107. Указание причины отказа от подписания документа

Ответная подпись (или отказ) будет отправлена контрагенту после

обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). После этого статус документа сменится на *Отправлена ответная подпись контрагенту* или *Отправлен отказ от подписи контрагенту* соответственно.

Запрос на уточнение

Покупатель может запросить у продавца уточнение полученного документа, если в документе были обнаружены ошибки. Запрос может быть выполнен после *подтверждения получения извещения*.

После отправки запроса на уточнение возможность подписания и аннулирования документа будет заблокирована — дальнейший документооборот с использованием данного документа будет приостановлен. Если документ подписан или подписание не требуется, возможность аннулирования после отправки запроса не блокируется.

Чтобы отправить запрос на уточнение:

1. Откройте полученный документ.
2. Нажмите кнопку **Запросить уточнение** на ленте карточки. Будет предложено указать комментарий к уточнению.

Счет-фактура № 1312 от 23.09.2019 (1)
Исправление № 10 от 22.10.2019 (1a)

Универсальный передаточный документ
Статус: ☐ 1 - счет-фактура и передаточный документ (акт)
☐ 2 - передаточный документ (акт)

Продавец
Адрес: ООО Пантеон, Санкт-Петербург, Большая Пушкарская, 28, 197198
ИНН КПП продавца: 9643947757 / 964301000
Грузоотправитель и его адрес: ООО Pantoon Spedition Co
Грузополучатель и его адрес: ООО Перспектива
К платежно-расчетному документу: ООО Перспектива

Покупатель
Адрес: ООО Перспектива, Санкт-Петербург, Большая морская, 12, 191186
ИНН КПП покупателя: 9676313083 / 9676451000
Валюта: наименование, код: Российский рубль

№ п/п	Код товара	Наименование товара (описание выполняемых работ, оказанных услуг), наименование права	Единица измерения	Количество (объем)	Цена (тариф) за единицу измерения	Стоимость товаров (работ, услуг), суммарная без налога	В том числе сумма акциза	Налоговая ставка	Сумма налога, предъявляемая покупателю	Стоимость товаров (работ, услуг), суммарная с налогом	Страна происхождения товара	Цифровой код	Краткое наименование
А	Б	1	2а	2б	3	4	5	6	7	8	9	10а	10б
1		Паллет 1200х800				40000.00	без НДС	без НДС	43000.00				
2		Паллет 1000х800				45000.00	без НДС	без НДС	45000.00				
Всего к оплате:						\$5000.00	х	х	без НДС	\$8000.00			

Руководитель организации или иное уполномоченное лицо: ООО Неолит Северо-Запад, Санкт-Петербург, Большая Пушкарская, 28
Индивидуальный предприниматель: (подпись) (ф.и.о.)

Основание передачи (сдачи) / получения (приема): Отсутствует (договор, доверенность и др.) [8]

Данные о транспортировке и грузе: ООО Неолит Северо-Запад, Санкт-Петербург, Большая Пушкарская, 28 [9]

Входящий Исправление №10 от 22.10.2022 к счет-фактуре №13 12 от 23.09.2022

Тип документа: УПД
Статус: УПД. Документ подписан отправителем.

Функция документа: УПД

ЗАПРОСИТЬ УТОЧНЕНИЕ ПЕЧАТЬ

ИСТОРИЯ

О ДОКУМЕНТЕ

Юридическое лицо: ООО Перспектива
Контрагент: ООО Пантеон
Номер документа: 1312
Дата документа: 23.09.2019
НДС: 88 000,00
Сумма с НДС: 88 000,00
Валюта: Российский рубль
Автор: Мясницкий Н.Р.
Комментарий:

ДАННЫЕ О ПОДПИСАНТЕ

Подписант: Сидоров Е.С.
Сведения о подписанте: Индивидуальный предприниматель
Область полномочий: Лицо, ответственное за оформление свершившегося события
Организация подписанта: ООО Неолит
Статус подписанта: Работник иной уполномоченной организации
Основание полномочий подписанта: Доверенность №022323
Основание полномочий организации: Устав ООО Неолит
Иные сведения:

Рисунок 108. Запрос уточнения

3. Укажите комментарий и нажмите кнопку **ОК**. Будет предложено выбрать сертификат для подписания запроса на уточнение.
4. Выберите квалифицированный сертификат и нажмите **ОК**.

Запрос будет отправлен контрагенту после обработки процессами модуля (БП

Отправка сообщений ЮЗДО). После этого статус документа сменится на **Отправлен запрос на уточнение**.

Контрагент может создать исправление или корректировку документа. Исправление или корректировка поступит в виде отдельного входящего документа, ссылка на который будет добавлена в документ, по которому было запрошено уточнение.

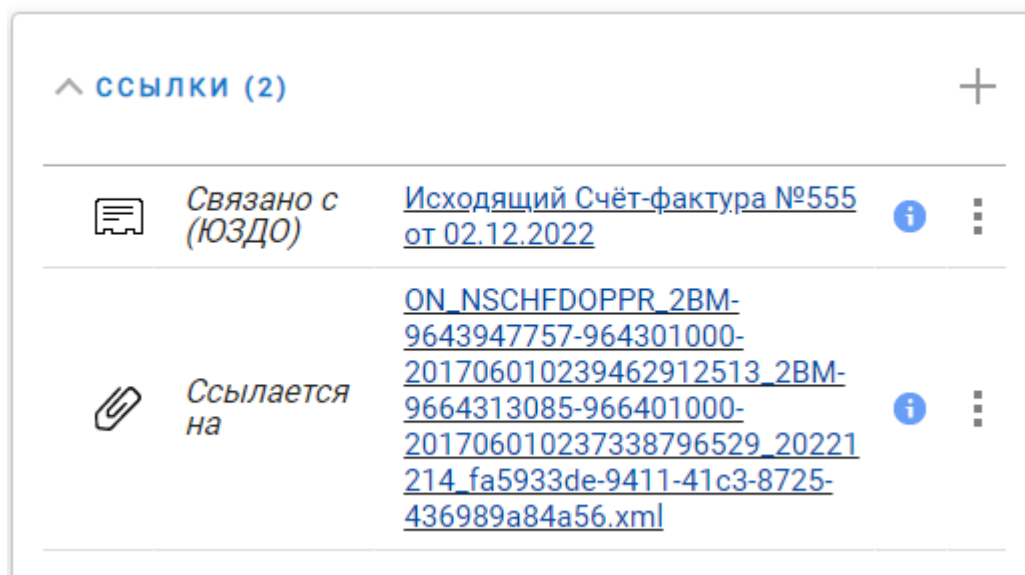


Рисунок 109. Ссылка на корректирующий документ в УПД

Подтверждение получения извещения

Оператор ЭДО формирует извещение, когда передаёт организации электронный документ или получает электронный документ от организации. При загрузке данного извещения в систему Docsvision статус обмена (на вкладке *Журнал обмена ЮЗДО*) изменится на **Требуется подтверждение получения**. Пользователь должен отправить подтверждение получения данного извещения.



ПОДПИСАН И ОТПРАВЛЕН

Входящий Исправление №10 от 22.10.2022 к счёт-фактуре №13 12 от 23.09.2022

Тип документа: УПД

Функция документа: УПД

Статус УПД: Требуется подтверждение получения

ОТПРАВИТЬ ИЗВЕЩЕНИЕ

ПЕЧАТЬ

ИСТОРИЯ

Рисунок 110. Статус обмена электронными документами

В зависимости от этапа жизненного цикла документа определенные операции на ленте карточки станут доступны только после отправки подтверждения получения извещения.

Чтобы подтвердить получение извещения:

1. Откройте карточку формализованного документа, по которому нужно отправить подтверждение.
2. Перейдите на вкладку *Журнал обмена ЮЗДО*.
3. Нажмите кнопку **Отправить извещение**.



ПОДПИСАН И ОТПРАВЛЕН

Входящий Исправление №10 от 22.10.2022 к счёт-фактуре №13 12 от 23.09.2022

Тип документа: УПД

Функция документа: УПД

Статус УПД: Требуется подтверждение получения

ОТПРАВИТЬ ИЗВЕЩЕНИЕ

ПЕЧАТЬ

ИСТОРИЯ

Рисунок 111. Кнопка "Отправить извещение" на вкладке "Журнал обмена ЮЗДО"

Будет открыто окно выбора сертификата для подписания извещения.

4. Выберите сертификат квалифицированной подписи и нажмите кнопку **ОК**.

Подтверждение будет отправлено контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). В "*Журнале обмена ЮЗДО*" появится запись о создании и (позже) отправке подтверждения получения извещения.

Аннулирование формализованного документа

Отправленный/полученный формализованный документ может быть аннулирован, если было **подтверждено получение извещения** от оператора ЭДО.

Отправленный документ может быть аннулирован в одностороннем порядке до получения извещения от оператора ЭДО. В данном случае подтверждение получения извещения не требуется.

Полученный документ может быть аннулирован продавцом, до подтверждения получения извещения или до подписания Акта/Накладной.

Чтобы отправить запрос на аннулирование:

1. Откройте аннулируемый документ.
2. Нажмите кнопку **Запросить аннулирование** на ленте карточки. Будет предложено указать причину аннулирования документа.

Введите комментарий

ОК

Отмена

Рисунок 112. Указание причины аннулирования документа

3. Укажите причину аннулирования и нажмите кнопку **ОК**. Будет предложено выбрать сертификат для подписания сообщения об аннулировании.
4. Выберите квалифицированный сертификат и нажмите **ОК**.

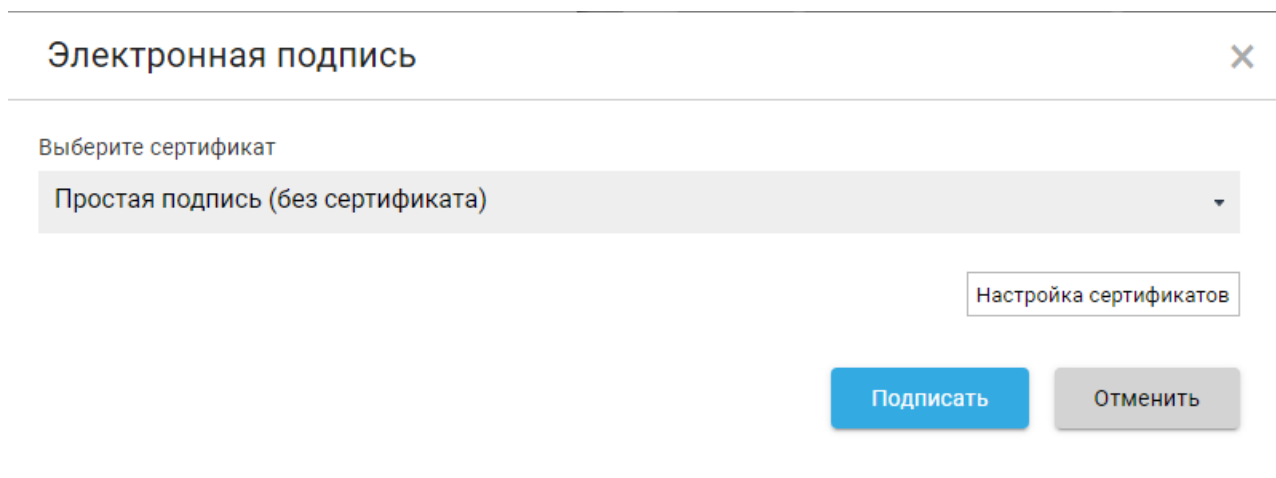
Запрос на аннулирование будет отправлен после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*).

После подтверждения аннулирования или при одностороннем аннулировании статус документа сменится на **Аннулирован**. В случае отказа в аннулировании со стороны контрагента статус вернётся в предыдущее значение, за исключением случая, когда осуществляется аннулирование документа, по которому был отправлен запрос на уточнение — в данном случае статус сменится на **Получен от контрагента** или **Отправлена ответная подпись контрагенту**. Причина отказа будет указана в *Журнале обмена ЮЗДО*.

Подтверждение аннулирования формализованного документа

Если аннулирование инициировано контрагентом, организация может подтвердить или отклонить аннулирование.

1. Откройте документ со статусом обмена в [Журнале обмена ЮЗДО](#): [Получен запрос на аннулирование](#).
2. Нажмите кнопку на ленте карточки для принятия решения по документу:
 - **Подтвердить аннулирование** — для подтверждения аннулирования. При этом нужно будет выбрать сертификат квалифицированной подписи для формирования ответной подписи.



Электронная подпись

Выберите сертификат

Простая подпись (без сертификата)

Настройка сертификатов

Подписать Отменить

Рисунок 113. Выбор сертификата электронной подписи

- **Отказать в аннулировании** — для отказа от аннулирования. При этом нужно будет указать причину отказа.



Введите комментарий

ОК Отмена

Рисунок 114. Указание причины отказа от аннулирования документа

Подтверждение (или отказ) аннулирования будет отправлено контрагенту после обработки процессами модуля (БП *Отправка сообщений ЮЗДО*). При подтверждении аннулирования статус документа сменится на [Аннулирован](#). При отказе в аннулировании статус документа сменится на [Получен от контрагента](#), [Получена подпись от контрагента](#), [Отправлен контрагенту](#), [Получена подпись от](#)

контрагента.

Печатная форма формализованного документа

Элемент разметки "Печатная форма" предназначен для просмотра электронного документа в печатном виде.

Универсальный передаточный документ

Статус: ☐

1 - счет-фактура и передаточный документ (акт)

2 - передаточный документ (акт)

Счет-фактура № 1312 от 23.09.2019 (1)

Исправление № 10 от 22.10.2019 (1a)

Продавец ООО Перспектива

Адрес Санкт-Петербург, Большая пушкарская, 28

ИНН/КПП продавца 9543541759 / 944371400

Грузоотправитель и его адрес ООО Пантеон, Г.Москва, вн.тер. г. Муниципальный Округ Арбат, ул Старый Арбат, дом 32, помещен комната 8

Грузополучатель и его адрес ООО Перспектива, Санкт-Петербург, Большая пушкарская, 28

К платежно-расчетному документу № 2015 от 04.09.2019

Покупатель ООО Перспектива

Адрес Санкт-Петербург, Большая пушкарская, 28

ИНН/КПП покупателя 9664313085 / 966401000

Валюта: наименование, код Российский рубль

Приложение № 1 к письму ФНС России от 21.10.2013 № МДБ-20-3/96@

Г. к постановлению Правительства Российской Федерации от 26 декабря 2011 года № 1234

N п/п	Код товара/работ, услуг	Наименование товара (описание выполненных работ, оказанных услуг), имущественного права	Единица измерения		Количество (объем)	Цена (тариф) за единицу измерения	Стоимость товаров (работ, услуг), имущественных прав без налога - всего	В том числе сумма акциза	Налоговая ставка	Сумма налога, предъявляемая покупателю	Стоимость товаров (работ, услуг), имущественных прав с налогом - всего	Страна происхождения товара	
			код	условное обозначение (национальное)								Цифровой код	Краткое наименование
A	B	1	2a	2б	3	4	5	6	7	8	9	10a	10б
1		Стулья офисные модуль "Перспектива 17"		шт	8	5000.00	40000.00	без акциза	без НДС	без НДС	43000.00		
2		Стулья офисные модуль "Перспектива 15"		шт	5	8000.00	45000.00	без акциза	без НДС	без НДС	45000.00		
Всего к оплате:							85000.00	x	x	без НДС	88000.00		
Руководитель организации или иное уполномоченное лицо			электронная подпись (подпись)		Мяснишкин Н.Р. (ф.и.о.)		Главный бухгалтер или иное уполномоченное лицо		электронная подпись (подпись)		Нечаев С.Д. (ф.и.о.)		
Индивидуальный предприниматель			(подпись)		(ф.и.о.)		(реквизиты свидетельства о государственной регистрации индивидуального предпринимателя)		(подпись)		(ф.и.о.)		
Основание передачи (сдачи) / получения (приемки) Отсутствует [8]													
(договор; доверенность и др.)													

Рисунок 115. Пример печатной формы

Для печати формы нажмите кнопку **Печать** на ленте карточки или используйте одноименную команду из контекстного меню *Печатной формы*.

Групповые операции с неформализованными документами

В данный момент модель не поддерживает групповые операции с неформализованными документами при работе в Web-клиенте.

Журнал обмена ЮЗДО

Журнал обмена ЮЗДО содержит историю электронного обмена по данному документу с оператором ЭДО.

Для просмотра журнала откройте страницу *Журнал обмена ЮЗДО* в карточке электронного документа. Пример журнала приведен на рисунке.

^ ЖУРНАЛ ОБМЕНА						
Статус: Отправлен отказ от подписи контрагенту						
Тип события	Дата	Файлы	Подписант	Проверка подписи	Описание	Ком...
От контрагента	13.10.2022 17:41	Корсчёт-фактура.xml	Кузнецов З. К.	✓	Добавлен входящий файл.	
От контрагента	13.10.2022 17:41	Корректировочный счёт-фактура №5 от 13.10.2022			Создан входящий документ <Корректировочный счёт-фактура №5 от 13.10.2022> от <ООО Пантеон>.	
От оператора обмена	13.10.2022 17:41	Отправлен подтверждающий счёт-фактура от 13.10.2022			Получено подтверждение оператора об отправке.	
Из Docsvision	03.11.2022 14:28	Корректировочный счёт-фактура №5 от 13.10.2022			Отказ от ответной подписи был отклонён	
Из Docsvision	03.11.2022 14:33	Корректировочный счёт-фактура №5 от 13.10.2022			Отказ от ответной подписи успешно отправлен контрагенту.	

Рисунок 116. Журнал обмена ЮЗДО в карточке входящего УЖД

Журнал содержит следующие элементы:

- Колонки:
 - *Тип события* — указывает на источник/тип события. Возможные типы событий: Из Docsvision, От оператора обмена, От контрагента, Ответ, Ошибка, Информация.
 - *Дата* — дата и время выполнения события.
 - *Файлы* — названия отправленных/полученных файлов.
 - *Подписант* — владелец квалифицированного сертификата, которым подписан отправленный файл.
 - *Проверка в подписи* — результат проверки электронной подписи.
 - *Описание* — описание события.
 - *Комментарий* — комментарий, поступивший или сформированный для

контрагента (например, при отказе в подписании).

Работа с приглашениями к обмену

Приглашения к обмену используются, чтобы предложить контрагенту перейти на электронный документооборот.

Контрагент должен быть зарегистрирован в *Справочнике контрагентов*.

Создание исходящего приглашения

Чтобы создать исходящее приглашение:

1. Создайте карточку *Исходящее приглашение ЭДО* из главного меню Web-клиента.

Рисунок 117. Пустая карточка исходящего приглашения

2. В карточке выберите контрагента в поле *Контрагент*.
3. Добавьте *Комментарий* и, при необходимости, файл, например, доп. соглашение:
 - Перетащите файл из файловой системы в область *Файлы* карточки.
 - Нажмите в области *Файлы* левой кнопкой мыши и выберите файл из файловой системы.
 - Нажмите на элемент управления , выберите файл из файловой системы.

Если в карточку не добавлен файл, приглашение отправляется без подписи с выбором подразделения/организации и ящика. Если файл добавлен, появится возможность **Подписать и отправить** приглашение.



Чтобы подписать документ, поле *Организация подписанта* всегда должно быть заполнено. У выбранной организации в справочнике сотрудников или контрагентов должен быть задан ИНН в корректном для типа подписанта формате.

ИНН необходимо указывать, чтобы чтение реквизитов проходило корректно. Для организаций, ИП или физических лиц следует создавать отдельную запись в справочнике с указанием ИНН.

4. При необходимости выберите другое подразделение и ящик ЮЗДО, через который будет отправлен электронный документ. Нажмите кнопку **ОК**. Будет открыто окно выбора сертификата для подписания документа.

Электронная подпись

Выберите вариант подписания

Подписать без метки

Выберите сертификат

Petrov A.F.; Panteon Neolit Certification 2; 03.05.2023

Настройки способов подписи

Просмотр доверенности

Подписать

Отменить

Рисунок 118. Диалог выбора сертификата для подписания

Состояние приглашения изменится на **Отправлено**.

Приглашение будет отправлено контрагенту. За состоянием приглашения можно следить в карточке.



Отправить приглашение возможно только если оно не было отправлено ранее. Для повторной отправки потребуется создать новую карточку.

Входящие приглашения

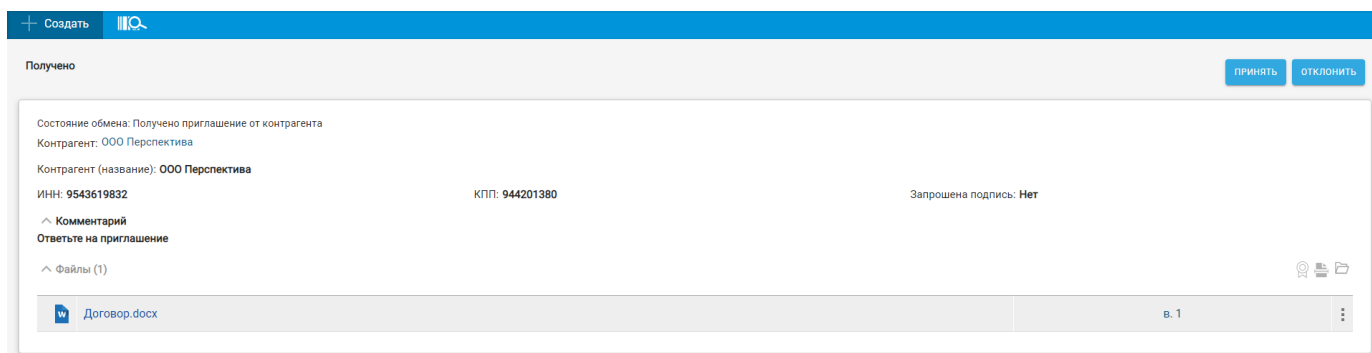


Рисунок 119. Входящее приглашение ЭДО

При получении входящего приглашения:

1. Откройте карточку входящего приглашения к обмену ЭДО.
2. Ознакомьтесь с комментарием и содержимым приложенного файла.
3. Примите решение нажатием кнопок **Принять** или **Отклонить**.
 - Если вы принимаете приглашение, потребуется подтвердить согласие:

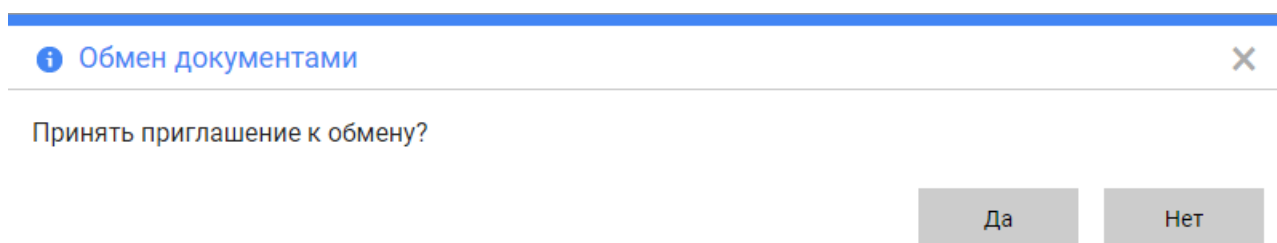


Рисунок 120. Подтверждение принятия приглашения

После подтверждения контрагенту будет отправлен положительный ответ на приглашение к обмену.

- В случае отказа от приглашения потребуется указать причину:

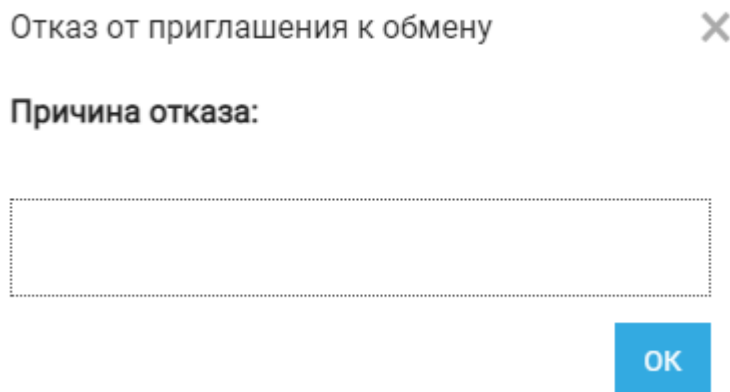


Рисунок 121. Отказ от приглашения к обмену

Контрагенту будет отправлен отказ от приглашения к обмену.

- Если контрагент приложил файл и запросил подпись в приглашении, будет выполнено подписание файла с выбором сертификата.

4. Состояние обмена изменится в зависимости от принятого решения.

Отозвать приглашение

Откройте карточку отправленного исходящего приглашения и нажмите кнопку **Отозвать**.

Руководство разработчика модуля

Разработка компонентов модуля Модуля интеграции с операторами ЭДО

Модуль предоставляет возможность разрабатывать собственные компоненты, представленные на странице *Компоненты* в *Справочнике настроек операторов ЮЗДО*.

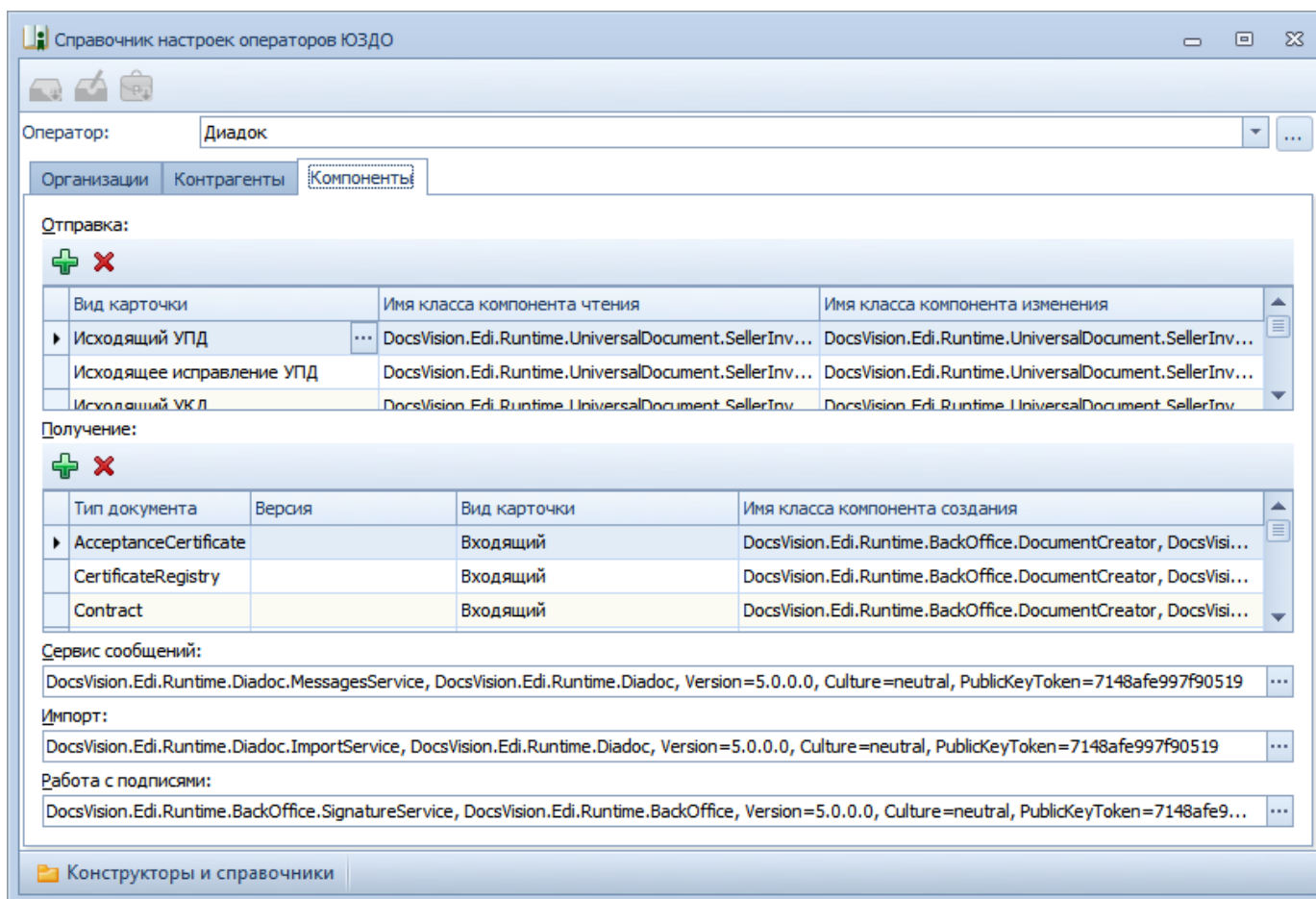


Рисунок 122. Настройки компонентов в Справочнике настроек операторов ЮЗДО

Вынесенная в компоненты функциональность модуля позволяет абстрагировать механизм отправки электронных документов от видов карточек и механизмов взаимодействия с операторами ЭДО.

Разрабатываемые компоненты делятся на две группы:

- Компоненты для работы с карточками Docsvision: компоненты чтения и изменения (данных карточек), а также компонент создания документов.
- Компоненты коннектора к оператору ЭДО: компоненты сервиса сообщений, импорта и работы с электронными подписями.

Разработка компонента чтения данных карточки

Компонент чтения данных предоставляет модулю не зависящий от типа карточки слой получения данных из карточки, необходимых для формирования *Карточки обмена сообщениями*, а также вспомогательные методы, связанные с получением данных из карточек определенного типа.

Компонент должен реализовывать программный интерфейс `undefined/programmer/dv5/Edi/api/IDocumentDataReader/[IDocumentDataReader]`.

Рекомендуется создавать свой компонент на основе абстрактного класса `BaseDataReader` (сборка `DocsVision.Edi.Runtime.UniversalDocument.dll`). Далее приведена часть класса `BaseDataReader` с описанием.

```
public abstract class BaseDataReaders : IDocumentDataReader
{
    public void Initialize(ObjectContext objectContext) ①
    {
        ObjectContext = objectContext;

        UserSession = ObjectContext.GetService<UserSession>();

        documentService = ObjectContext.GetService<IDocumentService>();
    }

    public string GetDigest(Guid cardId) ②
    {
        Document document = GetDocument(cardId);
        if (document == null)
        {
            throw Error.InvalidOperation(Resources.DocumentNotExists, cardId);
        }

        return document.Description;
    }

    public string GetFileDigest(Guid cardId, Guid fileId) ③
    {
        Document document = GetDocument(cardId);
        if (document == null)
        {
            throw Error.InvalidOperation(Resources.DocumentNotExists, cardId);
        }

        DocumentFile documentFile = document.Files.FirstOrDefault(item => item
```

```

.FileVersionRowId == fileId);
    if (documentFile == null)
    {
        throw Error.InvalidOperation(Resources.FileNotExists, fileId, cardId);
    }

    return GetUniversalDocumentDescription(documentFile.FileVersionRowId,
MessageFileType.File) ?? documentFile.FileName;
}

public string GetFileContent(Guid cardId, Guid fileId) ④
{
    Document document = GetDocument(cardId);
    if (document == null)
    {
        throw Error.InvalidOperation(Resources.DocumentNotExists, cardId);
    }

    DocumentFile documentFile = document.Files.FirstOrDefault(item => item
.FileVersionRowId == fileId);
    if (documentFile == null)
    {
        throw Error.InvalidOperation(Resources.FileNotExists, fileId, cardId);
    }

    if (!UserSession.FileManager.FileExists(documentFile.FileVersionRowId))
    {
        return null;
    }

    FileData fileData = UserSession.FileManager.GetFile(documentFile.
FileVersionRowId);
    using (Stream stream = fileData.OpenReadStream())
    {
        using (TextReader textReader = new StreamReader(stream))
        {
            return textReader.ReadToEnd();
        }
    }
}

public string GetPrintFormContent(Guid cardId) ⑤
{
    Document document = GetDocument(cardId);
    if (document == null)
    {
        throw Error.InvalidOperation(Resources.DocumentNotExists, cardId);
    }
}

```

```

        return GetUniversalDocumentPrintContent(document);
    }

    public string GetReplyFileDigest(Guid cardId, Guid fileId, MessageFileType fileType)
    {
        return GetUniversalDocumentDescription(fileId, fileType);
    }

    public string GetReplyFileComment(Guid cardId, Guid fileId, MessageFileType fileType)
    {
        if (!UserSession.FileManager.FileExists(fileId))
        {
            return null;
        }

        try
        {
            FileData fileData = UserSession.FileManager.GetFile(fileId);
            using (Stream stream = fileData.OpenReadStream())
            {
                switch (fileType)
                {
                    case MessageFileType.InvoiceCorrectionRequest:
                    case MessageFileType.SignatureRejection:
                        return InvoiceCorrectionNotice.GetComment(stream);
                }
            }
        }
        catch
        {
        }

        return null;
    }

    public ItemCollection<DocumentRecipientData> GetAllRecipients(Guid cardId) ⑦
    {
        Document document = GetDocument(cardId);
        if (document == null)
        {
            throw Error.InvalidOperation(Resources.DocumentNotExists, cardId);
        }

        ItemCollection<DocumentRecipientData> recipients = new ItemCollection
        <DocumentRecipientData>();
    }

```

```

var contractSection = document.GetSection(CardDocument.Contract.ID);
if (contractSection.Count == 0)
{
    return recipients;
}

BaseCardSectionRow contractRow = (BaseCardSectionRow)contractSection[0];
Guid companyId = contractRow.GetGuid(CardDocument.Contract.PartnerCompany);
if (companyId == Guid.Empty)
{
    return recipients;
}

PartnersCompany company = ObjectContext.GetObject<PartnersCompany>(companyId);
if (company != null)
{
    recipients.Add(new DocumentRecipientData(ObjectContext.GetObjectRef(company
).Id, company.Name));
}

return recipients;
}

public ItemCollection<DocumentSignature> GetActualSignatures(Guid cardId) ⑧
{
    return null;
}

public ItemCollection<DocumentFileData> GetAllFiles(Guid cardId) ⑨
{
    Document document = GetDocument(cardId);
    if (document == null)
    {
        throw Error.InvalidOperation(Resources.DocumentNotExists, cardId);
    }

    return new ItemCollection<DocumentFileData>(document.Files.Where(CheckFile)
        .Select(item => new DocumentFileData(item.FileVersionRowId, item.FileName,
item.FileType == DocumentFileType.Main)));
}

public ItemCollection<DocumentFileData> GetLastSignedFiles(Guid cardId) ⑩
{
    return GetSignedFiles(cardId, Guid.Empty);
}

public ItemCollection<DocumentFileData> GetSignedFiles(Guid cardId, Guid
signatureGroupId) ⑪

```

```

{
    Document document = GetDocument(cardId);
    if (document == null)
    {
        throw Error.InvalidOperation(Resources.DocumentNotExists, cardId);
    }

    if (document.MainInfo.SignatureList == null || !document.MainInfo.SignatureList
        .Signatures.Any())
    {
        return null;
    }

    BaseCardSignature lastSignature = signatureGroupId != Guid.Empty
        ? document.MainInfo.SignatureList.Signatures.FirstOrDefault(item =>
            ObjectContext.GetObjectRef(item).Id == signatureGroupId
            : document.MainInfo.SignatureList.Signatures.OrderByDescending(item => item
                .TimeStamp).FirstOrDefault());

    if (lastSignature == null)
    {
        return null;
    }

    ItemCollection<DocumentFileData> result = new ItemCollection<DocumentFileData>();

    foreach (DocumentFile documentFile in document.Files.Where(CheckFile))
    {
        if (UserSession.CardManager.GetCardState(documentFile.FileId) != ObjectState
            .Existing)
        {
            continue;
        }

        VersionedFileCard verFileCard = (VersionedFileCard)UserSession.CardManager
            .GetCard(documentFile.FileId);

        BaseCardSignaturePart filePart = lastSignature.Parts
            .FirstOrDefault(item => item.FileVersion == verFileCard.CurrentVersion
                .VersionId &&
                ((documentFile.FileType ==
                    DocumentFileType.Main &&
                    documentService
                    .IsDocumentSignaturePartMainFile(item)) ||
                    (documentFile.FileType ==
                    DocumentFileType.Additional &&
                    documentService
                    .IsDocumentSignaturePartAttachments(item))));
    }
}

```



```

        if (filePart == null)
        {
            continue;
        }

        result.Add(new DocumentFileData(documentFile.FileVersionRowId, documentFile
.FileName,
            documentFile.FileType == DocumentFileType.Main, filePart.Signature));
    }

    return result;
}

public MessageData PrepareMessageData(Guid cardId, string documentType, string
documentVersion) ⑬
{
    Document document = GetDocument(cardId);

    if (document.MainInfo.SignatureList == null || !document.MainInfo.SignatureList
.Signatures.Any())
    {
        throw Error.InvalidOperation(Resources.DocumentSignaturesNotExists, cardId);
    }

    MessageData messageData = new MessageData(cardId, documentType, documentVersion,
document.MainInfo.Name);

    DateTime? regDate = document.MainInfo[CardDocument.MainInfo.RegDate] as
DateTime?;
    messageData.DocumentDate = regDate ?? document.CreateDate;
    messageData.DocumentComment = document.MainInfo[CardDocument.MainInfo.Content] as
string;

    Guid numberId = document.MainInfo.GetGuid(CardDocument.MainInfo.RegNumber);
    if (numberId != Guid.Empty)
    {
        BaseCardNumber number = document.Numbers.FirstOrDefault(item =>
ObjectContext.GetObjectRef(item).Id == numberId);
        if (number != null)
        {
            messageData.DocumentNumber = number.Number;
        }
    }

    return messageData;
}

public MessageFile PrepareFileData(Guid cardId, Guid fileId, Guid signatureId, bool

```

```

isMain, string tempFolder) ⑭
{
    return null;
}

// Реализация метода проверки подписи
public SignatureValidation ValidateSignature(Guid cardId, Guid fileId, Guid
signatureId) ⑮
{
    if (cardId == Guid.Empty)
    {
        throw Error.ArgumentNull("cardId");
    }

    if (fileId == Guid.Empty)
    {
        throw Error.ArgumentNull("fileId");
    }

    if (signatureId == Guid.Empty)
    {
        throw Error.ArgumentNull("signatureId");
    }

    if (!UserSession.FileManager.FileExists(fileId))
    {
        return null;
    }

    byte[] signatureData = null;
    try
    {
        signatureData = UserSession.AccessManager.GetCryptObject(signatureId);
    }
    catch
    {
    }
    if (signatureData == null)
    {
        return null;
    }

    string signerName = null;
    X509Certificate2 certificate = null;
    bool isValid = false;
    bool isCertificateValid = false;
    string validationError = null;

```

```

try
{
    FileData fileData = UserSession.FileManager.GetFiles(fileId);
    using (Stream stream = fileData.OpenReadStream())
    {
        using (BinaryReader reader = new BinaryReader(stream))
        {
            ContentInfo contentInfo = new ContentInfo(reader.ReadBytes((int)
)reader.BaseStream.Length));
            SignedCms signedCms = new SignedCms(contentInfo, true);
            signedCms.Decode(signatureData);

            if (signedCms.SignerInfos.Count == 0 || signedCms.Certificates.Count
== 0)
            {
                return null;
            }

            certificate = signedCms.Certificates[0];
            signerName = CertificateHelper.GetCertificateSignerName(certificate);

            signedCms.CheckSignature(false);
            isValid = true;

            signedCms.CheckSignature(new X509Certificate2Collection(certificate),
true);

            isCertificateValid = true;
        }
    }
}
catch (Exception ex)
{
    validationError = ex.Message;
}

return new SignatureValidation(isValid, isCertificateValid, certificate,
signerName, validationError);
}

public string GetInvoiceReplyData(Guid cardId) ⑩
{
    Document document = GetDocument(cardId);
    if (document == null)
    {
        throw Error.InvalidOperation(Resources.DocumentNotExists, cardId);
    }

    return GetInvoiceReplyData(document);
}

```

```

    }

    protected virtual bool CheckFile(DocumentFile documentFile) ⑰
    {
        return false;
    }

    protected virtual string GetUniversalDocumentDescription(Guid fileId, MessageFileType
fileType) ⑱
    {
        return null;
    }

    protected virtual string GetUniversalDocumentPrintContent(Document document) ⑲
    {
        return null;
    }

    protected virtual string GetInvoiceReplyData(Document document) ⑳
    {
        return null;
    }
}

```

- ① Инициализируем компонент чтения. При инициализации получаем сервисы API Docsvision, которые потребуются в дальнейшем.
- ② Реализация метода получения дайджеста карточки.
- ③ Реализация метода получения дайджест файла карточки.
- ④ Реализация метода получения содержимого файла карточки.
Для неформализованного документа реализация функциональности не требуется.
В стандартной реализации для УПД метод возвращает содержимое приложенного файла XML.
- ⑤ Реализация метода получения содержимое карточки для печати.
Для неформализованного документа реализация функциональности не требуется.
В стандартной реализации для УПД используется XSLT-преобразование данных карточки.
К печатной форме требуемого вида, отображаемой в карточке УПД.
- ⑥ Реализация метода получения дайджеста для файла ответа.
- ⑦ Реализация метода, возвращающего получателей электронного документа.

В качестве получателей выбираются все контрагенты из карточки *Документ*.

- ⑧ Реализация метода получения актуальных подписей (действительная подпись и действительный сертификат) карточки.
- ⑨ Реализация метода получения всех приложенных к карточке файлов.
- ⑩ Реализация метода получения последних подписанных файлов.
- ⑪ Реализация метода получения файлов карточки, подписанных указанной подписью.
- ⑫ Только подписанные отдельной подписью файлы.
- ⑬ Реализация метода подготовки данных для создания сообщения электронного обмена.
- ⑭ Реализация метода подготовки файла сообщения электронного обмена.
Если метод возвращает null будет использована стандартная реализация функции получения `MessageFile`.
Если требуется, может быть реализован собственный метод получения `MessageFile`, см. [undefined/programmer/dv5/Edi/api/MessageFile/\[MessageFile — класс\]](http://undefined/programmer/dv5/Edi/api/MessageFile/[MessageFile—класс]) подробнее.
- ⑮ По умолчанию выгружаем файл по ID непосредственно при создании сообщения.
- ⑯ Реализация метода формирования содержимого для ответного сообщения.
Используется при формировании ответа на полученный формализованный документ.
- ⑰ В своей реализации нужно переопределить метод, добавив алгоритм проверки ЭП файла.
- ⑱ В своей реализации можно переопределить метод, добавив алгоритм получения описания для файла `fileId` в зависимости от типа сообщения.
- ⑲ В своей реализации нужно переопределить метод, добавив алгоритм формирования печатной формы для документа `document`.
- ⑳ В своей реализации нужно переопределить метода, добавив алгоритм формирования содержимого для ответного сообщения на полученный формализованный документ.
Пример реализации в классе `DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoiceDataReader` (сборка `DocsVision.Edi.Runtime.UniversalDocument.dll`).

Разработка компонента изменения данных карточки

Компонент изменения данных предоставляет модулю не зависящий от типа карточки слой изменения данных карточки, обеспечивающий возможность внесения изменения при получении ответа от контрагента.

Компонент должен реализовывать программный интерфейс `undefined/programmer/dv5/Edi/api/IDocumentUpdater/[IDocumentUpdater]`.

Рекомендуется создавать свой компонент на основе абстрактного класса `BaseUpdater` (сборка `DocsVision.Edi.Runtime.UniversalDocument.dll`). Далее приведена часть класса `BaseDataReader` с описанием.

```
public abstract class BaseUpdater : IDocumentUpdater
{
    public void Initialize(ObjectContext objectContext) ①
    {
        ObjectContext = objectContext;
        UserSession = objectContext.GetService<UserSession>();

        documentService = objectContext.GetService<IDocumentService>();
        baseCardService = objectContext.GetService<IBaseCardService>();
        staffService = objectContext.GetService<IStaffService>();
    }

    public virtual Guid AddSignatureToDocument(Guid cardId, Guid fileId, MessageFile
messageFile, string partnerName) ②
    {
        Document document = GetDocument(cardId);

        DocumentFile documentFile = document.Files.FirstOrDefault(item => item
.FileVersionRowId == fileId);
        if (documentFile == null)
        {
            return Guid.Empty;
        }

        byte[] signatureData = null;
        if (messageFile.SignatureData != null && messageFile.SignatureData.Length > 0) ③
        {
            signatureData = messageFile.SignatureData;
        }
        else if (!string.IsNullOrEmpty(messageFile.SignatureFilePath))
        {
            signatureData = File.ReadAllBytes(messageFile.SignatureFilePath);
        }
    }
}
```

```

    }

    if (signatureData == null)
    {
        return Guid.Empty;
    }

    SignedCms signedCms = new SignedCms();
    signedCms.Decode(signatureData);

    if (signedCms.SignerInfos.Count == 0 || signedCms.Certificates.Count == 0)
    {
        return Guid.Empty;
    }

    BaseCardSignature cardSignature = document.MainInfo.SignatureList.Signatures
.FirstOrDefault(item => item.SignedFromConfirmation == partnerName); ④
    if (cardSignature == null)
    {
        cardSignature = baseCardService.AddSignature(document.MainInfo.SignatureList,
signedCms.Certificates[0], document.Description, document.SystemInfo.State);
        cardSignature.Signer = staffService.GetCurrentEmployee();
        cardSignature.SignedFromConfirmation = partnerName;
        cardSignature.Certificate = signedCms.Certificates[0].Thumbprint;
        cardSignature.Imported = true;
    }

    ⑤
    BaseCardSignature newSignature = baseCardService.AddSignaturePart(cardSignature,
MainFileSignaturePartType, signatureData,
        string.Format(Resources.MainFilePartnerSignature, documentFile.FileName),
documentFile.FileId, documentFile.FileVersionId, document);

    ObjectContext.SaveObject(document);

    return ObjectContext.GetObjectRef(newSignature).Id;
}

public virtual Guid CreateNewSignatures(Guid cardId, X509Certificate2 certificate,
bool signAdditionalFiles) ⑥
{
    Document document = GetDocument(cardId);

    if (document.MainInfo.SignatureList == null) ⑦
    {
        SignatureList signatureList = documentService.CreateSignatureList();
        ObjectContext.SaveObject(signatureList);
        document.MainInfo.SignatureList = signatureList;
    }
}

```

```

    }

    foreach (DocumentFile documentFile in document.Files)
    {
        UpdateInvoiceFile(document, documentFile, certificate);
        string validateResult = ValidateInvoiceFile(document, documentFile); ⑧
        if (!string.IsNullOrEmpty(validateResult))
        {
            throw new IncompleteSignatureDataException(string.Format(Resources
                .UnableToValidateInvoice, documentFile.FileName, validateResult));
        }
    }

    DocumentSetting documentSetting = documentService.GetKindSettings(document
        .SystemInfo.CardKind);

    BaseCardSignature baseCardSignature = documentService.AddSignature(document,
        certificate, false, documentSetting.DocumentSignature.Fields); ⑨

    ObjectContext.SaveObject(document);

    return ObjectContext.GetObjectRef(baseCardSignature).Id;
}

public virtual string CheckCertificate(Guid cardId, X509Certificate2 certificate) ⑩
{
    return CheckDocumentSignerForCertificate(GetDocument(cardId), certificate);
}

public virtual bool ChangeDocumentState(Guid cardId, Guid newStateId) ⑪
{
    if (cardId == Guid.Empty)
    {
        throw Error.ArgumentNull("cardId");
    }

    Document document = GetDocument(cardId);

    IStateService stateService = ObjectContext.GetService<IStateService>();
    StatesStateMachineBranch branch = stateService.FindLineBranchesByStartState
        (document.SystemInfo.State).FirstOrDefault(item => item.EndState.BuiltInState ==
        newStateId);
    if (branch == null)
    {
        return false;
    }

    if (!stateService.IsOperationAllowedFull(branch.Operation, document))

```



```

    {
        return false;
    }

    stateService.ChangeState(document, branch, true, out string processErrors);

    return true;
}

public virtual Guid GetNewStateId(MessageFileType messageFileType) ⑫
{
    switch (messageFileType)
    {
        case MessageFileType.InvoiceReply:
        case MessageFileType.ReplySignature:
            return SignedStateId;

        case MessageFileType.SignatureRejection:
            return RejectedStateId;

        case MessageFileType.InvoiceCorrectionRequest:
            return CorrectionRequiredStateId;

        case MessageFileType.RevocationRequest:
            return RevocationRequestStateId;

        case MessageFileType.RevocationReply:
            return RevocationReplyStateId;

        case MessageFileType.RevocationRejection:
            return RevocationRejectionStateId;
    }

    return Guid.Empty;
}

public void UpdateDocumentDataFromFile(Guid cardId, Guid fileId) ⑬
{
    if (cardId == Guid.Empty)
    {
        throw Error.ArgumentNull("cardId");
    }

    if (fileId == Guid.Empty)
    {
        throw Error.ArgumentNull("fileId");
    }
}

```

```

        Document document = GetDocument(cardId);

        DocumentFile documentFile = document.Files.FirstOrDefault(item => item
.FileVersionRowId == fileId);
        if (documentFile == null)
        {
            throw Error.InvalidOperation(Resources.FileNotExists, fileId, cardId);
        }

        UpdateDocumentDataFromFile(document, documentFile);
    }

    protected virtual void UpdateInvoiceFile(Document document, DocumentFile
documentFile, X509Certificate2 certificate) ⑭
    {
    }

    protected virtual string ValidateInvoiceFile(Document document, DocumentFile
documentFile) ⑮
    {
        return null;
    }

    protected virtual string CheckDocumentSignerForCertificate(Document document,
X509Certificate2 certificate) ⑯
    {
        CardData documentData = UserSession.CardManager.GetCardData(ObjectContext
.GetObjectRef(document).Id);
        RowData signerRow = documentData.Sections[documentData.Type.AllSections[CardDefs
.UniversalDocumentSignature.Alias].Id].FirstRow;
        Guid employeeId = signerRow.GetGuid(CardDefs.UniversalDocumentSignature.Signer
).GetValueOrDefault();
        StaffEmployee employee = (employeeId != Guid.Empty) ? ObjectContext.GetObject
<StaffEmployee>(employeeId) : null;

        if (employee == null)
        {
            return Resources.NoSignerData;
        }

        string certificateName = CertificateHelper.GetCertificateSignerName(certificate);
        if (string.IsNullOrEmpty(certificateName))
        {
            return Resources.NoCertificateSignerData;
        }

        if (string.Compare($"{employee.LastName} {employee.FirstName} {employee
.MiddleName}", certificateName, StringComparison.InvariantCultureIgnoreCase) != 0)
    }

```

```

        {
            return Resources.DifferentSignerData;
        }

        return null;
    }

    ⑰ protected virtual void UpdateDocumentDataFromFile(Document document, DocumentFile
documentFile)
    {
    }
}

```

- ① Инициализируем компонент отправки.
При инициализации получаем сервисы API Docsvision, которые потребуются в дальнейшем.
- ② Реализация метода добавления подписи к указанному файлу карточки.
- ③ Получаем данные подписи из сообщения.
Подпись может быть в сообщении или в файле, указанном в сообщении.
- ④ Получаем карточку подписей.
- ⑤ Добавляем подпись в документ.
- ⑥ Реализация метода подписания карточки.
- ⑦ Если в документе не списка подписей — создаём.
- ⑧ Обновляем информацию в титуле продавца.
- ⑨ Добавляем подпись в документ.
- ⑩ Реализация метода проверка сертификата подписи в карточке.
Для неформализованного документа реализация функциональности не требуется.
В стандартной реализации для УПД выполняется сравнение данных сотрудника, подписавшего УПД с данными сотрудника-владельца указанного сертификата.
- ⑪ Реализация метода изменения состояния карточки.
- ⑫ Реализация метода получения идентификатора состояния карточки.
Для неформализованного документа реализация функциональности не требуется.
В стандартной реализации для УПД метод возвращает идентификатор одного из встроенных состояний карточки УПД из Конструктора состояний.

⑬ Реализация метода обновления данных карточки данными из файла титула продавца.

Для неформализованного документа реализация функциональности не требуется.

В стандартной реализации для УПД метод получает содержимое файла (XML-формата) и устанавливает значения полей карточки.

⑭ В своей реализации нужно переопределить метод, добавив алгоритм обновления титула продавца.

Пример реализации в классе
`DocsVision.Edi.Runtime.UniversalDocument.SellerInvoiceUpdater` (сборка
`DocsVision.Edi.Runtime.UniversalDocument.dll`).

⑮ В своей реализации нужно переопределить метод, добавив алгоритм проверки файла титула продавца.

⑯ В своей реализации нужно переопределить метод, добавив алгоритм проверки подписанта.

⑰ В своей реализации нужно переопределить метод, добавив алгоритм загрузки данных в карточку из приложенного файла титула продавца.

Пример реализации в классе
`DocsVision.Edi.Runtime.UniversalDocument.BuyerInvoiceUpdater` (сборка
`DocsVision.Edi.Runtime.UniversalDocument.dll`).

Разработка компонента создания документов

Компонент создания документов предоставляет модулю функцию создания карточек в системе Docsvision по поступившим от оператора ЭДО электронным документам.

Компонент должен реализовывать программный интерфейс `undefined/programmer/dv5/Edi/api/IDocumentCreator/[IDocumentCreator]`.

Рекомендуется создавать свой компонент на основе класса `DocumentCreator` (сборка `DocsVision.Edi.Runtime.BackOffice.dll`). Далее приведена часть класса `DocumentCreator` с описанием.

```
public class DocumentCreator : IDocumentCreator
{
    public void Initialize(ObjectContext objectContext, Guid defaultKindId) ①
    {
        ObjectContext = objectContext;
    }
}
```

```

        this.defaultKindId = defaultKindId;

        UserSession = ObjectContext.GetService<UserSession>();

        documentService = ObjectContext.GetService<IDocumentService>();
        staffService = ObjectContext.GetService<IStaffService>();
        baseCardService = ObjectContext.GetService<IBaseCardService>();

        documentFilesHelper = new DocumentFilesHelper(UserSession, documentService,
        baseCardService, staffService);
        parentLinksCreator = new ParentLinksCreator(ObjectContext, UserSession);
    }

    public virtual bool CreateDocument(MessageData messageData, Guid partnerId, Guid
    partnerDepartmentId) ②
    {
        bool isSaved = false;
        Document document = null;

        try
        {
            KindsCardKind newCardKind = GetNewDocumentKind(messageData, partnerId); ③
            if (newCardKind == null)
            {
                return false;
            }

            document = documentService.CreateDocument(null, newCardKind); ④

            if (messageData.DocumentDate.HasValue)
            {
                document.MainInfo.DeliveryDate = messageData.DocumentDate.Value;
            }

            document.MainInfo[CardDocument.MainInfo.ExternalNumber] = messageData
            .DocumentNumber;
            document.MainInfo[CardDocument.MainInfo.Content] = messageData
            .DocumentComment;
            document.MainInfo[CardDocument.MainInfo.RegDate] = DateTime.Now;
            document.MainInfo.Name = messageData.DocumentName;

            var senderRows = document.GetSection(CardDocument.SenderPartner.ID);
            BaseCardSectionRow senderRow;
            if (senderRows.Count == 0)
            {
                senderRow = new BaseCardSectionRow();
                senderRows.Add(senderRow);
            }
        }
    }

```

```

    }
    else
    {
        senderRow = (BaseCardSectionRow)senderRows[0];
    }

    PartnersCompany partnersCompany = ObjectContext.GetObject<PartnersCompany>(partnerId);

    senderRow[CardDocument.SenderPartner.SenderOrg] = partnerId;
    senderRow[CardDocument.SenderPartner.SenderDep] = partnerDepartmentId;

    StaffEmployee employee = GetNewDocumentAuthor(messageData, partnerId);
    Guid employeeId = employee == null ? Guid.Empty : ObjectContext.GetObjectRef(
(employee).Id);

    if (employee != null)
    {
        document.MainInfo.Registrar = employee;
    }

    StaffUnit recipientDepartment = null;

⑤
    StaffEmployee recipient = null;
    StaffGroup recipientGroup = null;
    if (!string.IsNullOrEmpty(messageData.Recipient.DepartmentId))
    {
        OperatorsDepartment operatorsDepartment = ObjectContext.FindObject<OperatorsDepartment>(
            new QueryObject(RefOperators.Departments.DepartmentId, messageData
.Recipient.DepartmentId));
        if (operatorsDepartment != null)
        {
            recipientDepartment = operatorsDepartment.Department;
            recipient = operatorsDepartment.Recipient;
            recipientGroup = operatorsDepartment.RecipientGroup;
        }
    }

    if (recipientDepartment == null)
    {
        OperatorsBox organizationBox = ObjectContext.FindObject<OperatorsBox>(
            new QueryObject(RefOperators.Boxes.BoxId, messageData.Recipient.
BoxId));
        if (organizationBox != null)
        {
            recipientDepartment = organizationBox.Unit.Unit;

```

```

        recipient = organizationBox.Unit.Recipient;
        recipientGroup = organizationBox.Unit.RecipientGroup;
    }
}

if (recipientDepartment != null)
{
    document.MainInfo[CardDocument.MainInfo.ResponsDepartment] =
ObjectContext.GetObjectRef(recipientDepartment).Id;
}

if (recipientGroup != null)
{
    foreach (StaffGroupItem groupItem in recipientGroup.GroupItems)
    {
        BaseCardSectionRow recRow = new BaseCardSectionRow();
        recRow[CardDocument.ReceiversStaff.ReceiverStaff] = groupItem
.EmployeeId;
        document.GetSection(CardDocument.ReceiversStaff.ID).Add(recRow);
    }
}
else if (recipient != null)
{
    BaseCardSectionRow recRow = new BaseCardSectionRow();
    recRow[CardDocument.ReceiversStaff.ReceiverStaff] = ObjectContext
.GetObjectRef(recipient).Id;
    document.GetSection(CardDocument.ReceiversStaff.ID).Add(recRow);
}

if (!string.IsNullOrEmpty(messageData.DocumentType))
{
    SetDocumentType(messageData, document);
}

ObjectContext.SaveObject(document);
isSaved = true;

foreach (MessageFile messageFile in messageData.DocumentFiles) ⑥
{
    AddFileToDocument(document, messageFile, employeeId, partnersCompany.
Name);
}

SetAdditionalAttributes(messageData, document);

ObjectContext.SaveObject(document);

```

```

        parentLinksCreator.CreateParentLinks(document, ⑦
            messageData.DocumentFiles.Select(item => item.OperatorParentEntityId
        ).Where(item => !string.IsNullOrEmpty(item)).Distinct().ToList());

        Guid cardId = ObjectContext.GetObjectRef(document).Id;

        document.Description = baseCardService.GenerateDigest(document, UserSession
        .CardManager.GetCardData(cardId), null);

        ObjectContext.SaveObject(document);

        messageData.CardId = cardId;

        return true;
    }
    catch (Exception)
    {
        SafeRollback();
        try
        {
            if (isSaved)
            {
                ObjectContext.DeleteObject(document);
                ObjectContext.AcceptChanges();
            }
        }
        catch
        {
        }

        throw;
    }
}

public virtual void SetDocumentType(MessageData messageData, Document document) ⑧
{
    CardSection dataSection = UserSession.CardManager.CardTypes[CardDocument.ID
].AllSections ⑨
    .FirstOrDefault(item => string.Equals(item.Alias, CardDefs
    .UniversalDocumentData.Alias));
    if (dataSection == null)
    {
        return;
    }

    if (!dataSection.Fields.Contains(CardDefs.UniversalDocumentData.DocumentType))

```



```

    {
        return;
    }

    Field docTypeField = dataSection.Fields[CardDefs.UniversalDocumentData
.DocumentType];
    var invoiceRows = document.GetSection(dataSection.Id);
    BaseCardSectionRow invoiceRow;
    if (invoiceRows.Count == 0)
    {
        invoiceRow = new BaseCardSectionRow();
        invoiceRows.Add(invoiceRow);
    }
    else
    {
        invoiceRow = (BaseCardSectionRow)invoiceRows[0];
    }

    invoiceRow[CardDefs.UniversalDocumentData.DocumentType] = docTypeField.EnumValues
.FirstOrDefault(item => string.Equals(messageData.DocumentType, item.Alias,
StringComparison.OrdinalIgnoreCase)).Value;
}

public virtual void AddReplyFilesToDocument(MessageData messageData, Guid partnerId)
⑩
{
    try
    {
        StaffEmployee employee = GetNewDocumentAuthor(messageData, partnerId);
        Guid employeeId = employee == null ? Guid.Empty : ObjectContext.GetObjectRef
(employee).Id;

        PartnersCompany partnersCompany = ObjectContext.GetObject<PartnersCompany
>(partnerId);

        Document document = ObjectContext.GetObject<Document>(messageData.CardId);
        foreach (MessageFile messageFile in messageData.DocumentFiles)
        {
            AddFileToDocument(document, messageFile, employeeId, partnersCompany.
Name);
        }

        ObjectContext.SaveObject(document);
    }
    catch
    {
        SafeRollback();
        throw;
    }
}

```

```

    }
}

public virtual KindsCardKind GetNewDocumentKind(MessageData messageData, Guid
partnerId) ⑪
{
    KindsCardKind cardKind = null;
    if (defaultKindId != Guid.Empty)
    {
        cardKind = ObjectContext.GetObject<KindsCardKind>(defaultKindId);
    }

    return cardKind ?? ObjectContext.GetObject<KindsCardKind>(
IncomingDocumentKindId);
}

public virtual StaffEmployee GetNewDocumentAuthor(MessageData messageData, Guid
partnerId) ⑫
{
    return staffService.GetCurrentEmployee();
}

public virtual void SetAdditionalAttributes(MessageData messageData, Document
document) ⑬
{
}

public virtual void AddFileToDocument(Document document, MessageFile messageFile,
Guid authorId, string partnerName) ⑭
{
    documentFilesHelper.AddSignedFileToDocument(document, messageFile, authorId,
partnerName, true);
}
}

```

- ① Инициализация компонента.
- ② Реализация метода создания карточки входящего документа по полученному от оператора ЭДО сообщению.
- ③ Получаем настройки вида карточек.
- ④ Создаём документ.

- ⑤ Группа из настроек подразделения или организации (указывается в получателях).
- ⑥ Добавление файлов из полученного сообщения в создаваемый документ.
- ⑦ Добавление ссылки на родительскую карточку, если поступило исправление формализованного документа.
- ⑧ Добавляет в карточку документа значение типа поступившего документа.
В собственной реализации метод может записывать типы документов по другой логике.
- ⑨ Секции и поля может не быть, если не грузили схему УПД.
- ⑩ Реализация метода переноса файлов из ответа контрагента в карточку.
- ⑪ Предоставляет вид создаваемого документа.
В собственной реализации метод может вычислять вид документа по другой логике.
- ⑫ Предоставляет автора создаваемого документа.
В собственной реализации метод может вычислять автора документа по другой логике.
- ⑬ Добавляет дополнительные атрибуты в создаваемый документ.
В собственной реализации метод может добавлять в карточку требуемые данные из полученного сообщения.
- ⑭ Добавляет в создаваемый документ файл из полученного сообщения.
В собственной реализации метод может добавлять файл, следуя иной логике.

Разработка компонента загрузки событий в существующей карточке

События отдельных документов Модуля интеграции с операторами ЭДО и коннекторах по умолчанию не синхронизируются, так как исходящие события от оператора ЮЗДО не запрашиваются. Данная возможность может быть реализована с помощью скриптов.

Ниже приведён пример загрузки событий произвольного документа в Windows-клиенте с помощью скриптов. Можно загрузить как уже существующий, так и отсутствующий в системе документ. Скрипт может быть назначен на кнопку, добавлен в БП (с другим вариантом ILogWriter) или использован иным способом.

```
private void LoadDocumentById(Guid boxRowId, string documentId, bool isOutgoing) ①
{
```

```

    try
    {
        OperatorsBox operatorsBox = objectContext.GetObject<OperatorsBox>(boxRowId);
        BoxesService boxesService = new BoxesService(objectContext, false);
        LogWriter logWriter = new LogWriter();
        boxesService.LoadDocumentById(operatorsBox, documentId, isOutgoing,
logWriter);
        MessageBox.Show(logWriter.LogText);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}

internal class LogWriter : ILogWriter
{
    private readonly StringBuilder logBuilder = new StringBuilder();

    public void WriteInformation(string message, params object[] args)
    {
        logBuilder.AppendLine(string.Format(message, args));
    }

    public void WriteError(string message, params object[] args)
    {
        logBuilder.AppendLine(string.Format(message, args));
    }

    public void WriteError(string message, Exception ex, params object[] args)
    {
        logBuilder.AppendLine(string.Format(message, args));
        logBuilder.AppendLine(ex.ToString());
    }

    public string LogText => logBuilder.ToString();
}

```

① Загрузка документа по ID.

Параметры:

- **boxRowId** — ID строки ящика из справочника операторов ЮЗДО.
- **documentId** — ID документа Диадок, то что у них в параметре **letterId** в адресе.
- **isOutgoing** — **true** — загрузить как исходящий, **false** — как входящий.

Следующий фрагмент кода демонстрирует загрузку недостающих событий в уже существующий документ при помощи скрипта Windows-клиента.

```
private void RefreshDocumentEvents_ItemClick(System.Object sender, DevExpress.XtraBars
.ItemClickEventArgs e)
{
    try
    {
        var searchQuery = CardControl.Session.CreateSearchQuery();
        var messageTypeQuery = searchQuery.AttributiveSearch.CardTypeQueries.AddNew
(CardEdiMessage.ID);
        var messageMainInfoQuery = messageTypeQuery.SectionQueries.AddNew(CardEdiMessage
.MainInfo.ID);
        messageMainInfoQuery.ConditionGroup.Operation = ConditionGroupOperation.And;
        messageMainInfoQuery.ConditionGroup.Conditions.AddNew(CardEdiMessage.MainInfo
.CardId, FieldType.UniqueId, ConditionOperation.Equals, CardControl.CardData.Id);

        CardDataCollection foundCards = CardControl.Session.CardManager.FindCards
(searchQuery.GetXml());
        if (foundCards.Count == 0)
        {
            CardControl.ShowMessage("У документа нет связанных карточек сообщений ЭДО.");
            return;
        }

        EdiMessage ediMessage = CardControl.ObjectContext.GetObject<EdiMessage>(foundCards
.Last().Id);

        if (string.IsNullOrEmpty(ediMessage.MainInfo.OperatorMessageId))
        {
            CardControl.ShowMessage("Карточка обмена сообщениями не связана с сообщением
оператора.");
            return;
        }

        LogWriter logWriter = new LogWriter();
        BoxesService boxesService = new BoxesService(CardControl.ObjectContext);
        boxesService.LoadDocumentById(ediMessage.MainInfo.OrganizationBox, ediMessage
.MainInfo.OperatorMessageId, !ediMessage.MainInfo.IncomingMessage, logWriter);
        CardControl.ShowMessage(logWriter.LogText);
    }
    catch (Exception ex)
    {
        CardControl.ShowMessage(ex.Message, "Документ", ex.ToString(), MessageType.Error,
MessageButtons.Ok);
    }
}
```

```

}

internal class LogWriter : ILogWriter
{
    private readonly StringBuilder logBuilder = new StringBuilder();

    public void WriteInformation(string message, params object[] args)
    {
        logBuilder.AppendLine(string.Format(message, args));
    }

    public void WriteError(string message, params object[] args)
    {
        logBuilder.AppendLine(string.Format(message, args));
    }

    public void WriteError(string message, Exception ex, params object[] args)
    {
        logBuilder.AppendLine(string.Format(message, args));
        logBuilder.AppendLine(ex.ToString());
    }

    public string LogText
    {
        get{return logBuilder.ToString();}
    }
}

```

Для работы скрипта потребуется указать дополнительные пространства имен:

```

using DocsVision.Edi.Runtime;
using DocsVision.Edi.CardLib.CardDefs;
using DocsVision.Platform.ObjectManager.SearchModel;
using DocsVision.Platform.ObjectManager.Metadata;
using DocsVision.Edi.ObjectModel;
using DocsVision.Edi.Interfaces;
using System.Linq;
using System.Text;

```

Также для компиляции следует подключить дополнительные сборки из папки C:\Program Files (x86)\Docsvision\Edi\5.5\:

```

DocsVision.Edi.ObjectModel.dll
DocsVision.Edi.Interfaces.dll

```

При использовании скрипта для новых видов документов, если в нём не было загрузки библиотек ЭДО и обращений к **EdiScriptHelper**, в инициализацию необходимо добавить следующие строки, например, для **CardActivated**:

```
EdiMapperFactory.RegisterFactory(cardControl.ObjectContext);
EdiServiceFactory.RegisterFactory(cardControl.ObjectContext);
```

А также дополнить сборки:

```
using DocsVision.Edi.ObjectModel.Mapping;
using DocsVision.Edi.ObjectModel.Services;
```



При использовании скрипта существует ограничение, связанное с отображением кнопок предыдущего состояния. При смене состояния документа в Диадок, например, подтверждении или отказе от аннулирования, состояние документа в Docsvision не изменится, а в карточке останутся доступны кнопки предыдущего состояния (запроса аннулирования).

Разработка коннектора

Модуль позволяет добавлять возможность обмениваться электронными документами с контрагентами через различных операторов ЭДО.

Чтобы добавить возможность обмениваться через оператора ЭДО, нужно реализовать три компонента:

- Компонент сервиса сообщений,.
- Компонент импорта.
- Компонент сервиса подписей.

Для каждого оператора ЭДО разрабатывается свой комплект компонентов коннектора. undefined/programmer/dv5/Edi/[Компоненты для работы с карточками] могут быть общими для всех операторов ЭДО или для каждого оператора может использоваться собственный набор компонентов.

Далее приведены требования и примеры реализации компонентов коннектора к оператору ЭДО, а также инструкция по подключению разработанного

коннектора к модулю.

Разработка компонента сервиса сообщений

Компонент сервиса сообщений предоставляет модуль не зависящий от оператора ЭДО слой отправки и получения электронных сообщений.

Компонент данного типа непосредственно взаимодействует с оператором ЭДО. При взаимодействии выполняется передача и получение сообщений электронного обмена, в т.ч. содержащих электронные документы (входящие и исходящие).

Данный компонент должен реализовывать программный интерфейс `undefined/programmer/dv5/Edi/api/IMessageService/[IMessageService]`. Методы реализуемого интерфейса вызываются при получении и отправке сообщений оператору ЭДО.

Вариант реализации данного компонента содержится в сборке `DocsVision.Edi.Runtime.Diadoc.dll` (добавляется при установке "Коннектор к Диадок").

Фактическая реализация компонента сервиса сообщений сильно зависит от метода взаимодействия с оператором ЭДО и предоставляемого им API.

Далее приведена часть исходного кода (только публичные методы) примера реализации интерфейса `IMessageService` для работы с оператором *Диадок*.

```
public class MessagesService: IMessageService ①
{
    private DiadocSession DiadocSession ②
    {
        get
        {
            if (diadocSession != null && !diadocSession.IsAlive())
            {
                diadocSession = null;
            }

            if (diadocSession == null)
            {
                if (!settings.ContainsKey(DiadocSettings.ApiUrl))
                {
                    throw Error.InvalidOperation(Resources.NoConnectionData);
                }
            }
        }
    }
}
```



```

        if (!settings.ContainsKey(DiadocSettings.Login))
        {
            throw Error.InvalidOperation(Resources.NoConnectionData);
        }

        diadocSession = new DiadocSession(
            settings[DiadocSettings.ApiUrl],
            settings[DiadocSettings.Login],
            settings[DiadocSettings.Password],
            settings.ContainsKey(DiadocSettings.ProxyUrl) ? settings
[DiadocSettings.ProxyUrl] : null,
            settings.ContainsKey(DiadocSettings.ProxyLogin) ? settings
[DiadocSettings.ProxyLogin] : null,
            settings.ContainsKey(DiadocSettings.ProxyPassword) ? settings
[DiadocSettings.ProxyPassword] : null);
    }

    return diadocSession;
}
}

③ public void Initialize(Dictionary<string, string> settings) ④
{
    this.settings = settings;
}

public void SendMessage(MessageData messageData) ⑤
{
    DiadocSession.SendMessage(messageData);
}

public void SendSignatureReply(MessageData messageData) ⑥
{
    DiadocSession.SendSignatureReply(messageData);
}

public void SendReceipt(MessageData messageData) ⑦
{
    DiadocSession.SendReceipt(messageData);
}

public void SendRevocation(MessageData messageData) ⑧
{
    DiadocSession.SendRevocation(messageData);
}

public void SendCorrection(MessageData messageData) ⑨

```

```

{
    DiadocSession.SendCorrection(messageData);
}

public int GetNewEventsCount(string boxId, string lastEventId) ⑩
{
    return DiadocSession.GetNewEventsCount(boxId, lastEventId);
}

public string GetLastEventId(string boxId) ⑪
{
    return DiadocSession.GetLastEventId(boxId, null);
}

public bool EventExists(string boxId, string eventId) ⑫
{
    return DiadocSession.EventExists(boxId, eventId);
}

public ItemCollection<MessageData> GetNewEvents(string boxId, string lastEventId,
DateTime? fromDate) ⑬
{
    return DiadocSession.GetNewEvents(boxId, lastEventId, fromDate);
}

public MessageFile GenerateInvoiceReceipt(string boxId, string messageId, string
entityId, X509Certificate2 certificate, string positionName) ⑭
{
    return DiadocSession.GenerateInvoiceReceipt(boxId, messageId, entityId,
certificate, positionName);
}

public MessageFile GenerateRevocationRequest(string boxId, string messageId, string
entityId, ⑮
X509Certificate2 certificate, string positionName, string comment)
{
    return DiadocSession.GenerateRevocationRequest(boxId, messageId, entityId,
certificate, positionName, comment);
}

public MessageFile GenerateInvoiceReply(string boxId, string messageId, string
entityId, X509Certificate2 certificate, string documentType, string replyData) ⑯
{
    return DiadocSession.GenerateInvoiceReply(boxId, messageId, entityId,
certificate, documentType, replyData);
}

public MessageFile GenerateSignatureRejection(string boxId, string messageId, string

```

```

entityId, ⑰
    X509Certificate2 certificate, string positionName, string comment)
{
    return DiadocSession.GenerateSignatureRejection(boxId, messageId, entityId,
certificate, positionName, comment);
}

public MessageFile GenerateCorrectionRequest(string boxId, string messageId, string
entityId, ⑱
    X509Certificate2 certificate, string positionName, string comment)
{
    return DiadocSession.GenerateCorrectionRequest(boxId, messageId, entityId,
certificate, positionName, comment);
}
}

```

- ① Компонент сервиса сообщение.
- ② Точка доступа к API Диадок.
- ③ Реализация интерфейса IMessageService.
- ④ Инициализация компонента.
- ⑤ Реализация метода отправки сообщения оператору ЭДО.
- ⑥ Реализация метода отправки ответной подписи.
- ⑦ Реализация метода отправки квитанции.
- ⑧ Реализация метода отправки запроса на аннулирование.
- ⑨ Реализация метода отправки запроса на уточнение.
- ⑩ Реализация метода получения количества новых событий.
- ⑪ Реализация метода получения идентификатора последнего события в ящике организации.
- ⑫ Реализация метода проверки наличия события в ящике.
- ⑬ Реализация метода получения новых событий из ящика.
- ⑭ Реализация метода формирования извещения.
- ⑮ Реализация метода формирования запроса на аннулирование.
- ⑯ Реализация метода формирования ответной подписи для поступившего документа.
- ⑰ Реализация метода формирования сообщения с отказом в подписании.
- ⑱ Реализация метода формирования запроса на уточнение.

Далее приведена часть исходного кода класса `DiadocSession`, в котором показан принцип взаимодействия с оператором ЭДО через его API на примере получения идентификатора последнего сообщения.

```
public class DiadocSession
{
    private readonly DiadocApi DiadocApi; ①
    private readonly string AuthToken;

    public DiadocSession(string apiUrl, string login, string password, string
proxyUrl, string proxyLogin, string proxyPassword)
    {
        WinApiCrypt diadocCrypt = new WinApiCrypt();
        DiadocApi = new DiadocApi(DefaultClientId, apiUrl, diadocCrypt);

        if (!string.IsNullOrEmpty(proxyUrl))
        {
            DiadocApi.DisableSystemProxyUsage();
            DiadocApi.SetProxyUri(proxyUrl);
            if (!string.IsNullOrEmpty(proxyLogin))
                DiadocApi.SetProxyCredentials(proxyLogin, proxyPassword);
        }

        AuthToken = DiadocApi.Authenticate(login, password);
    }

    public string GetLastEventId(string boxId, string lastEventId) ②
    {
        while (true)
        {
            BoxEventList eventList = DiadocApi.GetNewEvents(AuthToken, boxId,
lastEventId);
            if (eventList == null || eventList.TotalCount == 0)
                return lastEventId;

            lastEventId = eventList.Events[eventList.Events.Count - 1].EventId;
            if (eventList.TotalCount == eventList.Events.Count)
                return lastEventId;
        }
    }
}
```

- ① `DiadocApi` — объект API Диадок, проксирующий работу с веб-сервисом Диадок. Описание API Диадок приведено на странице с документацией Диадок API.

② Получаем идентификатор последнего события в ящике организации.

Разработка компонента сервиса ответных сообщений

Далее приведена часть исходного кода (только публичные методы) примера реализации класса `OutgoingMessagesService` для отправки ответа на неформализованный документ.

```
OutgoingMessagesService outgoingMessagesService = new OutgoingMessagesService(
    cardControl.ObjectContext); ①

EdiMessage messageCard = outgoingMessagesService.FindIncomingMessageCards(cardId
).FirstOrDefault(); ②

③

DocumentUpdaterService documentUpdaterService = new DocumentUpdaterService(
    objectContext); ④

Guid newSignatureGroupId = documentUpdaterService.CreateNewSignatures(cardId,
    certificate, false); ⑤

⑥

DataReaderService dataReaderService = new DataReaderService(objectContext); ⑦

ItemCollection<DocumentFileData> signedFiles = dataReaderService.GetSignedFiles(cardId,
    newSignatureGroupId); ⑧

if (outgoingMessagesService.CreateOutgoingSignatureReplies(cardId, signedFiles).Any())
{
} ⑨
else
{
    ⑩
}
```

① Сервис отправки исходящих событий.

② Поиск карточки сообщений, для которой нужен ответ, для документа `cardId`.

③ Данный момент можно пропустить, если уже есть готовая ответная подпись. Будьте внимательны, при пропуске существует риск отправить оригинальную подпись контрагента как ответную.

- ④ Сервис изменения документа.
- ⑤ `X509Certificate2 certificate` на входе — это сертификат, которым делаем новые подписи.
- ⑥ Конец создания новой подписи.
- ⑦ Сервис чтения данных из карточек.
- ⑧ Получаем свежеподписанные файлы.
При наличии готовой ответной подписи, `newSignatureGroupId` — это ID подписи из списка подписей.
- ⑨ Все хорошо, подписи отправили.
Ответные подписи успешно отправлены.
- ⑩ По карточке сообщения уже некому отправлять ответ — отработал другой скрипт, кто-то отправил руками и т.п.
Не существует связанных писем в ящике ЮЗДО, ожидающих ответных действий по ним. Отправка произведена не будет.

Далее приведён пример отправки отказа от подписи, сервисы те же, что в предыдущем примере:

```
outgoingMessagesService.CreateOutgoingRejectionReplies(cardId, commentText, certificate);  
①
```

- ① `commentText` — причина отказа, `X509Certificate2 certificate` — сертификат.

Следующий пример демонстрирует отправку положительного ответа на УПД или УКД:

```
string invoiceReplyData = dataReaderService.GetInvoiceReplyData(cardId); ①  
  
if (outgoingMessagesService.CreateOutgoingInvoiceReplies(cardId, certificate,  
invoiceReplyData).Any()) ②  
{  
    ③  
}  
else  
{  
    ④  
}
```

- ① Получение данных ответа для ответа из карточки.

- ② Генерация файла ответа, подписание его сертификатом `X509Certificate2 certificate`, отправка.
- ③ Отправили успешно.
- ④ Не отправили, то же, что и для неформализованных.

Разработка компонента импорта настроек организации из системы оператора ЭДО

Компонент импорта предоставляет модуль зависящий не от оператора ЭДО слой получения списка подразделений организации, контрагентов и ящиков, зарегистрированных в системе оператора ЭДО. Информация, предоставляемая данным компонентом, требуется для определения связей между контрагентами, зарегистрированными в системе Docsvision, и контрагентами организации в системе оператора ЭДО.

Данный компонент должен реализовывать программный интерфейс `undefined/programmer/dv5/Edi/api/IImportService/[IImportService]`. Методы реализуемого интерфейса вызываются из компонента *Справочника настроек операторов ЮЗДО* при загрузке подразделений и ящиков организации, контрагентов и ящиков контрагентов.

Реализация данного компонента, получающего информацию от оператора Диадок, содержится в сборке `DocsVision.Edi.Runtime.Diadoc.dll` (добавляется при установке *Коннектор к Диадок*).

Фактическая реализация компонента импорта сильно зависит от метода взаимодействия с оператором ЭДО и предоставляемого им API.

Далее приведена часть исходного кода (только публичные методы) примера реализации интерфейса `ImportService` для работы с оператором *Диадок*.

```
public class ImportService : IImportService
{
    private DiadocSession DiadocSession ①
    {
        get
        {
            if (diadocSession != null && !diadocSession.IsAlive())
                diadocSession = null;

            if (diadocSession == null)
            {
```

```

        if (!settings.ContainsKey(DiadocSettings.ApiUrl))
            throw Error.InvalidOperation(Resources.NoConnectionData);
        if (!settings.ContainsKey(DiadocSettings.Login))
            throw Error.InvalidOperation(Resources.NoConnectionData);

        diadocSession = new DiadocSession(
            settings[DiadocSettings.ApiUrl],
            settings[DiadocSettings.Login],
            settings[DiadocSettings.Password],
            settings.ContainsKey(DiadocSettings.ProxyUrl) ? settings
[DiadocSettings.ProxyUrl] : null,
            settings.ContainsKey(DiadocSettings.ProxyLogin) ? settings
[DiadocSettings.ProxyLogin] : null,
            settings.ContainsKey(DiadocSettings.ProxyPassword) ? settings
[DiadocSettings.ProxyPassword] : null);
    }

    return diadocSession;
}
}

```

②

```

public void Initialize(Dictionary<string, string> settings) ③
{
    this.settings = settings; ④
}

public ItemCollection<Unit> ImportUnits() ⑤
{
    return DiadocSession.ImportUnits();
}

public ItemCollection<Partner> ImportPartners() ⑥
{
    return DiadocSession.ImportPartners();
}
}

```

- ① Точка подключения к API Диадок.
- ② Реализация интерфейса **IImportService**.
- ③ Реализация метода инициализации компонента.
- ④ Загрузка настроек.

- ⑤ Реализация метода получения от оператора ЭДО списка организаций.
- ⑥ Реализация метода получения от оператора ЭДО списка контрагентов.

Далее приведена часть исходного кода класса `DiadocSession`, показывающая принцип взаимодействия с оператором ЭДО через его API на примере загрузки данных организации.

```
internal class DiadocSession {
    private readonly DiadocApi diadocApi;
    private readonly string authToken;

    public DiadocSession(string apiUrl, string login, string password) {

        ① WinApiCrypt diadocCrypt = new WinApiCrypt();
        diadocApi = new DiadocApi(DefaultClientId, apiUrl, diadocCrypt);
        authToken = diadocApi.Authenticate(login, password);
    }

        ② public ICollection<Unit> ImportUnits() {
            ICollection<Unit> units = new ICollection<Unit>();

            ③ units.AddRange(diadocApi.GetMyOrganizations(authToken).Organizations.Select(GetUnit));
            return units;
        }

        ④ }
    }
```

- ① Подключаемся и авторизуемся в *Диадок*.
- ② Получаем данные подразделений организации.
- ③ Получаем подразделения организации с помощью метода `GetMyOrganizations` API Диадок метод.
- ④ Часть кода пропущена.

Разработка компонента сервиса подписей

Компонент сервиса подписей предоставляет модулю зависящий от оператора ЭДО слой подписание отправляемых электронных сообщений.

Данный компонент должен реализовывать программный интерфейс

undefined/programmer/dv5/Edi/api/ISignatureService/[ISignatureService]. Методы реализуемого интерфейса вызываются при подписании электронных сообщений, отправляемых оператору ЭДО.

Вариант реализации данного компонента содержится в сборке `DocsVision.Edi.Runtime.BackOffice.dll`.

Далее приведена часть исходного кода стандартной реализации компонента сервиса подписей.

```
public class SignatureService : ISignatureService
{
    public void Initialize(ObjectContext objectContext)
    {
        this.objectContext = objectContext;

        baseCardService = this.objectContext.GetService<IBaseCardService>();
    }

    public Guid CreateCommentSignature(string commentText, X509Certificate2 certificate)
    ①
    {
        using (MemoryStream stream = new MemoryStream(Encoding.UTF8.GetBytes(
commentText)))
        {
            byte[] signature = baseCardService.ComputeSignature(stream, certificate);
            return objectContext.GetService<UserSession>().AccessManager.StoreSignature
(Guid.NewGuid(), signature, certificate.Thumbprint);
        }
    }

    public void CreateFileSignature(MessageFile messageFile, X509Certificate2
certificate) ②
    {
        if (messageFile.FileData != null && messageFile.FileData.Length > 0) ③
        {
            using (MemoryStream stream = new MemoryStream(messageFile.FileData))
            {
                byte[] signature = baseCardService.ComputeSignature(stream, certificate);
                messageFile.SignatureData = signature;
            }
        }
        else if (!string.IsNullOrEmpty(messageFile.FilePath))
        {
            using (Stream fileStream = new FileStream(messageFile.FilePath, FileMode.
```

```

Open))
    {
        byte[] signature = baseCardService.ComputeSignature(fileStream,
certificate);
        messageFile.SignatureData = signature;
    }
}
}
}

```

- ① Реализация метода подписания текстового комментария.
- ② Реализация метода подписания файла сообщения.
- ③ Подписываемые данные могут быть получены непосредственно из файла сообщения (`messageFile.FileData`) или с файловой системы (`messageFile.FilePath`).

Регистрация коннектора к оператору ЭДО

После разработки компонентов коннектора к оператору ЭДО, необходимо добавить в Docsvision нового оператора, который будет использовать данные компоненты.

Чтобы добавить нового оператора ЭДО:

1. Откройте карточку *Справочник настроек операторов ЮЗДО* в программе "Docsvision Explorer" (входит в *Resource Kit*).



Инструкция по работе с утилитой *Docsvision Explorer* приведена в документации *Resource Kit*, разделе с описанием утилиты "DVExplorer".

2. Добавьте в секцию *Операторы* новую строку.
3. В поле *Name*, добавленной строки, введите название оператора, которое будет отображаться при настройке в *Справочнике настроек операторов ЮЗДО*.

Card data

Card sections

Sections:

Name
Операторы
Типы документов
Типы полученных доку...

Buttons: BeginUpdate, UpdateNow, CancelUpdate, EndUpdate, Export..., Clear, Refresh

Section rows

Selected section rows:

{FACAB9E7-8527-47A2-825A-588C51F6E68A}
{A224094C-7A22-40BD-90AA-E0473C8A9D20}

Buttons: Enter section, Leave section, Add new, Copy, Delete, Security..., Refresh, Find by Id

Row data

Selected row data:

Field	Type	Value
ImportClassName	FIELD_UNISTRING	<null>
Name	FIELD_UNISTRING	Новый оператор обмена
ServiceClassName	FIELD_UNISTRING	<null>
SignaturesClass...	FIELD_UNISTRING	<null>

Buttons: BeginUpdate, UpdateNow, CancelUpdate, EndUpdate

☐ Show system fields

Рисунок 123. Секция "Операторы" карточки "Справочника настроек операторов ЮЗДО", в Docsvision Explorer

Следующие шаги необходимо выполнить, чтобы добавить настраиваемые из *Справочника настроек операторов ЮЗДО* параметры нового оператора ЭДО. В частности, можно добавить настраиваемые параметры соединения с оператором ЭДО.

4. Откройте строку секции добавленного оператора ЭДО.
5. Добавьте в секцию *Настройки* новую строку.
6. В поле *Name*, добавленной строки, введите название настраиваемого параметра. Значение параметра можно будет установить в пользовательском интерфейсе *Справочника настроек операторов ЮЗДО*.

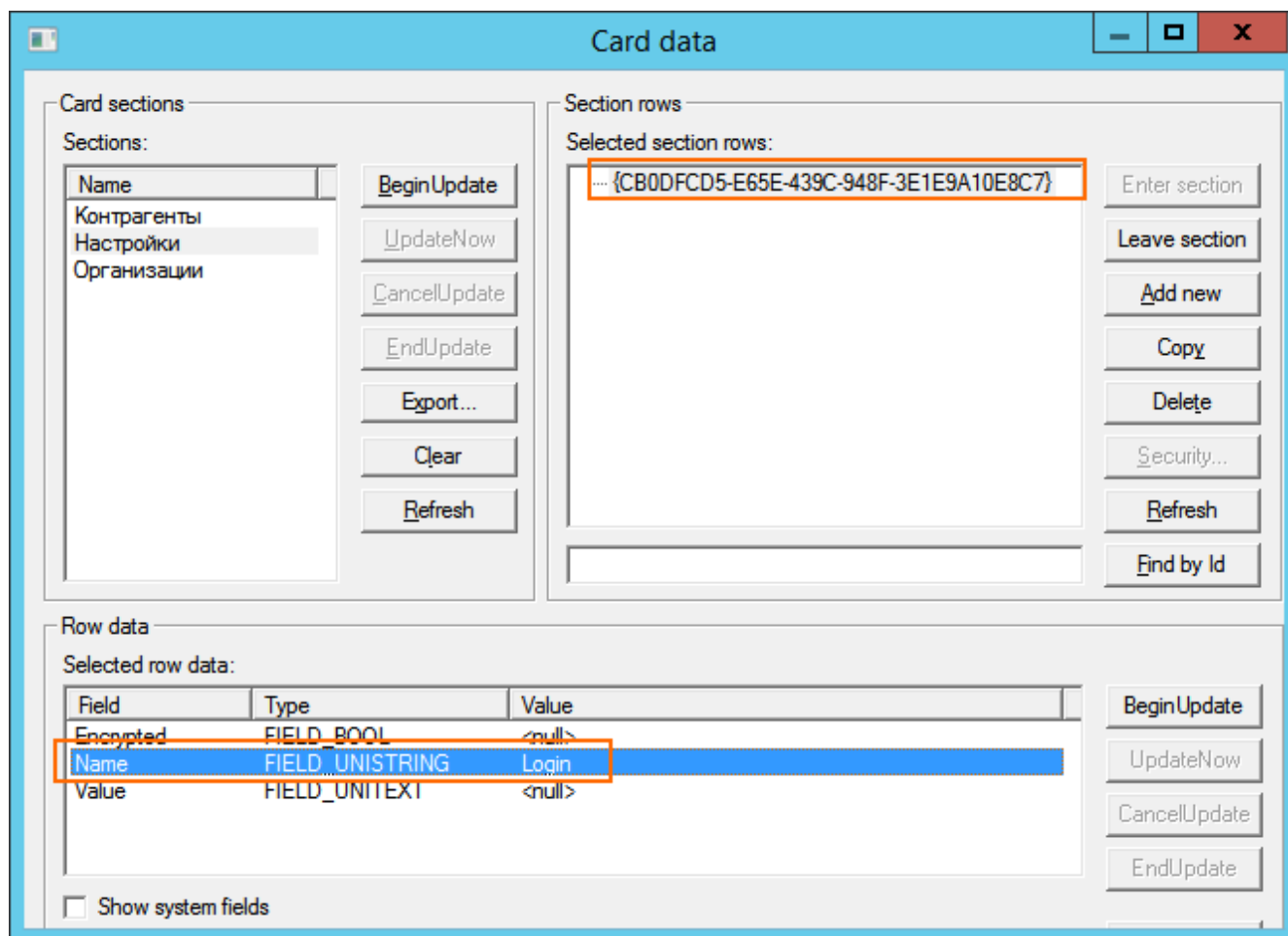


Рисунок 124. Секция "Настройки" карточки "Справочника настроек операторов ЮЗДО", открытая в Docsvision Explorer

- Повторите шаги 5-6, чтобы добавить все необходимые настраиваемые параметры.

Добавьте разработанные компоненты в конфигурацию оператора ЭДО в Справочнике настроек операторов ЮЗДО:

- Зарегистрируйте разработанные компоненты в GAC на сервере Docsvision.
- Откройте Справочнике настроек операторов ЮЗДО и выберите нового оператора ЭДО.

Если оператор недоступен, перезагрузите сервер Docsvision.

- Выберите разработанные компоненты на странице "Компоненты".

Объектная модель модуля

Модуль предоставляет собственную объектную модуль для undefined/programmer/dv5/Edi/[реализации компонентов] и работы с данными

карточек модуля.

В данном разделе перечислены основные члены объектной модули модуль.

Box — класс

Содержит данные ящика электронного обмена.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public class Box
```

Конструкторы

Имя	Описание
<code>Box(String, String)</code>	Инициализирует новый экземпляр класса <code>Box</code> . <i>Параметры:</i> • <code>id</code> — идентификатор ящика. • <code>name</code> — название ящика.

Свойства

Имя	Описание
<code>Id</code>	Задаёт или возвращает идентификатор ящика.
<code>Name</code>	Задаёт или возвращает название ящика.

OperatorsOperator — класс

Содержит данные оператора ЮЗДО.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public class OperatorsOperator : BaseCardSectionRow
```

Свойства

Имя	Описание
Name	Задает или возвращает имя Тип: string
ServiceClassName	Задает или возвращает имя класса сервиса Тип: string
SignaturesClassName	Задает или возвращает имя классов подписей Тип: string
ImportClassName	Задает или возвращает имя класса. Тип: string
LoaderClassName	Задает или возвращает имя класса загрузившего Тип: string
InvitationClassName	Задает или возвращает имя класса приглашения Тип: string
PowerOfAttorneyClassName	Задает или возвращает имя класса МЧД Тип: string
OperatorCertificate	Задает или возвращает сертификат оператора Тип: bool

Имя	Описание
Units	Задает или возвращает подразделения Тип: OperatorsUnit
Partners	Задает или возвращает контрагентов Тип: OperatorsPartner
Settings	Задает или возвращает настройки Тип: OperatorsSetting

Extension — класс

Расширение модуля ЮЗДО для Storage Server.

- **Пространство имен:** DocsVision.Edi.ServerExtension
- **Сборка:** DocsVision.Edi.ObjectModel.dll

Синтаксис

```
public class Extension : StorageServerExtension
```

Методы

Имя	Описание
public void ResetServices(Guid serviceId)	Сбрасывает сервисы
public void ResetAllServices()	Сбрасывает все сервисы
public void InitializeMessageService(Guid serviceId, string serviceName, string settingsXml)	Инициализирует сервис сообщений
public void InitializeImportService(Guid serviceId, string serviceName, string settingsXml)	Инициализирует сервис импорта

Имя	Описание
<code>public void InitializeInvitationService(Guid serviceId, string serviceClassName, string settingsXml)</code>	Инициализирует сервис приглашений
<code>public void InitializePowerOfAttorneyService(Guid serviceId, string serviceClassName, string settingsXml)</code>	Инициализирует сервис машиночитаемой доверенности
<code>public void SendMessage(Guid serviceId, string messageDataXml)</code>	Отправляет сообщение
<code>public void SendSignatureReply(Guid serviceId, string messageDataXml)</code>	Отправляет ответные подписи или отказ подписания
<code>public void SendReceipt(Guid serviceId, string messageDataXml)</code>	Отправляет квитанцию
<code>public void SendRevocation(Guid serviceId, string messageDataXml)</code>	Отправляет отказ
<code>public void SendCorrection(Guid serviceId, string messageDataXml)</code>	Отправляет исправление
<code>public int GetNewEventsCount(Guid serviceId, string boxId, string lastEventId)</code>	Получить счётчик новых событий
<code>public string GetLastEventId(Guid serviceId, string boxId)</code>	Получить идентификатор последнего события ящика
<code>public bool EventExists(Guid serviceId, string boxId, string eventId)</code>	Проверяет, существует ли событие ящика
<code>public string GetNewEvents(Guid serviceId, string boxId, string lastEventId)</code>	Получить новые события
<code>public string GetNewEventsFromDate(Guid serviceId, string boxId, string lastEventId, DateTime fromDate)</code>	Получит новые события с указанной даты

Имя	Описание
<code>public string GenerateInvoiceReceipt(Guid serviceId, string boxId, string messageId, string entityId, string certificateBytes, string positionName)</code>	Генерация формализованной квитанции
<code>public string GenerateRevocationRequest(Guid serviceId, string boxId, string messageId, string entityId, string certificateBytes, string positionName, string comment)</code>	Сгенерировать запрос на отказ
<code>public string GenerateInvoiceReply(Guid serviceId, string boxId, string messageId, string entityId, string certificateBytes, string documentType, string replyData)</code>	Сгенерировать ответный счёт
<code>public string GenerateSignatureRejection(Guid serviceId, string boxId, string messageId, string entityId, string certificateBytes, string positionName, string comment)</code>	Сгенерировать запрос подписи
<code>public string GenerateCorrectionRequest(Guid serviceId, string boxId, string messageId, string entityId, string certificateBytes, string positionName, string comment)</code>	Сгенерировать запрос исправления
<code>public string ImportUnits(Guid serviceId)</code>	Получить список зарегистрированных подразделений
<code>public string ImportPartners(Guid serviceId)</code>	Получить список зарегистрированных контрагентов
<code>public string ImportPartnersFromId(Guid serviceId, string unitId, string lastPartnerId)</code>	Получить список зарегистрированных контрагентов из идентификатора
<code>public string ImportPartner(Guid serviceId, string inn, string kpp)</code>	Получить одного контрагента

Имя	Описание
<code>public void SendInvitation(Guid serviceId, string messageDataXml)</code>	Отправить приглашение
<code>public void AcceptInvitation(Guid serviceId, string messageDataXml)</code>	Принять приглашение
<code>public void RejectInvitation(Guid serviceId, string messageDataXml)</code>	Отклонить приглашение
<code>public void CheckInvitationStatus(Guid serviceId, string messageDataXml)</code>	Проверяет статус приглашения
<code>public string GetNewInvitations(Guid serviceId, string boxId, string lastEventId)</code>	Получает новые приглашения
<code>public string GetNewInvitationsFromDate(Guid serviceId, string boxId, string lastEventId, DateTime fromDate)</code>	Получает новые приглашения с указанной даты
<code>public void RegisterPowerOfAttorney(Guid serviceId, string messageDataXml)</code>	Регистрирует доверенность
<code>public void RevokePowerOfAttorney(Guid serviceId, string messageDataXml)</code>	Отвязывает МЧД от сотрудника в Диадок, в реестре ФНС доверенность не отзывается
<code>public void CheckPowerOfAttorneyStatus(Guid serviceId, string messageDataXml)</code>	Проверяет статус доверенности
<code>public string GetLastError(string errorKey)</code>	Получает последнюю ошибку

BoxesService — класс

BoxesService — основной сервис модуля для выполнения всех стандартных операций.

- **Пространство имен:** `DocsVision.Edi.Runtime`
- **Сборка:** `DocsVision.Edi.Runtime.dll`

```
public class BoxesService
```

Методы



Не рекомендуется самостоятельно вызывать методы обработки очередей сообщений (например, `ProcessAllNewEvents`) без необходимости, т.к. это может привести к конфликтам со штатными БП модуля.

Имя	Описание
<code>public void ResetExtensionServices(OperatorsOperator)</code>	Сбрасывает сервисы серверного расширения.
<code>public bool ImportUnits(OperatorsOperator)</code>	Импортирует подразделения.
<code>public bool ImportPartners(OperatorsOperator)</code>	Импортирует контрагентов.
<code>public bool ImportPartner(OperatorsOperator, string, string)</code>	Импортирует одного контрагента по ИНН и КПП.
<code>public bool ImportPartner(OperatorsOperator, PartnersCompany)</code>	Импортирует одного контрагента по организации контрагента.
<code>public string GetLastEventId(OperatorsBox)</code>	Получает идентификатор последнего события.
<code>public int GetNewEventsCount(OperatorsBox)</code>	Получает количество новых событий ящика.
<code>public bool BoxLastEventExists(OperatorsBox)</code>	Проверяет существование последнего события ящика.
<code>public bool ProcessAllNewEvents(ILogWriter)</code>	Находит и обрабатывает все входящие события.
<code>public int ProcessNewEvents(OperatorsBox, ILogWriter)</code>	Обрабатывает все события в ящике.

Имя	Описание
public bool ProcessAllNewMessagesToSend(ILogWriter)	Находит и обрабатывает все новые исходящие сообщения.
public int ProcessNewMessagesToSend(OperatorsOperator, ILogWriter)	Находит и обрабатывает новые исходящие сообщения для оператора.
public bool ProcessAllNewTransferMessages(ILogWriter)	Находит и обрабатывает все новые сообщения о передаче файлов.
public MessageFile GenerateInvoiceReceipt(OperatorsOperator, string, string, string, X509Certificate2, string)	Генерирует получение счёт-фактуры. <i>Параметры:</i> • ID ящика • ID сообщения • ID сущности • Наименование должности
public MessageFile GenerateRevocationRequest(OperatorsOperator, string, string, string, X509Certificate2, string, string)	Генерирует запрос на аннулирование. <i>Параметры:</i> • ID ящика • ID сообщения • ID сущности • Наименование должности • Комментарий
public MessageFile GenerateInvoiceReply(OperatorsOperator, string, string, string, X509Certificate2, string, string)	Генерирует ответный счёт-фактуру <i>Параметры:</i> • ID ящика • ID сообщения • ID сущности • Тип документа • Ответные данные

Имя	Описание
<pre>public MessageFile GenerateSignatureRejection(OperatorsOperator, string, string, string, X509Certificate2, string, string)</pre>	Генерирует запрос на подписание. <i>Параметры:</i> • ID ящика • ID сообщения • ID сущности • Наименование должности • Комментарий
<pre>public MessageFile GenerateCorrectionRequest(OperatorsOperator, string, string, string, X509Certificate2, string, string)</pre>	Генерирует запрос на исправление. <i>Параметры:</i> • ID ящика • ID сообщения • ID сущности • Наименование должности • Комментарий
<pre>public string LoadIncomingDocuments(OperatorsBox, DateTime, DateTime, string, ILogWriter)</pre>	Загружает входящие документы. <i>Параметры:</i> • Дата с ... • Дата по ... • ID последнего документа
<pre>public string LoadOutgoingDocuments(OperatorsBox, DateTime, DateTime, string, ILogWriter)</pre>	Загружает исходящие документы. <i>Параметры:</i> • Дата с ... • Дата по ... • ID последнего документа
<pre>public bool ProcessAllNewInvitations(ILogWriter)</pre>	Находит и обрабатывает все новые входящие приглашения.
<pre>public bool ProcessAllNewInvitationsToSend(ILogWriter)</pre>	Находит и обрабатывает все новые исходящие приглашения.

Имя	Описание
<code>public DateTime? CheckInvitationsStatus(ILogWriter, DateTime?)</code>	Проверяет статус приглашений с указанной даты.
<code>public bool ProcessAllNewPowersOfAttorneyToSend(ILogWriter logWriter)</code>	Находит и обрабатывает все новые исходящие доверенности.
<code>public DateTime? CheckPowersOfAttorneyStatus(ILogWriter logWriter, DateTime? dateFrom)</code>	Проверяет статус доверенности

DataReaderService — класс

Клас выполняет чтение данных карточек, обычно для чтения получателей и подписанных файлов.

Класс вызывает экземпляры `IDataReader` по привязке к виду или типу карточки. Можно создать свой `IDataReader` с нуля или на основе существующего и прописать его в Справочнике настроек операторов ЮЗДО.

`DataReaderService` можно использовать в своих компонентах или скриптах для получения данных при отправке.

- **Пространство имен:** `DocsVision.Edi.Runtime`
- **Сборка:** `DocsVision.Edi.Runtime.dll`

Синтаксис

```
public class DataReaderService
```

Методы

Имя	Описание
<code>public string GetDigest(Guid)</code>	Получает дайджест карточки по её ID
<code>public string GetFileDigest(Guid, Guid)</code>	Получает дайджест файла документа по ID карточки и ID файла

Имя	Описание
<code>public string GetFileContent(Guid, Guid)</code>	Получает содержимое файла документа по ID карточки и ID файла
<code>public string GetPrintFormContent(Guid)</code>	Получает печатную форму документа из содержимого по ID карточки
<code>public string GetReplyFileDigest(Guid, Guid, MessageFileType)</code>	Получает дайджест ответного файла по ID карточки и ID файла
<code>public string GetReplyFileComment(Guid, Guid, MessageFileType)</code>	Получает комментарий к ответному файлу по ID карточки и ID файла
<code>public ItemCollection<DocumentRecipientData> GetAllRecipients(Guid)</code>	Получает всех получателей для документа по ID карточки
<code>public ItemCollection<DocumentSignature> GetActualSignatures(Guid)</code>	Получает действующие подписи документа по ID карточки
<code>public ItemCollection<DocumentFileData> GetAllFiles(Guid)</code>	Получает все файлы для документа по ID карточки
<code>public ItemCollection<FileValidationResult> ValidateFiles(Guid)</code>	Проверяет структуру файлов, например, действительно ли файл является счёт-фактурой в формате XML по ID карточки
<code>public ItemCollection<DocumentFileData> GetLastSignedFiles(Guid)</code>	Получает последние подписанные файлы для документа по ID карточки
<code>public ItemCollection<DocumentFileData> GetSignedFiles(Guid, Guid)</code>	Получает подписанные файлы для документа по ID карточки и ID группы подписи
<code>public ItemCollection<Guid> GetLinkedCards(Guid)</code>	Получает ссылочные файлы для документа по ID карточки
<code>public MessageData PrepareMessageData(Guid)</code>	Подготавливает данные сообщения для документа по ID карточки
<code>public MessageFile PrepareFileData(Guid, Guid, Guid, bool, string)</code>	Подготавливает данные файла для документа по ID карточки, основного файла, подписи и временной папки.

Имя	Описание
<code>public SignatureValidation ValidateSignature(Guid, Guid, Guid)</code>	Проверяет действительность подписи ID: карточки, файла, подписи
<code>public string GetInvoiceReplyData(Guid)</code>	Получает ответный счёт-фактуру на документ по ID карточки

Department — класс

Содержит данные подразделения организации или контрагента, полученные от оператора ЭДО.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public class Department
```

Конструкторы

Имя	Описание
<code>Department(String, String, String)</code>	Инициализирует новый экземпляр класса <code>Department</code>. <i>Параметры:</i> • <code>id</code> — идентификатор подразделения у оператора ЭДО • <code>name</code> — название подразделения у оператора ЭДО • <code>kpp</code> — КПП подразделения у оператора ЭДО

Свойства

Имя	Описание
Departments	Возвращает коллекцию подчиненных подразделений. Тип: <code>ICollection<Department></code>
Id	Задаёт или возвращает идентификатор подразделения в системе оператора ЭДО. Тип: <code>System.String</code>
Kpp	Задаёт или возвращает КПП подразделения в системе оператора ЭДО. Тип: <code>System.String</code>
Name	Задаёт или возвращает название подразделения в системе оператора ЭДО. Тип: <code>System.String</code>

DocumentFileData — класс

Содержит данные файла карточки *Документ*.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public class DocumentFileData
```

Конструкторы

Имя	Описание
<code>DocumentFileData(Guid, String)</code>	Инициализирует новый экземпляр класса <code>DocumentFileData</code>. <i>Параметры:</i> • <code>id</code> — идентификатор карточки файла. • <code>name</code> — название файла.
<code>DocumentFileData(Guid, String, Boolean)</code>	Инициализирует новый экземпляр класса <code>DocumentFileData</code>. <i>Параметры:</i> • <code>id</code> — идентификатор карточки файла. • <code>name</code> — название файла. • <code>isMain</code> — флаг, определяющий, является ли файл основным.
<code>DocumentFileData(Guid, String, Boolean, Guid)</code>	Инициализирует новый экземпляр класса <code>DocumentFileData</code>. <i>Параметры:</i> • <code>id</code> — идентификатор карточки файла. • <code>name</code> — название файла. • <code>isMain</code> — флаг, определяющий, является ли файл основным. • <code>signatureId</code> — идентификатор подписи файла.

Имя	Описание
<code>DocumentFileData(Guid, String, Boolean, Guid, FileContentType)</code>	Инициализирует новый экземпляр класса <code>DocumentFileData</code>. <i>Параметры:</i> • <code>id</code> — идентификатор карточки файла. • <code>name</code> — название файла. • <code>isMain</code> — флаг, определяющий, является ли файл основным. • <code>signatureId</code> — идентификатор подписи файла. • <code>fileContentType</code> — тип содержимого файла.

Свойства

Имя	Описание
<code>FileContentType</code>	Задаёт или возвращает тип содержимого файла. Тип: <code>undefined/programmer/dv5/Edi/api/FileContentType/[FileContentType]</code>
<code>Id</code>	Возвращает идентификатор карточки версионного файла. Тип: <code>System.Guid</code>
<code>IsMain</code>	Возвращает флаг, определяющий, является ли файл основным: <code>true</code> — основной, <code>false</code> — дополнительный Тип: <code>Boolean</code>
<code>Name</code>	Возвращает название файла. Тип: <code>System.String</code>

Имя	Описание
<code>SignatureId</code>	Задаёт или возвращает идентификатор подписи файла. Тип: <code>System.Guid</code>

DocumentRecipientData — класс

Содержит данные получателя документа.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public class DocumentRecipientData
```

Конструкторы

Имя	Описание
<code>DocumentRecipientData(Guid, String)</code>	Инициализирует новый экземпляр класса <code>DocumentRecipientData</code>. <i>Параметры:</i> • <code>id</code> — идентификатор получателя. • <code>name</code> — название получателя.

Свойства

Имя	Описание
<code>Id</code>	Возвращает идентификатор получателя — идентификатор контрагента в Справочнике контрагентов. Тип: <code>System.Guid</code>

Имя	Описание
Name	Возвращает название получателя — название контрагента. Тип: <code>System.String</code>

DocumentSignature — класс

Содержит данные подписи документа.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public class DocumentSignature
```

Конструкторы

Имя	Описание
<code>DocumentSignature(Guid, DateTime, String, String, X509Certificate2)</code>	Инициализирует новый экземпляр класса <code>DocumentSignature</code> . <i>Параметры:</i> • <code>id</code> — идентификатор подписи в Docsvision. • <code>date</code> — штамп времени подписи. • <code>signerName</code> — отображаемое имя подписавшего. • <code>label</code> — локализованное название метки подписи. • <code>certificate</code> — сертификат подписи.

Свойства

Имя	Описание
<code>Certificate</code>	Возвращает сертификат подписи. Тип: <code>X509Certificate2</code>
<code>Date</code>	Возвращает штамп времени подписи. Тип: <code>DateTime</code>
<code>Id</code>	Возвращает идентификатор подписи в Docsvision. Тип: <code>System.Guid</code>
<code>Label</code>	Возвращает локализованное название метки подписи. Тип: <code>System.String</code>
<code>SignerName</code>	Возвращает отображаемое имя подписавшего. Тип: <code>System.String</code>

DocumentUpdaterService — класс

Класс запись данных в документ, обычно при получении ответов. Вызывает экземпляры `IDocumentUpdater` по привязке к виду или типу карточки.

Можно создать собственный свой `IDocumentUpdater` с нуля или на основе нашего и прописать его в Справочнике настроек операторов ЮЗДО. Тут для своих скриптов или компонентов можно использовать метод `CreateNewSignatures` для создания новых подписей в документе, будет делать новую подпись с учетом вида документа, для примера, для неформализованного это будет просто подпись файла, для формализованного - генерация ответного титула и уже его подпись.

- **Пространство имен:** `DocsVision.Edi.Runtime`
- **Сборка:** `DocsVision.Edi.Runtime.dll`

Синтаксис

```
public class DocumentUpdaterService
```

Методы

Имя	Описание
<code>public Guid AddReplySignatureToDocument(Guid cardId, Guid fileId, MessageFile messageFile, string partnerName)</code>	Добавляет к документу ответную подпись по ID карты и ID файла
<code>public Guid CreateNewSignatures(Guid cardId, X509Certificate2 certificate, bool signAdditionalFiles)</code>	Создаёт новые подписи по ID карты
<code>public string CheckCertificate(Guid cardId, X509Certificate2 certificate)</code>	Проверяет сертификат для карточки по ID карты
<code>public bool ChangeDocumentState(Guid cardId, Guid newStateId)</code>	Изменяет статус документа по ID карты и ID нового статуса
<code>public Guid GetNewStateId(Guid cardId, MessageFileType messageFileType)</code>	Получает ID нового статуса по ID карты
<code>public void UpdateDocumentDataFromFile(Guid cardId, Guid fileId)</code>	Обновляет данные документа из файла по ID карты и ID файла
<code>public bool ChangePowerOfAttorneyState(Guid cardId, Guid powerOfAttorneyCardId, MessageFileType messageFileType)</code>	Изменяет статус МЧД

EdiMessage — класс

Содержит данные *Карточки обмена сообщениями*.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public class EdiMessage : BaseCard
```

Свойства

Имя	Описание
Files	Возвращает файлы сообщения. Тип: ObjectCollection<undefined/programmer/dv5/Edi/api/EdiMessageFile/[EdiMessageFile]>
Log	Возвращает записи их журнал обмена. Тип: ObjectCollection<undefined/programmer/dv5/Edi/api/EdiMessageLogItem/[EdiMessageLogItem]>
MainInfo	Возвращает основную информацию. Тип: undefined/programmer/dv5/Edi/api/EdiMessageMainInfo/[EdiMessageMainInfo]
Processes	Не используется.
SignedComments	Подписанные комментарии. Тип: ObjectCollection<undefined/programmer/dv5/Edi/api/EdiMessageSignedComment/[EdiMessageSignedComment]>
SignedFiles	Подписанные файлы. Тип: ObjectCollection<undefined/programmer/dv5/Edi/api/EdiMessageSignedFile/[EdiMessageSignedFile]>
SystemInfo	Не используется.
TransferData	Данные для переноса. ObjectCollection<undefined/programmer/dv5/Edi/api/EdiMessageTransferData/[EdiMessageTransferData]>

Поля

Имя	Описание
FilesProperty	Определяет свойство <i>Файлы</i> .
LogProperty	Определяет свойство <i>""Журнал обмена""</i> .
MainInfoProperty	Определяет свойство <i>Основная информация</i> .
SignedCommentsProperty	Определяет свойство <i>Подписанные комментарии</i> .
SignedFilesProperty	Определяет свойство <i>Подписанные файлы</i> .
TransferDataProperty	Определяет свойство <i>Файлы для переноса</i> .

EdiMessageFile — класс

Содержит данные файла из *Карточки обмена сообщениями*.

- **Пространство имен:** DocsVision.Edi.ObjectModel
- **Сборка:** DocsVision.Edi.ObjectModel.dll

Синтаксис

```
public class EdiMessageFile : BaseCardSectionRow
```

Свойства

Имя	Описание
ContentType	Задаёт или возвращает тип содержимого файла. Тип: undefined/programmer/dv5/Edi/api/EdiMessageFileContentType/[EdiMessageFileContentType]

Имя	Описание
FileId	Задает или возвращает идентификатор файла. Тип: <code>System.Guid</code>
FileName	Задает или возвращает название файла. Тип: <code>System.String</code>
FileState	Задает или возвращает статус файла. Тип: <code>undefined/programmer/dv5/Edi/api/EdiMessageFileState/[EdiMessageFileState]</code>
FileType	Задает или возвращает тип файла. Тип: <code>undefined/programmer/dv5/Edi/api/EdiMessageFileType/[EdiMessageFileType]</code>
OperatorFileId	Задает или возвращает идентификатор файла у оператора ЭДО. Тип: <code>System.String</code>
OperatorSignatureId	Задает или возвращает идентификатор подписи у оператора ЭДО. Тип: <code>System.String</code>
RejectionComment	Задает или возвращает комментарий к файлу, если пользователь отказывает в подписании. Тип: <code>undefined/programmer/dv5/Edi/api/EdiMessageSignedComment/[EdiMessageSignedComment]</code>

Имя	Описание
ReplyFiles	Возвращает Тип: undefined/programmer/dv5/Edi/api/EdiMessageReplyFile/[EdiMessageReplyFile]
ReplySignatureId	Задает или возвращает идентификатор ответной подписи. Тип: System.Guid
SignatureId	Задает или возвращает идентификатор подписи. Тип: System.Guid

Поля

Имя	Описание
ContentTypeProperty	Определяет свойство <i>Тип содержимого</i> .
FileIdProperty	Определяет свойство <i>Файл</i> .
FileNameProperty	Определяет свойство <i>Имя файла</i> .
FileStateProperty	Определяет свойство <i>Состояние файла</i> .
FileTypeProperty	Определяет свойство <i>Тип файла</i> .
OperatorFileIdProperty	Определяет свойство <i>Идентификатор файла у оператора</i> .
OperatorSignatureIdProperty	Определяет свойство <i>Идентификатор подписи у оператора</i> .
RejectionCommentProperty	Определяет свойство <i>Комментарий при отказе от подписи</i> .
ReplyFilesProperty	Определяет свойство <i>Ответные файлы</i> .
ReplySignatureIdProperty	Определяет свойство <i>Ответная подпись</i> .
SignatureIdProperty	Определяет свойство <i>Подпись</i> .

EdiMessageLogItem — класс

Содержит данные записи в журнале из *Карточки обмена сообщениями*.

- **Пространство имен:** DocsVision.Edi.ObjectModel
- **Сборка:** DocsVision.Edi.ObjectModel.dll

Синтаксис

```
public class EdiMessageLogItem : BaseCardSectionRow
```

Свойства

Имя	Описание
Comment	Задаёт или возвращает текст комментария. Тип: System.String
Date	Задаёт или возвращает дату записи. Тип: System.DateTime
Description	Задаёт или возвращает текст сообщения. Тип: System.String
EventType	Задаёт или возвращает тип события. Тип: undefined/programmer/dv5/Edi/api/EdiMessageEventType/[EdiMessageEventType]
FileName	Задаёт или возвращает название файла, с которым связано событие. Тип: System.String
SignedFile	Задаёт или возвращает подписанный файл. Тип: undefined/programmer/dv5/Edi/api/EdiMessageSignedFile/[EdiMessageSignedFile]

Поля

Имя	Описание
<code>CommentProperty</code>	Определяет свойство <i>Комментарий</i> .
<code>DateProperty</code>	Определяет свойство <i>Дата</i> .
<code>DescriptionProperty</code>	Определяет свойство <i>Описание</i> .
<code>EventTypeProperty</code>	Определяет свойство <i>Тип события</i> .
<code>FileNameProperty</code>	Определяет свойство <i>Имя файла</i> .
<code>SignedFileProperty</code>	Определяет свойство <i>Подписанный файл</i> .

EdiMessageMainInfo — класс

Содержит основную информацию *Карточки обмена сообщениями*.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public class EdiMessageMainInfo : BaseCardSectionRow
```

Свойства

Имя	Описание
<code>CardId</code>	Задаёт или возвращает идентификатор карточки, с которой связано сообщение. Тип: <code>System.Guid</code>
<code>MessageState</code>	Задаёт или возвращает состояние сообщения. Тип: <code>EdiMessageState</code>
<code>OperatorMessageId</code>	Задаёт или возвращает идентификатор сообщения у оператора. Тип: <code>System.String</code>

Имя	Описание
OrganizationBox	Задает или возвращает ящик организации. Тип: OperatorsBox
OrganizationDepartment	Задает или возвращает подразделение организации. Тип: OperatorsDepartment
Partner	Задает или возвращает контрагента. Тип: PartnersCompany
PartnerBox	Задает или возвращает ящик контрагента. Тип: OperatorsPartnerBox
PartnerDepartment	Задает или возвращает подразделение контрагента. Тип: OperatorsPartnerDepartment

Поля

Имя	Описание
CardIdProperty	Определяет свойство <i>Карточка</i> .
MessageStateProperty	Определяет свойство <i>Состояние</i> .
OperatorMessageIdProperty	Определяет свойство <i>Идентификатор сообщения у оператора</i> .
OrganizationBoxProperty	Определяет свойство <i>Ящик организации</i> .
OrganizationDepartmentProperty	Определяет свойство <i>Подразделение организации</i> .
PartnerBoxProperty	Определяет свойство <i>Ящик контрагента</i> .
PartnerDepartmentProperty	Определяет свойство <i>Подразделение контрагента</i> .

Имя	Описание
PartnerProperty	Определяет свойство <i>Контрагент</i> .

EdiMessageReplyFile — класс

Содержит данные ответного файла, полученного от контрагента, в карточке обмена сообщениями.

- **Пространство имен:** DocsVision.Edi.ObjectModel
- **Сборка:** DocsVision.Edi.ObjectModel.dll

Синтаксис

```
public class EdiMessageReplyFile : BaseCardSectionRow
```

Свойства

Имя	Описание
NeedReceipt	Задает или возвращает флаг, указывающий на необходимость подтверждения получения. Тип: System.Boolean
ParentMessageFile	Возвращает ссылку на родительское сообщение электронного обмена. Тип: undefined/programmer/dv5/Edi/api/EdiMessageFile/[EdiMessageFile]
ReplyFile	Задает или возвращает ответный файл. Тип: undefined/programmer/dv5/Edi/api/EdiMessageSignedFile/[EdiMessageSignedFile]

Имя	Описание
ReplyFileType	Задает или возвращает тип ответного файла. Тип: undefined/programmer/dv5/Edi/api/EdiMessageReplyFileType/[EdiMessageReplyFileType]
ReplyReceipt	Задает или возвращает ссылку на ответный файл (квитанцию). Тип: undefined/programmer/dv5/Edi/api/EdiMessageSignedFile/[EdiMessageSignedFile]

Поля

Имя	Описание
NeedReceiptProperty	Определяет свойство <i>Требуется извещение</i> .
ReplyFileProperty	Определяет свойство <i>Ответный файл</i> .
ReplyFileTypeProperty	Определяет свойство <i>Тип ответного файла</i> .
ReplyReceiptProperty	Определяет свойство <i>Ответная квитанция</i> .

EdiMessageSignedComment — класс

Содержит данные подписанного сообщения из *Карточки обмена сообщениями*.

- **Пространство имен:** DocsVision.Edi.ObjectModel
- **Сборка:** DocsVision.Edi.ObjectModel.dll

Синтаксис

```
public class EdiMessageSignedComment : BaseCardSectionRow
```

Свойства

Имя	Описание
<code>Comment</code>	Задает или возвращает содержимое комментария.
<code>SignatureId</code>	Задает или возвращает идентификатор подписи, которой подписан комментарий.

Поля

Имя	Описание
<code>CommentProperty</code>	Определяет свойство <i>Комментарий</i> .
<code>SignatureIdProperty</code>	Определяет свойство <i>Подпись</i> .

EdiMessageSignedFile — класс

Содержит данные подписанного файла в карточке обмена сообщениями.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public class EdiMessageSignedFile : BaseCardSectionRow
```

Свойства

Имя	Описание
<code>FileId</code>	Задает или возвращает идентификатор файла в Docsvision.
<code>FileName</code>	Задает или возвращает название файла.
<code>OperatorFileId</code>	Задает или возвращает идентификатор файла у оператора ЭДО.
<code>OperatorSignatureId</code>	Задает или возвращает идентификатор подписи файла у оператора ЭДО.

Имя	Описание
<code>SignatureId</code>	Задает или возвращает идентификатор подписи файла в Docsvision.

Поля

Имя	Описание
<code>FileIdProperty</code>	Определяет свойство <i>Файл</i> .
<code>FileNameProperty</code>	Определяет свойство <i>Имя файла</i> .
<code>OperatorFileIdProperty</code>	Определяет свойство <i>Идентификатор файла у оператора</i> .
<code>OperatorSignatureIdProperty</code>	Определяет свойство <i>Идентификатор подписи у оператора</i> .
<code>SignatureIdProperty</code>	Определяет свойство <i>Подпись</i> .

EdiMessageTransferData — класс

Содержит полученные от контрагента файлы, которые должны быть загружены в карточку УПД, а также новое состояние для карточки УПД.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public class EdiMessageTransferData : BaseCardSectionRow
```

Свойства

Имя	Описание
<code>Date</code>	Задает или возвращает дату получения данных от контрагента.
<code>NewStateId</code>	Задает или возвращает новое состояние для карточки УПД.

Имя	Описание
<code>TransferFiles</code>	Задает или возвращает список файлов, которые должны быть загружены в карточку УПД.

Поля

Имя	Описание
<code>DateProperty</code>	Определяет свойство "Дата".
<code>NewStateIdProperty</code>	Определяет свойство "Новое состояние карточки".
<code>TransferFilesProperty</code>	Определяет свойство "Файлы для переноса".

EdiPowerOfAttorney — класс

Машиночитаемая доверенность ЮЗДО.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public class EdiPowerOfAttorney : BaseCard
```

Свойства

Имя	Описание
<code>MainInfo</code>	Задает или возвращает основную информацию о доверенности. Тип: <code>EdiPowerOfAttorneyMainInfo</code>
<code>Log</code>	Задает или возвращает запись журнала МЧД Тип: <code>EdiPowerOfAttorneyLogItem</code>

Имя	Описание
<code>undefined/programmer/dv5/Edi/api/PowerOfAttorneyData/[PowerOfAttorneyData]</code>	Задает или возвращает данные МЧД Тип: <code>EdiPowerOfAttorneyData</code>

EdiPowerOfAttorneyData — класс

Данные машиночитаемой доверенности ЮЗДО.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public class EdiPowerOfAttorneyData : BaseCardSectionRow
```

Свойства

Имя	Описание
<code>RegistrationNumber</code>	Задает или возвращает регистрационный номер доверенности. Тип: <code>Guid</code>
<code>IssuerInn</code>	Задает или возвращает ИНН доверителя Тип: <code>string</code>
<code>RepresentativeInn</code>	ИНН доверенного лица Тип: <code>string</code>
<code>IsDefault</code>	Задает или возвращает признак использования по умолчанию Тип: <code>bool</code>

EdiPowerOfAttorneyEventType — перечисление

Тип события МЧД

- **Пространство имен:** DocsVision.Edi.ObjectModel
- **Сборка:** DocsVision.Edi.ObjectModel.dll

Синтаксис

```
public enum EdiPowerOfAttorneyEventType
```

Члены

Имя	Описание
Information = 0,	Информация
Error = 1,	Ошибка
Reply = 2,	Ответ
FromDocsvision = 3,	От Docsvision
FromOperator = 5	От оператора ЮЗДО

EdiPowerOfAttorneyLogItem — класс

Запись в журнале МЧД

- **Пространство имен:** DocsVision.Edi.ObjectModel
- **Сборка:** DocsVision.Edi.ObjectModel.dll

Синтаксис

```
public class EdiPowerOfAttorneyLogItem : BaseCardSectionRow
```

Свойства

Имя	Описание
EventType	Задаёт или возвращает тип события Тип: undefined/programmer/dv5/Edi/api/EdiPowerOfAttorneyEventType/[EdiPowerOfAttorneyEventType]

Имя	Описание
Date	Задает или возвращает дату события Тип: <code>DateTime</code>
Description	Задает или возвращает описание события Тип: <code>string</code>
Comment	Задает или возвращает комментарий к событию Тип: <code>string</code>
FileName	Задает или возвращает имя файла Тип: <code>string</code>

EdiPowerOfAttorneyMainInfo — класс

Основная информация о доверенности

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public class EdiPowerOfAttorneyMainInfo : BaseCardSectionRow
```

Свойства

Имя	Описание
CardId	Задает или возвращает идентификатор карточки Тип: <code>Guid</code>

Имя	Описание
MessageState	Задает или возвращает состояние сообщения. Тип: <code>EdiPowerOfAttorneyState</code>
OrganizationBox	Задает или возвращает ящик организации Тип: <code>OperatorsBox</code>
OrganizationDepartment	Задает или возвращает подразделение организации Тип: <code>OperatorsDepartment</code>
OperatorMessageId	Задает или возвращает идентификатор сообщения оператора ЮЗДО Тип: <code>string</code>
DocumentType	Задает или возвращает тип документа Тип: <code>string</code>
DocumentVersion	Задает или возвращает версию документа Тип: <code>string</code>
IncomingMessage	Задает или возвращает входящее сообщение оператора ЮЗДО Тип: <code>bool</code>
IsRecalled	Задает или возвращает признак отозванности Тип: <code>bool</code>

Имя	Описание
<code>PowerOfAttorneyCardId</code>	Задает или возвращает идентификатор карточки доверенности Тип: <code>Guid</code>
<code>SendAsFile</code>	Отправляет МЧД как файл. Тип: <code>bool</code>
<code>OperatorFileId</code>	Задает или возвращает идентификатор файла у оператора ЭДО Тип: <code>string</code>
<code>OperatorSignatureId</code>	Задает или возвращает идентификатор подписи оператора ЮЗДО Тип: <code>string</code>

EdiPowerOfAttorneyState — перечисление

Состояние доверенности

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public enum EdiPowerOfAttorneyState
```

Свойства

Имя	Описание
<code>Creating = 0</code>	Создание
<code>ToSend = 1</code>	Готово к отправке
<code>SentToOperator = 2</code>	Отправлена оператору
<code>Registered = 3</code>	Зарегистрирована
<code>AssignedToEmployee = 4</code>	Передана сотруднику

Имя	Описание
Error = 5	Ошибка
Revoked = 6	Отозвана

MessageAddress — класс

Содержит данные отправителя или получателя сообщения электронного обмена.

- **Пространство имен:** DocsVision.Edi.Entities
- **Сборка:** DocsVision.Edi.Interfaces.dll

Синтаксис

```
public class MessageAddress
```

Конструкторы

Имя	Описание
MessageAddress(String)	Инициализирует новый экземпляр класса MessageAddress. Параметры: • boxId — идентификатор ящика электронного обмена
MessageAddress(String, String)	Инициализирует новый экземпляр класса MessageAddress. Параметры: • boxId — идентификатор ящика электронного обмена • name — название ящика электронного обмена

Свойства

Имя	Описание
BoxId	Задает или возвращает идентификатор ящика электронного обмена. Тип: <code>System.String</code>
DepartmentId	Задает или возвращает идентификатор подразделения отправителя или получателя. Тип: <code>System.String</code>
Name	Задает или возвращает название отправителя или получателя. Тип: <code>System.String</code>

MessageData — класс

Содержит данные сообщения электронного обмена с оператором ЭДО.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public class MessageData
```

Конструкторы

Имя	Описание
MessageData(Guid)	Инициализирует новый экземпляр класса <code>MessageData</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки

Имя	Описание
<code>MessageData(Guid, String)</code>	Инициализирует новый экземпляр класса <code>MessageData</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки • <code>documentName</code> — название документа
<code>MessageData(String, String, Boolean)</code>	Инициализирует новый экземпляр класса <code>MessageData</code> по данным входящего сообщения. <i>Параметры:</i> • <code>operatorEventId</code> — идентификатор события у оператора ЭДО • <code>operatorMessageId</code> — идентификатор сообщения у оператора ЭДО • <code>isPatch</code> — является дополнением к имеющемуся сообщению

Свойства

Имя	Описание
<code>AdditonalAttributes</code>	Возвращает коллекцию дополнительных атрибутов. Тип: <code>ItemCollection<MessageAdditonalAttribute></code>
<code>CardId</code>	Задаёт или возвращает идентификатор карточки, по которой получено сообщение. Тип: <code>System.Guid</code>
<code>DocumentName</code>	Задаёт или возвращает название документа. Тип: <code>System.String</code>

Имя	Описание
DocumentNumber	Задает или возвращает регистрационный номер документа. Тип: <code>System.String</code>
DocumentDate	Задает или возвращает дату регистрации или дату создания документа. Тип: <code>DateTime?</code>
DocumentComment	Задает или возвращает содержание документа. Тип: <code>System.String</code>
OperatorMessageId	Задает или возвращает идентификатор сообщения у оператора ЭДО. Тип: <code>System.String</code>
OperatorEventId	Задает или возвращает идентификатор события у оператора ЭДО. Тип: <code>System.String</code>
OperatorDate	Задает или возвращает штамп времени оператора ЭДО. Тип: <code>DateTime</code>
IsPatch	Задает или возвращает флаг, указывающий является ли данное сообщение дополнением к имеющемуся сообщению: <code>true</code> — является, <code>false</code> — не является. Тип: <code>Boolean</code>

Имя	Описание
DocumentFiles	Возвращает коллекцию файлов сообщения. Тип: ItemCollection<undefined/programmer/dv5/Edi/api/MessageFile/[MessageFile]>
Sender	Задаёт или возвращает данные отправителя сообщения. Тип: undefined/programmer/dv5/Edi/api/MessageAddress/[MessageAddress]
Recipient	Задаёт или возвращает данные получателя сообщения. Тип: undefined/programmer/dv5/Edi/api/MessageAddress/[MessageAddress]
InvitationPartnerData	Задаёт или возвращает данные о приглашении партнёра к обмену. Тип: InvitationPartnerData
PowerOfAttorneyData	Задаёт или возвращает данные МЧД. Тип: undefined/programmer/dv5/Edi/api/PowerOfAttorneyData/[PowerOfAttorneyData]
AdditionalAttributes	Задаёт или возвращает дополнительные атрибуты сообщения. Тип: MessageAdditionalAttribute

MessageFile — класс

Содержит данные файла электронного обмена с оператором ЭДО.

- **Пространство имен:** DocsVision.Edi.Entities
- **Сборка:** DocsVision.Edi.Interfaces.dll

```
public class MessageFile
```

Конструкторы

Имя	Описание
<code>MessageFile(Guid, String)</code>	Инициализирует новый экземпляр класса <code>MessageFile</code>. <i>Параметры:</i> • <code>fileId</code> — идентификатор файла. • <code>fileName</code> — название файла.
<code>MessageFile(String, String, String)</code>	Инициализирует новый экземпляр класса <code>MessageFile</code> с указанием идентификатора файла у оператора ЭДО, названием файла и идентификатором подписи у оператора ЭДО. <i>Параметры:</i> • <code>operatorFileId</code> — идентификатор файла у оператора ЭДО. • <code>fileName</code> — название файла. • <code>operatorSignatureId</code> — идентификатор подписи файла у оператора ЭДО.

Имя	Описание
<code>MessageFile(MessageFileType, String, String, String)</code>	Инициализирует новый экземпляр класса <code>MessageFile</code> с указанием типа сообщения, идентификатора файла у оператора ЭДО, названием файла и идентификатором подписи у оператора ЭДО. <i>Параметры:</i> • <code>fileType</code> — тип содержимого. • <code>operatorFileId</code> — идентификатор файла у оператора ЭДО. • <code>fileName</code> — название файла. • <code>operatorParentEntityId</code> — идентификатор сущности, к которой относится файл у оператора ЭДО.

Свойства

Имя	Описание
<code>FileComment</code>	Задаёт или возвращает комментарий к файлу. Тип: <code>System.String</code>
<code>FileData</code>	Задаёт или возвращает бинарные данные файла. Тип: <code>Byte</code>
<code>FileId</code>	Задаёт или возвращает идентификатор файла в Docsvision. Тип: <code>System.Guid</code>
<code>FileName</code>	Задаёт или возвращает название файла. Тип: <code>System.String</code>

Имя	Описание
FilePath	Задает или возвращает путь, по которому выгружен файл. Тип: <code>System.String</code>
FileType	Задает или возвращает тип содержимого. Тип: <code>undefined/programmer/dv5/Edi/api/MessageFileType/[MessageFileType]</code>
OperatorFileId	Задает или возвращает идентификатор файла у оператора ЭДО. Тип: <code>System.String</code>
OperatorParentEntityId	Задает или возвращает идентификатор сущности, к которой относится файл у оператора ЭДО. Тип: <code>System.String</code>
OperatorSignatureId	Задает или возвращает идентификатор подписи у оператора ЭДО. Тип: <code>System.String</code>
SignatureData	Задает или возвращает бинарные данные подписи файла. Тип: <code>Byte</code>
SignatureFilePath	Задает или возвращает путь, по которому выгружена подпись файла. Тип: <code>System.String</code>

MessageFileType — перечисление

Определяет возможные типы содержимого в `MessageFile`.

- **Пространство имен:** `DocsVision.Edi.Entities`

- **Сборка:** DocsVision.Edi.Interfaces.dll

Синтаксис

```
public enum MessageType
```

Члены

Имя члена	Описание
File	Файл.
SignatureConfirmation	Ответная подпись.
ReplySignature	Подпись файла.
DeliveryFailureReport	Не удалось доставить.
FileComment	Текстовый комментарий.
SignatureRejection	Отказ в формировании запрошенной подписи.
Invoice	Извещение.
InvoiceConfirmation	Подтверждение получения УПД.
InvoiceReceipt	Извещение о получении УПД.
InvoiceReply	Ответный титул УПД.
InvoiceCorrectionRequest	Запрос на уточнение УПД.

Partner — класс

Содержит данные контрагента, полученные от оператора ЭДО.

- **Пространство имен:** DocsVision.Edi.Entities
- **Сборка:** DocsVision.Edi.Interfaces.dll

Синтаксис

```
public class Partner
```

Конструкторы

Имя	Описание
<code>Partner(String, String, String)</code>	Инициализирует новый экземпляр класса <code>Partner</code>. <i>Параметры:</i> • <code>id</code> — идентификатор контрагента у оператора ЭДО. • <code>name</code> — название контрагента у оператора ЭДО. • <code>inn</code> — ИНН контрагента у оператора ЭДО.

Свойства

Имя	Описание
<code>Departments</code>	Возвращает коллекцию подразделений контрагента. Тип: <code>ItemCollection<undefined/programmer/dv5/Edi/api/Department/[Department]></code>
<code>Id</code>	Задаёт или возвращает идентификатор контрагента в системе оператора ЭДО. Тип: <code>System.String</code>
<code>Inn</code>	Задаёт или возвращает ИНН контрагента в системе оператора ЭДО. Тип: <code>System.String</code>
<code>LinkedBoxes</code>	Возвращает коллекцию связанных ящиков контрагента. Тип: <code>ItemCollection<undefined/programmer/dv5/Edi/api/Box/[Box]></code>

Имя	Описание
<code>Name</code>	Задаёт или возвращает название контрагента в системе оператора ЭДО. Тип: <code>System.String</code>
<code>PartnerBoxes</code>	Возвращает коллекцию ящиков контрагента. Тип: <code>ItemCollection<undefined/programmer/dv5/Edi/api/Box/[Box]></code>

SignatureValidation — класс

Содержит данные с результатами проверки электронной подписи.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public class SignatureValidation
```

Конструкторы

Имя	Описание
<code>SignatureValidation(Boolean, Boolean, X509Certificate2, String, String)</code>	Инициализирует новый экземпляр класса <code>SignatureValidation</code> с использованием переданных данных.

Свойства

Имя	Описание
<code>Certificate</code>	Возвращает сертификат подписи, с помощью которого выполнялась проверка.
<code>IsCertificateValid</code>	Возвращает флаг, указывающий что сертификат является действительным.

Имя	Описание
<code>IsValid</code>	Возвращает флаг, указывающий что подпись является действительной.
<code>SignerName</code>	Возвращает имя владельца сертификата.
<code>ValidationError</code>	Возвращает сообщение об ошибке, если она была при проверке подписи.

Unit — класс

Содержит данные организации, полученные от оператора ЭДО.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public class Unit
```

Конструкторы

Имя	Описание
<code>Unit(String, String, String)</code>	Инициализирует новый экземпляр класса <code>Unit</code>. <i>Параметры:</i> • <code>id</code> — идентификатор организации у оператора ЭДО. • <code>name</code> — название организации у оператора ЭДО. • <code>inn</code> — ИНН организации у оператора ЭДО.

Свойства

Имя	Описание
Boxes	Возвращает коллекцию ящиков организации. Тип: <code>ItemCollection<undefined/programmer/dv5/Edi/api/Box/[Box]></code>
Departments	Возвращает коллекцию подразделений организации. Тип: <code>ItemCollection<undefined/programmer/dv5/Edi/api/Department/[Department]></code>
Id	Задаёт или возвращает идентификатор организации в системе оператора ЭДО. Тип: <code>System.String</code>
Inn	Задаёт или возвращает идентификатор контрагента в системе оператора ЭДО. Тип: <code>System.String</code>
Name	Задаёт или возвращает название организации в системе оператора ЭДО. Тип: <code>System.String</code>

IDocumentCreator — интерфейс

Интерфейс `IDocumentCreator` определяет методы, которые должен реализовывать компонент `undefined/programmer/dv5/Edi/doc-creating/[создания документов]`.

- **Пространство имен:** `DocsVision.Edi.Interfaces`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public interface IDocumentCreator
```

Свойства

Имя	Описание
<code>AddReplyFilesToDocument(MessageData, Guid)</code>	Переносит файлы из электронного сообщения в целевую карточку. <i>Параметры:</i> • <code>messageData</code> — электронное сообщение, полученное от контрагента • <code>partnerId</code> — идентификатор контрагента (в Справочнике сотрудников), от которого получено сообщение В стандартной конфигурации модуля метод вызывается бизнес-процессом "Перенос файлов из сообщений ЮЗДО".

Имя	Описание
CreateDocument(MessageData, Guid, Guid)	Создаёт новую карточку по полученному от оператору ЭДО сообщению. Если карточка успешно создана, возвращает True. Тип возвращаемого значения: Boolean Параметры: • messageData — сообщение от оператора ЭДО с типом undefined/programmer/dv5/Edi/api/MessageData/[MessageData]. • partnerId — идентификатор отправившего электронный документ контрагента в системе Docsvision. • partnerDepartmentId — идентификатор подразделения контрагента, если электронный документ отправлен подразделением контрагента. В стандартной конфигурации модуля метод вызывается, когда поступил новый электронный документ.
Initialize(ObjectContext, Guid)	Инициализирует экземпляр компонента создания документов. Параметры: • objectContext — контекст объектов. • defaultKindId — идентификатор вида создаваемого документа.

IDocumentDataReader — интерфейс

Интерфейс **IDocumentDataReader** определяет методы, которые должен реализовывать undefined/programmer/dv5/Edi/card-data-reading/[компонент чтения данных карточки].

- **Пространство имен:** DocsVision.Edi.Interfaces
- **Сборка:** DocsVision.Edi.Interfaces.dll

Синтаксис

```
public interface IDocumentDataReader
```

Методы

Имя	Описание
<code>GetActualSignatures(Guid)</code>	Возвращает актуальные подписи (действительные с действующим сертификатом) карточки. Тип возвращаемого значения: <code>ItemCollection<undefined/programmer/dv5/Edi/api/DocumentSignature/[DocumentSignature]></code> <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки, подписи которой возвращаются. В стандартной конфигурации модуля метод вызывается при открытии формы отправки электронного документа.

Имя	Описание
<code>GetAllFiles(Guid)</code>	Возвращает основные и дополнительные файлы карточки. Тип возвращаемого значения: <code>ItemCollection<undefined/programmer/dv5/Edi/api/DocumentFileData/[DocumentFileData]></code> <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки, файлы которой возвращаются. В стандартной конфигурации модуля метод вызывается при отправке электронного документа.
<code>GetAllRecipients(Guid)</code>	Возвращает список получателей электронного документа. Тип возвращаемого значения: <code>ItemCollection<undefined/programmer/dv5/Edi/api/DocumentRecipientData/[DocumentRecipientData]></code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки, из которой запрашиваются получатели. В стандартной конфигурации модуля метод вызывается при отправке электронного документа.

Имя	Описание
<code>GetDigest(Guid)</code>	Предоставляет описание (дайджест) карточки. Тип возвращаемого значения: <code>String</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. В стандартной конфигурации модуля метод вызывается при открытии формы отправки электронного документа.
<code>GetFileContent(Guid, Guid)</code>	Предоставляет содержимое файла карточки. Тип возвращаемого значения: <code>System.String</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>fileId</code> — идентификатор файла, для которого возвращается содержимое. В стандартной конфигурации модуля метод вызывается при открытии формы отправки электронного документа.

Имя	Описание
GetFileDigest(Guid, Guid)	Предоставляет описание (дайджест) файла карточки. Тип возвращаемого значения: System.String. <i>Параметры:</i> • cardId — идентификатор карточки. • fileId — идентификатор файла, для которого возвращается описание. В стандартной конфигурации модуля метод вызывается при открытии формы отправки электронного документа.

Имя	Описание
<code>GetInvoiceReplyData(Guid)</code>	Возвращает строку с ответом на сообщение, полученное от оператора ЭДО. Тип возвращаемого значения: <code>System.String</code>. Параметры: • <code>cardId</code> — идентификатор карточки. В стандартной конфигурации модуля метод вызывается при формировании ответной подписи. Метод <code>GetInvoiceReplyData</code> должен вернуть строку с данными карточки в определенном формате, который ожидается методом <code>IMessageService.GenerateInvoiceReply</code> коннектора к оператору ЭДО. В реализации коннектора к Диадок для ответа по УПД метод <code>GetInvoiceReplyData</code> должен вернуть сериализованный объект <code>DocsVision.Edi.Entities.BuyerReplyData</code>, заполненный данными карточки УПД, по которой формируется ответ.
<code>GetLastSignedFiles(Guid)</code>	Возвращает последние (по времени) подписанные файлы карточки. Тип возвращаемого значения: <code>ItemCollection<undefined/programmer/dv5/Edi/api/DocumentFileData/[DocumentFileData]></code>. Параметры: • <code>cardId</code> — идентификатор карточки.

Имя	Описание
<code>GetPrintFormContent(Guid)</code>	Возвращает содержимое карточки, подготовленное для печати. Тип возвращаемого значения: <code>String</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. В стандартной конфигурации модуля метод вызывается печати УПД.
<code>GetReplyFileComment(Guid, Guid, MessageFileType)</code>	Формирует строку с комментарием к файлу отправленного электронного сообщения. Тип возвращаемого значения: <code>System.String</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>fileId</code> — идентификатор файла, для которого нужен комментарий. • <code>fileType</code> — ответ контрагента на переданный в электронном сообщении файл. В стандартной конфигурации модуля метод вызывается при формировании сообщений для журнала обмена ЮЗДО.

Имя	Описание
<code>GetReplyFileDigest(Guid, Guid, MessageFileType)</code>	Формирует строку с дайджестом файла отправленного электронного сообщения. Тип возвращаемого значения: <code>System.String</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>fileId</code> — идентификатор файла, для которого нужен дайджест. • <code>fileType</code> — ответ контрагента на переданный в электронном сообщении файл. В стандартной конфигурации модуля метод вызывается при формировании сообщений для журнала обмена ЮЗДО.
<code>GetSignedFiles(Guid, Guid)</code>	Возвращает файлы карточки, которые подписаны указанной подписью. Тип возвращаемого значения: <code>ItemCollection<undefined/programmer/dv5/Edi/api/DocumentFileData/[DocumentFileData]></code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>signatureGroupId</code> — идентификатор ЭЦП. В стандартной конфигурации модуля метод вызывается при открытии формы отправки электронного документа.

Имя	Описание
<code>Initialize(ObjectContext)</code>	Инициализирует экземпляр компонента отправки электронного документа. <i>Параметры:</i> • <code>objectContext</code> — контекст объектов.
<code>PrepareFileData(Guid, Guid, Guid, Boolean, String)</code>	Создаёт экземпляр файла сообщения электронного обмена. Тип возвращаемого значения: <code>undefined/programmer/dv5/Edi/api/MessageFile/[MessageFile]</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки, содержащий файл. • <code>fileId</code> — идентификатор файла, передаваемого в сообщении. • <code>signatureId</code> — идентификатор подписи, которой подписан файл. • <code>isMain</code> — файл является основным. • <code>tempFolder</code> — временный каталог, для загрузки файла. В стандартной конфигурации модуля метод вызывается при отправке оператору ЭДО сообщения электронного обмена с новым документом. В большинстве случаев полностью реализовывать данный метод не требуется, т.к. принцип подготовки файла электронного сообщения стандартный. Чтобы использовать стандартный механизм формирования, достаточно вернуть из метода значение <code>null</code>.

Имя	Описание
<code>PrepareMessageData(Guid, String, String)</code>	Создаёт экземпляр электронного сообщения электронного обмена. Тип возвращаемого значения: <code>undefined/programmer/dv5/Edi/api/MessageData/[MessageData]</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки, для которой формируется сообщение электронного обмена для отправки оператору ЭДО. • <code>documentType</code> — тип документа. • <code>documentVersion</code> — версия документа. В стандартной конфигурации модуля метод вызывается при отправке оператору ЭДО сообщения электронного обмена с новым документом.

Имя	Описание
<code>ValidateSignature(Guid, Guid, Guid)</code>	Проверяет подпись файла и возвращает объект, содержащий результаты проверки. Тип возвращаемого значения: <code>undefined/programmer/dv5/Edi/api/SignatureValidation/[SignatureValidation]</code> <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки с проверяемым файлом. • <code>fileId</code> — идентификатор файла карточки, подпись которого проверяется. • <code>signatureId</code> — идентификатор подписи. В стандартной конфигурации модуля метод вызывается при работе с Журналом обмена ЮЗДО, если в процессе обмена УПД от контрагента поступили подписанные файлы.

IDocumentUpdater — интерфейс

Интерфейс `IDocumentUpdater` определяет методы, которые должен реализовывать `undefined/programmer/dv5/Edi/card-data-editing/[компонент изменения данных карточки]`.

- **Пространство имен:** `DocsVision.Edi.Interfaces`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public interface IDocumentUpdater
```

Методы

Имя	Описание
AddSignatureToDocument(Guid, Guid, MessageFile, String)	Добавляет подпись файла в карточку. Тип возвращаемого значения: System.Guid. <i>Параметры:</i> • cardId — идентификатор карточки. • fileId — идентификатор подписываемого файла. • messageFile — поступившее сообщение электронного обмена, содержащее подпись файла. • partnerName — название контрагента. В стандартной конфигурации модуля метод вызывается, когда от оператора ЭДО поступило сообщение с ответной подписью.

Имя	Описание
CreateNewSignatures(Guid, X509Certificate2, Boolean)	Подписывает карточку переданным сертификатом. Тип возвращаемого значения: System.Guid. <i>Параметры:</i> • cardId — идентификатор подписываемой карточки. • certificate — сертификат, которым подписывается карточка. • signAdditionalFiles — подписывать дополнительные файлы: true — подписывать; false — не подписывать. В стандартной конфигурации модуля метод вызывается, когда пользователь подписывает карточку отправляемого электронного документа.
CheckCertificate(Guid, X509Certificate2)	Проверяет соответствие подписи в указанной карточке переданному сертификату и возвращает результат проверки. Тип возвращаемого значения: System.String. <i>Параметры:</i> • cardId — идентификатор проверяемой карточки. • certificate — сертификат, для которого выполняется проверка. В стандартной конфигурации модуля метод вызывается при отправке УПД пользователем.

Имя	Описание
<code>ChangeDocumentState(Guid, Guid)</code>	Изменяет состояние карточки. Если изменение было успешным, метод возвращает <code>true</code>, иначе — <code>false</code>. Тип возвращаемого значения: <code>System.Boolean</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>newStateId</code> — идентификатор нового состояния. В стандартной конфигурации модуля метод вызывается при обработке входящих сообщений бизнес-процессом <i>Перенос файлов из сообщений ЮЗДО</i>.
<code>GetNewStateId(MessageFileType)</code>	Возвращает идентификатор встроенного состояния для карточки УПД, сопоставленного типу ответа контрагента на отправленный ранее УПД. Тип возвращаемого значения: <code>System.Guid</code>. <i>Параметры:</i> • <code>messageFileType</code> — Тип файла сообщения В стандартной конфигурации модуля метод вызывается, когда пользователь подписывает карточку отправляемого электронного документа.

Имя	Описание
<code>Initialize(ObjectContext)</code>	Инициализирует экземпляр компонента создания документов. <i>Параметры:</i> • <code>objectContext</code> — контекст объектов.
<code>UpdateDocumentDataFromFile(Guid, Guid)</code>	Обновляет содержимое карточки данными из файла. <i>Параметры:</i> • <code>cardId</code> — идентификатор обновляемой карточки. • <code>fileId</code> — идентификатор файла, содержащего данные для обновления карточки. В стандартной конфигурации модуля метод вызывается, когда пользователь использует команду Заполнить данные из файла в карточке УПД.
<code>AddLinkToMessageCard(Guid, Guid);</code>	Добавляет ссылку на карточку сообщения ЮЗДО <i>Параметры:</i> • <code>messageCardId</code> — Идентификатор карточки сообщения • <code>documentCardId</code> — Идентификатор карточки документа
<code>UpdatePowerOfAttorneyStatus(Guid, PowerOfAttorneyData)</code>	Устанавливает или снимает флаг Использовать по умолчанию <i>Параметры:</i> • <code>cardId</code> — Идентификатор карточки • <code>powerOfAttorneyData</code> — Данные МЧД

Имя	Описание
<code>ChangePowerOfAttorneyState(Guid, Guid, MessageType);</code>	Изменяет состояние документа и МЧД <i>Параметры:</i> • <code>cardId</code> — Идентификатор карточки • <code>powerOfAttorneyCardId</code> — Идентификатор карточки доверенности • <code>messageFileType</code> — Тип файла сообщения Возвращаемое значение: <code>true</code>, если состояние изменилось

IEdiPowerOfAttorneyService — интерфейс

Сервис для работы с машиночитаемой доверенностью ЮЗДО

- **Пространство имен:** `DocsVision.Edi.ObjectModel.Services`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public interface IEdiPowerOfAttorneyService
```

Методы

Имя	Описание
<code>CreateEdiPowerOfAttorney(Guid, OperatorsBox)</code>	Создаёт Карточку обмена сообщениями для МЧД: <i>Параметры:</i> • <code>cardId</code> — Идентификатор карточки • <code>organizationBox</code> — Ящик организации

Имя	Описание
<code>AddLogMessage(EdiPowerOfAttorney, EdiPowerOfAttorneyEventType, string)</code>	Добавляет сообщение в журнал Параметры: • <code>message</code> — Карточка сообщения • <code>eventType</code> — Тип события • <code>description</code> — Описание
<code>AddLogMessage(EdiPowerOfAttorney, EdiPowerOfAttorneyEventType, string, string)</code>	Добавляет сообщение в журнал Параметры: • <code>message</code> — Карточка сообщения • <code>eventType</code> — Тип события • <code>description</code> — Описание • <code>comment</code> — Комментарий
<code>AddLogMessage(EdiPowerOfAttorney, EdiPowerOfAttorneyEventType, string, string, string)</code>	Добавляет сообщение в журнал Параметры: • <code>message</code> — Карточка сообщения • <code>eventType</code> — Тип события • <code>description</code> — Описание • <code>comment</code> — Комментарий • <code>fileName</code> — Имя файла

IMessageService — интерфейс

Интерфейс `IMessageService` определяет методы, которые должен реализовывать компонент `undefined/programmer/dv5/Edi/message-service/[сервиса сообщений]`.

- **Пространство имен:** `DocsVision.Edi.Interfaces`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public interface IMessageService
```


Методы

Имя	Описание
<code>EventExists(String, String)</code>	Проверяет наличие события с указанным идентификатором в ящике организации. Возвращает <code>True</code>, если событие есть. Тип возвращаемого значения: <code>Boolean</code>. <i>Параметры:</i> • <code>boxId</code> — идентификатор ящика организации. • <code>eventId</code> — идентификатор события. В стандартной конфигурации модуля метод вызывается при получении новых сообщений от оператора ЭДО.

Имя	Описание
GenerateCorrectionRequest(String, String, String, X509Certificate2, String, String)	Формирует запрос на уточнение. Тип возвращаемого значения: <code>undefined/programmer/dv5/Edi/api/MessageFile/[MessageFile]</code>. <i>Параметры:</i> • <code>boxId</code> — идентификатор ящика организации. • <code>messageId</code> — идентификатор сообщения у оператора обмена. • <code>entityId</code> — идентификатор сущности у оператора ЭДО, по которому отправляется запрос на уточнение. • <code>certificate</code> — сертификат, которым подписывается запрос. • <code>positionName</code> — должность подписывающего запрос сотрудника. • <code>comment</code> — комментарий к запросу. В стандартной конфигурации модуля метод вызывается, когда пользователь запрашивает уточнение по поступившему документу.

Имя	Описание
GenerateInvoiceReceipt(String, String, String, X509Certificate2, String)	Формирует ответную подпись на полученный документ. Тип возвращаемого значения: undefined/programmer/dv5/Edi/api/MessageFile/[MessageFile]. <i>Параметры:</i> • boxId — идентификатор ящика организации. • messageId — идентификатор сообщения у оператора обмена. • entityId — идентификатор сущности у оператора ЭДО, извещение о получении которой отправляется. • certificate — сертификат, которым подписывается извещение. • positionName — должность подписывающего извещение сотрудника. В стандартной конфигурации модуля метод вызывается, когда пользователь вызывает функцию отправки извещения.

Имя	Описание
<code>GenerateInvoiceReply(String, String, String, X509Certificate2, String, String)</code>	Формирует ответную подпись для поступившего документа. Тип возвращаемого значения: <code>undefined/programmer/dv5/Edi/api/MessageFile/[MessageFile]</code>. <i>Параметры:</i> • <code>boxId</code> — идентификатор ящика организации. • <code>messageId</code> — идентификатор сообщения у оператора обмена. • <code>entityId</code> — идентификатор сущности у оператора ЭДО, по которой формируется подпись. • <code>certificate</code> — сертификат, которым подписывается запрос. • <code>documentType</code> — тип документа. • <code>replyData</code> — содержимое ответа. В стандартной конфигурации модуля метод вызывается, когда пользователь подписывает поступивший документ.

Имя	Описание
GenerateRevocationRequest(String, String, String, X509Certificate2, String, String)	Формирует запрос на аннулирование. Тип возвращаемого значения: <code>undefined/programmer/dv5/Edi/api/MessageFile/[MessageFile]</code>. <i>Параметры:</i> • <code>boxId</code> — идентификатор ящика организации. • <code>messageId</code> — идентификатор сообщения у оператора обмена. • <code>entityId</code> — идентификатор сущности у оператора ЭДО, по которому отправляется запрос на аннулирование. • <code>certificate</code> — сертификат, которым подписывается запрос. • <code>positionName</code> — должность подписывающего запрос сотрудника. • <code>comment</code> — комментарий к запросу. В стандартной конфигурации модуля метод вызывается, когда пользователь запрашивает аннулирование.

Имя	Описание
GenerateSignatureRejection(String, String, String, X509Certificate2, String, String)	Формирует отказ в подписании. Тип возвращаемого значения: <code>undefined/programmer/dv5/Edi/api/MessageFile/[MessageFile]</code>. <i>Параметры:</i> • <code>boxId</code> — идентификатор ящика организации. • <code>messageId</code> — идентификатор сообщения у оператора обмена. • <code>entityId</code> — идентификатор сущности у оператора ЭДО, по которому отправляется отказ в подписании. • <code>certificate</code> — сертификат, которым подписывается ответ. • <code>positionName</code> — должность подписывающего запрос сотрудника. • <code>comment</code> — комментарий к отказу. В стандартной конфигурации модуля метод вызывается, когда пользователь отказывает в подписании поступившего документа.

Имя	Описание
<code>GetLastEventId(String)</code>	Возвращает идентификатор последнего события в ящике организации. Тип возвращаемого значения: <code>String</code>. <i>Параметры:</i> • <code>boxId</code> — идентификатор ящика организации. В стандартной конфигурации модуля метод используется при активации ящика организации.
<code>GetNewEvents(String, String, DateTime?)</code>	Получает новые события из ящика организации. Тип возвращаемого значения: <code>ItemCollection<undefined/programmer/dv5/Edi/api/MessageData/[MessageData]></code>. <i>Параметры:</i> • <code>boxId</code> — идентификатор ящика организации. • <code>lastEventId</code> — идентификатор последнего полученного события (может отсутствовать). • <code>fromDate</code> — дата минимальная дата создания сообщения. Если значение указано, то должны возвращаться сообщения, созданные после указанной даты. В стандартной конфигурации модуля метод вызывается при получении новых сообщений от оператора ЭДО.

Имя	Описание
<code>GetNewEventsCount(String, String)</code>	Возвращает количество новых событий. Тип возвращаемого значения: <code>Int32</code>. <i>Параметры:</i> • <code>boxId</code> — идентификатор ящика организации. • <code>lastEventId</code> — идентификатор последнего полученного события (может отсутствовать). В стандартной конфигурации модуля метод вызывается при активации ящика организации.
<code>Initialize(Dictionary<String, String>)</code>	Создаёт экземпляр электронного сообщения электронного обмена. <i>Параметры:</i> • <code>settings</code> — коллекция настроек оператора ЭДО из <i>Справочника настроек операторов ЮЗДО</i>. Каждая запись содержит ключ и значение, например: <code>Login</code> и <code>Docsvision@diadoc.ru</code>.
<code>SendCorrection(MessageData)</code>	Отправляет запрос на уточнение. <i>Параметры:</i> • <code>messageData</code> — сообщение типа <code>undefined/programmer/dv5/Edi/api/MessageData/[MessageData]</code>, содержащее запрос на уточнение. В стандартной конфигурации модуля метод используется отправке новых сообщений оператору ЭДО.

Имя	Описание
<code>SendMessage(MessageData)</code>	Отправляет новое сообщение оператору ЭДО. <i>Параметры:</i> <code>messageData</code> — сообщение типа <code>undefined/programmer/dv5/Edi/api/MessageData/[MessageData]</code>. В стандартной конфигурации модуля метод используется отправке новых сообщений оператору ЭДО.
<code>SendReceipt(MessageData)</code>	Отправляет квитанцию. <i>Параметры:</i> <code>messageData</code> — сообщение типа <code>undefined/programmer/dv5/Edi/api/MessageData/[MessageData]</code>.
<code>SendRevocation(MessageData)</code>	Отправляет запрос на аннулирование. <i>Параметры:</i> <code>messageData</code> — сообщение типа <code>undefined/programmer/dv5/Edi/api/MessageData/[MessageData]</code>, содержащее запрос на аннулирование В стандартной конфигурации модуля метод используется отправке новых сообщений оператору ЭДО.

Имя	Описание
<code>SendSignatureReply(MessageData)</code>	Отправляет ответную подпись на полученное сообщение. <i>Параметры:</i> <code>messageData</code> — сообщение типа <code>undefined/programmer/dv5/Edi/api/MessageData/[MessageData]</code>. В стандартной конфигурации модуля метод используется при отправке оператору ЭДО ответа на полученное сообщение.

InvitationService — класс

Сервис работы с приглашениями, может быть использован для замены стандартных БП модуля.

Вызывает экземпляры `IInvitationService` для соответствующего оператора обмена.

- **Пространство имен:** `DocsVision.Edi.Runtime`
- **Сборка:** `DocsVision.Edi.Runtime.dll`

Синтаксис

```
public class InvitationService
```

Методы

Имя	Описание
<code>public void ProcessInvitationToSend(EdiInvitation, ILogWriter)</code>	Обрабатывает карточку приглашения
<code>public void CheckInvitationStatus(EdiInvitation, ILogWriter)</code>	Проверяет статус нового приглашения

Имя	Описание
<code>public ICollection<MessageData> GetNewInvitations(string, string, DateTime?)</code>	Получает новые приглашения с указанной даты по ID ящика и ID последнего события
<code>public void ProcessEvent(MessageData, ILogWriter)</code>	Обрабатывает событие

IPowerOfAttorneyService — интерфейс

Интерфейс `IPowerOfAttorneyService` определяет методы для работы с МЧД.

- **Пространство имен:** `DocsVision.Edi.Interfaces`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public interface IPowerOfAttorneyService
```

Методы

Имя	Описание
<code>RegisterPowerOfAttorney(MessageData);</code>	Регистрирует доверенность <i>Параметры:</i> • <code>undefined/programmer/dv5/Edi/api/MessageData/[messageData]</code>
<code>RevokePowerOfAttorney(MessageData);</code>	Отвязывает МЧД от сотрудника в Диадок, в реестре ФНС доверенность не отзывается <i>Параметры:</i> • <code>undefined/programmer/dv5/Edi/api/MessageData/[messageData]</code>
<code>CheckPowerOfAttorneyStatus(MessageData);</code>	Проверяет статус доверенности <i>Параметры:</i> • <code>undefined/programmer/dv5/Edi/api/MessageData/[messageData]</code>

OutgoingPowerOfAttorneyData — класс

Содержит данные исходящей доверенности.

- **Пространство имен:** DocsVision.Edi.Entities
- **Сборка:** DocsVision.Edi.Interfaces.dll

Синтаксис

```
public class OutgoingPowerOfAttorneyData
```

Свойства

Имя	Описание
CardId	Задаёт или возвращает идентификатор карточки. Тип: Guid
PowerOfAttorneyCardId	Задаёт или возвращает идентификатор карточки доверенности. Тип: Guid
SendAsFile	Отправляет МЧД как файл. Тип: bool
PowerOfAttorneyData	Задаёт или возвращает данные доверенности. Тип: undefined/programmer/dv5/Edi/api/PowerOfAttorneyData/[PowerOfAttorneyData]

OutgoingPowerOfAttorneyService — класс

Сервис создания исходящей доверенности

- **Пространство имен:** DocsVision.Edi.Runtime
- **Сборка:** DocsVision.Edi.Runtime.dll

```
public class OutgoingPowerOfAttorneyService
```

Методы

Имя	Описание
CreateOutgoingPowerOfAttorney(OperatorsBox, OperatorsDepartment, OutgoingPowerOfAttorneyData, out string)	Создаёт исходящую доверенность Тип возвращаемого значения: undefined/programmer/dv5/Edi/api/EdiPowerOfAttorney/[EdiPowerOfAttorney] <i>Параметры:</i> • organizationBox — Ящик организации • organizationDepartment — подразделение организации • powerOfAttorneyData — Данные МЧД • sendErrors — отправлять ошибки
CreateOutgoingPowerOfAttorneyRecall(OperatorsBox, OperatorsDepartment, OutgoingPowerOfAttorneyData, out string)	Создаёт отзыв исходящей доверенности Тип возвращаемого значения: undefined/programmer/dv5/Edi/api/EdiPowerOfAttorney/[EdiPowerOfAttorney] <i>Параметры:</i> • organizationBox — Ящик организации • organizationDepartment — подразделение организации • powerOfAttorneyData — Данные МЧД • sendErrors — отправлять ошибки

Имя	Описание
<code>FindAnyPowerOfAttorneyCards(Guid, Guid)</code>	Находит любую из карточек доверенности Тип возвращаемого значения: <code>undefined/programmer/dv5/Edi/api/EdiPowerOfAttorney/[EdiPowerOfAttorney]</code> <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки, для которой запрашивается МЧД • <code>powerOfAttorneyCardId</code> — идентификатор карточки МЧД
<code>FindAssignedPowerOfAttorneyCards(Guid, Guid)</code>	Находит назначенные карточки доверенности Тип возвращаемого значения: <code>undefined/programmer/dv5/Edi/api/EdiPowerOfAttorney/[EdiPowerOfAttorney]</code> <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки, для которой запрашивается МЧД • <code>powerOfAttorneyCardId</code> — идентификатор карточки МЧД

PowerOfAttorneyData — класс

Содержит данные машиночитаемой доверенности.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public class PowerOfAttorneyData
```

Свойства

Имя	Описание
RegistrationNumber	Задает или возвращает регистрационный номер Тип: Guid
IssuerInn	Задает или возвращает ИНН доверителя Тип: string
RepresentativeInn	Задает или возвращает ИНН доверенного лица Тип: string
IsDefault	Задает или возвращает признак использования по умолчанию Тип: bool
PowerOfAttorneyStatus	Задает или возвращает статус МЧД Тип: undefined/programmer/dv5/Edi/api/PowerOfAttorneyStatus/[PowerOfAttorneyStatus]
ValidationStatus	Задает или возвращает статус проверки доверенности Тип: ValidationStatus
ValidationErrors	Выявленные при проверке ошибки. Тип: string

PowerOfAttorneyDataReader — класс

Содержит методы для чтения машиночитаемой доверенности

- **Пространство имен:** DocsVision.Edi.Runtime.BackOffice

- **Сборка:** DocsVision.Edi.Runtime.Backoffice.dll

Синтаксис

```
public class PowerOfAttorneyDataReader : DocumentDataReader
```

Методы

Имя	Описание
<code>PrepareMessageData(Guid)</code>	Создаёт экземпляр электронного сообщения электронного обмена. Тип возвращаемого значения: undefined/programmer/dv5/Edi/api/MessageData/[MessageData]. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки, для которой формируется сообщение электронного обмена для отправки оператору ЭДО. • <code>documentType</code> — тип документа. • <code>documentVersion</code> — версия документа. В стандартной конфигурации модуля метод вызывается при отправке оператору ЭДО сообщения электронного обмена с новым документом.
<code>ReadAdditionalAttributes(MessageData, Document)</code>	Читает дополнительные атрибуты <i>Параметры:</i> • <code>messageData</code> — электронное сообщение, полученное от контрагента • <code>document</code> — документ

Имя	Описание
<pre>PrepareFileData(Guid, Guid, Guid, bool, string)</pre>	Создаёт экземпляр файла сообщения электронного обмена. Тип возвращаемого значения: undefined/programmer/dv5/Edi/api/MessageFile/[MessageFile]. <i>Параметры:</i> • cardId — идентификатор карточки, содержащий файл. • fileId — идентификатор файла, передаваемого в сообщении. • signatureId — идентификатор подписи, которой подписан файл. • isMain — файл является основным. • tempFolder — временный каталог, для выгрузки файла. В стандартной конфигурации модуля метод вызывается при отправке оператору ЭДО сообщения электронного обмена с новым документом. В большинстве случаев полностью реализовывать данный метод не требуется, т.к. принцип подготовки файла электронного сообщения стандартный. Чтобы использовать стандартный механизм формирования, достаточно вернуть из метода значение null.

Имя	Описание
<code>GetFileDigest(Guid, Guid)</code>	Предоставляет описание (дайджест) файла карточки. Тип возвращаемого значения: <code>System.String</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>fileId</code> — идентификатор файла, для которого возвращается описание. В стандартной конфигурации модуля метод вызывается при открытии формы отправки электронного документа.
<code>GetFileContent(Guid, Guid)</code>	Предоставляет содержимое файла карточки. Тип возвращаемого значения: <code>System.String</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>fileId</code> — идентификатор файла, для которого возвращается содержимое. В стандартной конфигурации модуля метод вызывается при открытии формы отправки электронного документа.

Имя	Описание
<code>GetReplyFileDigest(Guid, Guid, MessageFileType)</code>	Предоставляет дайджест файла ответа Тип возвращаемого значения: <code>System.String</code>. <i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>fileId</code> — идентификатор файла, для которого возвращается содержимое. • <code>fileType</code> — тип файла, для которого возвращается содержимое.

PowerOfAttorneyService — класс

Сервис машиночитаемой доверенности

- **Пространство имен:** `DocsVision.Edi.Runtime.Diadoc`
- **Сборка:** `DocsVision.Edi.Runtime.Diadoc.dll`

Синтаксис

```
public class PowerOfAttorneyService : IPowerOfAttorneyService
```

Методы

Имя	Описание
<code>Initialize(Dictionary<string, string>)</code>	Создаёт экземпляр электронного сообщения электронного обмена. <i>Параметры:</i> • <code>settings</code> — коллекция настроек оператора ЭДО из <i>Справочника настроек операторов ЮЗДО</i>. Каждая запись содержит ключ и значение, например: <code>Login</code> и <code>Docsvision@diadoc.ru</code>.

Имя	Описание
<code>RegisterPowerOfAttorney(MessageData)</code>	Регистрирует доверенность <i>Параметры:</i> <code>messageData</code> — сообщение типа <code>undefined/programmer/dv5/Edi/api/MessageData/[MessageData]</code>, содержащее запрос на регистрацию МЧД.
<code>RevokePowerOfAttorney(MessageData)</code>	Отвязывает МЧД от сотрудника в Диадок, в реестре ФНС доверенность не отзывается <i>Параметры:</i> <code>messageData</code> — сообщение типа <code>undefined/programmer/dv5/Edi/api/MessageData/[MessageData]</code>, содержащее запрос на отзыв МЧД.
<code>CheckPowerOfAttorneyStatus(MessageData)</code>	Проверяет статус доверенности <i>Параметры:</i> <code>messageData</code> — сообщение типа <code>undefined/programmer/dv5/Edi/api/MessageData/[MessageData]</code>, содержащее запрос на проверку статуса МЧД.

PowerOfAttorneyService — класс

Сервис машиночитаемой доверенности

- **Пространство имен:** `DocsVision.Edi.Runtime`
- **Сборка:** `DocsVision.Edi.Runtime.dll`

Синтаксис

```
public class PowerOfAttorneyService
```

Методы

Имя	Описание
<code>PowerOfAttorneyService(OperatorsOperator)</code>	Получает МЧД <i>Параметры:</i> • <code>ediOperator</code> — оператор ЮЗДО
<code>PowerOfAttorneyService(ObjectContext, OperatorsOperator, bool)</code>	Получает МЧД <i>Параметры:</i> • <code>objectContext</code> — контекст объектов. • <code>ediOperator</code> — оператор ЮЗДО • <code>useExtension</code> — использовать расширение
<code>ProcessPowerOfAttorneyToSend(EdiPowerOfAttorney, ILogWriter)</code>	Обрабатывает карточку МЧД <i>Параметры:</i> • <code>ediPowerOfAttorney</code> — машиночитаемая доверенность • <code>logWriter</code> — запись в журнале
<code>CheckPowerOfAttorneyStatus(EdiPowerOfAttorney, ILogWriter)</code>	Проверяет статус МЧД <i>Параметры:</i> • <code>ediPowerOfAttorney</code> — машиночитаемая доверенность • <code>logWriter</code> — запись в журнале

PowerOfAttorneyStatus — перечисление

Представляет статус доверенности.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public enum PowerOfAttorneyStatus
```

Члены

Имя	Описание
<code>Creating = 0</code>	Подготовка
<code>Registering = 2</code>	Регистрация
<code>Registered = 3</code>	Зарегистрирована
<code>AssignedToEmployee = 4</code>	Передана сотруднику
<code>Error = 5</code>	Ошибка
<code>Recalled = 6</code>	Отозвана

PowerOfAttorneyUpdater — класс

Класс обновления машиночитаемой доверенности

- **Пространство имен:** `DocsVision.Edi.Runtime.BackOffice`
- **Сборка:** `DocsVision.Edi.Runtime.Backoffice.dll`

Синтаксис

```
public class PowerOfAttorneyUpdater : IDocumentUpdater
```

Методы

Имя	Описание
<code>PowerOfAttorneyService(OperatorsOperator)</code>	Получает МЧД <i>Параметры:</i> • <code>ediOperator</code> — оператор ЮЗДО
<code>PowerOfAttorneyService(ObjectContext, OperatorsOperator, bool)</code>	Получает МЧД <i>Параметры:</i> • <code>objectContext</code> — контекст объектов. • <code>ediOperator</code> — оператор ЮЗДО • <code>useExtension</code> — использовать расширение

Имя	Описание
<code>Initialize(ObjectContext)</code>	<i>Параметры:</i> • <code>objectContext</code> — контекст объектов.
<code>AddSignatureToDocument(Guid, Guid, MessageFile, string)</code>	<i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>fileId</code> — идентификатор файла. • <code>messageFile</code> — подписываемый файл. • <code>partnerName</code> — название контрагента.
<code>CreateNewSignatures(Guid, X509Certificate2, bool)</code>	<i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>certificate</code> — сертификат подписи. • <code>signAdditionalFiles</code> — подписывать дополнительные файлы: <code>true</code> — подписывать; <code>false</code> — не подписывать.
<code>CheckCertificate(Guid, X509Certificate2)</code>	<i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>certificate</code> — сертификат подписи.
<code>ChangeDocumentState(Guid, Guid)</code>	<i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>newStateId</code> — идентификатор нового состояния.
<code>Guid GetNewStateId(MessageFileType)</code>	<i>Параметры:</i> • <code>messageFileType</code> — Тип файла сообщения
<code>UpdateDocumentDataFromFile(Guid, Guid)</code>	<i>Параметры:</i> • <code>cardId</code> — идентификатор карточки. • <code>fileId</code> — идентификатор подписываемого файла.

Имя	Описание
AddLinkToMessageCard(Guid, Guid)	<i>Параметры:</i> <code>messageCardId</code> — Идентификатор карточки сообщения. <code>documentCardId</code> — Идентификатор карточки документа.
UpdatePowerOfAttorneyStatus(Guid, PowerOfAttorneyData)	<i>Параметры:</i> <code>cardId</code> — идентификатор карточки. <code>powerOfAttorneyData</code> — Данные МЧД.
ChangePowerOfAttorneyState(Guid, Guid, MessageFileType)	<i>Параметры:</i> <code>cardId</code> — идентификатор карточки. <code>powerOfAttorneyCardId</code> — идентификатор карточки МЧД. <code>messageFileType</code> — Тип файла сообщения.

IImportService — интерфейс

Интерфейс `IImportService` определяет методы, которые должен реализовывать компонент `undefined/programmer/dv5/Edi/import-config/[импорта настроек ЭДО]`.

- **Пространство имен:** `DocsVision.Edi.Interfaces`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public interface IImportService
```

Методы

Имя	Описание
ImportPartners	Предоставляет список контрагентов и их ящиков, зарегистрированных у организации в системе оператора ЭДО. Тип возвращаемого значения: <code>ItemCollection<undefined/programmer/dv5/Edi/api/Partner/[Partner]></code>. В стандартной конфигурации модуля метод используется при вызове в <i>Справочнике настроек операторов ЮЗДО</i> команды загрузки данных контрагентов.
ImportUnits	Предоставляет список подразделений организации и их ящиков ящиками, зарегистрированных у организации в системе оператора ЭДО. Тип возвращаемого значения: <code>ItemCollection<undefined/programmer/dv5/Edi/api/Unit/[Unit]></code>. В стандартной конфигурации модуля метод используется при вызове в <i>Справочнике настроек операторов ЮЗДО</i> команды загрузки данных организации.
Initialize(Dictionary<string, string>)	Создаёт экземпляр электронного сообщения электронного обмена. <i>Параметры:</i> • <code>settings</code> — коллекция настроек оператора ЭДО из <i>Справочника настроек операторов ЮЗДО</i>. Каждая запись содержит ключ и значение, например: <code>Login</code> и <code>Docsvision@diadoc.ru</code>.

Примеры

См. [пример реализации интерфейса](#) в разделе [undefined/programmer/dv5/Edi/import-config/\[Разработка компонента импорта настроек организации из системы оператора ЭДО\]](#).

ISignatureService — интерфейс

Интерфейс `ISignatureService` определяет методы, которые должен реализовывать компонент [undefined/programmer/dv5/Edi/signature-service/\[сервиса подписей\]](#).

- **Пространство имен:** `DocsVision.Edi.Interfaces`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public interface ISignatureService
```

Свойства

Имя	Описание
<code>CreateCommentSignature(String, X509Certificate2)</code>	Создаёт в Docsvision подпись для текстового сообщения. Метод должен вернуть идентификатор созданной подписи. Тип возвращаемого значения: <code>Guid</code>. <i>Параметры:</i> • <code>commentText</code> — подписываемое текстовое сообщение. • <code>certificate</code> — сертификат подписи. В стандартной конфигурации модуля метод вызывается, например, при подписании комментария к отказу в подписании полученного документа.

Имя	Описание
<code>CreateFileSignature(MessageFile, X509Certificate2)</code>	Создаёт в Docsvision подпись для файла сообщения. Метод должен вернуть идентификатор созданной подписи. Тип возвращаемого значения: <code>Guid</code>. <i>Параметры:</i> • <code>messageFile</code> — подписываемый файл. • <code>certificate</code> — сертификат подписи. В стандартной конфигурации модуля метод вызывается, например, при подписании файла сообщения с запросом уточнения для полученного документа.
<code>Initialize(ObjectContext)</code>	Инициализирует экземпляр компонента сервиса подписей. <i>Параметры:</i> • <code>objectContext</code> — контекст объектов.

LoaderService — класс

Сервис загрузки входящих и исходящих документов. Не рекомендуется использовать напрямую доступен, предпочтительнее вызывать методы через `undefined/programmer/dv5/Edi/api/BoxesService/[BoxesService]`.

Вызывает экземпляры `ILoaderService` для соответствующего оператора обмена.

- **Пространство имен:** `DocsVision.Edi.Runtime`
- **Сборка:** `DocsVision.Edi.Runtime.dll`

Синтаксис

```
public class LoaderService
```

Методы

Имя	Описание
<code>public ICollection<MessageData> LoadIncomingDocuments(string, DateTime, DateTime, string)</code>	Загружает входящие документы в указанном промежутке с ... по ..., по ID ящика и ID последнего документа
<code>public ICollection<MessageData> LoadOutgoingDocuments(string, DateTime, DateTime, string)</code>	Загружает исходящие документы в указанном промежутке с ... по ..., по ID ящика и ID последнего документа

OutgoingMessagesService — класс

Сервис отвечает за отправку исходящих документов, ответов и прочих задач. Может быть использован в собственных скриптах и приложениях для отправки документов и прочего.

- **Пространство имен:** `DocsVision.Edi.Runtime`
- **Сборка:** `DocsVision.Edi.Runtime.dll`

Синтаксис

```
public class OutgoingMessagesService
```

Методы

Имя	Описание
<code>public IEnumerable<EdiMessage> CreateOutgoingMessageCards(OperatorsBox, OperatorsDepartment, IEnumerable<RecipientFiles>, out string)</code>	Создаёт карточки исходящих сообщений <i>Параметры:</i> • <code>organizationBox</code> — ящик организации • <code>organizationDepartment</code> — подразделение организации • <code>recipientFiles</code> — коллекция файлов получателя • <code>sendErrors</code> — отправлять ошибки

Имя	Описание
<pre>public IEnumerable<EdiMessage> CreateOutgoingMessageCards(OperatorsBox, OperatorsDepartment, string, string, IEnumerable<RecipientFiles>, out string)</pre>	Создаёт карточки исходящего сообщения <i>Параметры:</i> • organizationBox — ящик организации • organizationDepartment — подразделение организации • documentType — тип документа • documentVersion — версия документа • recipientFiles — коллекция файлов получателя • sendErrors — отправлять ошибки
<pre>public IEnumerable<EdiMessage> CreateOutgoingRejectionReplies(Guid, string, Guid)</pre>	Создаёт исходящий ответный отказ в подписании <i>Параметры:</i> • cardId — ID карточки документа • rejectionText — текст комментария с отказом • rejectionSignatureId — ID отклонённой подписи
<pre>public IEnumerable<EdiMessage> CreateOutgoingRejectionReplies(Guid, string, X509Certificate2)</pre>	Создаёт исходящий ответный отказ в подписании <i>Параметры:</i> • cardId — ID карточки документа • rejectionText — текст комментария с отказом • certificate — сертификат

Имя	Описание
<pre>public IEnumerable<EdiMessage> CreateOutgoingSignatureReplies(Guid, ItemCollection<DocumentFileData>)</pre>	Создаёт исходящую ответную подпись <i>Параметры:</i> • cardId — ID карточки • signedFiles — подписанные файлы
<pre>public IEnumerable<EdiMessage> CreateOutgoingInvoiceReplies(Guid, X509Certificate2, string)</pre>	Создаёт ответные исходящие счёт-фактуры <i>Параметры:</i> • cardId — ID карточки документа • replyData — ответные данные • certificate — сертификат
<pre>public IEnumerable<EdiMessage> CreateRevocationRequests(IEnumerable<EdiMe ssage>, string, X509Certificate2, out string)</pre>	Создаёт исходящие запросы на аннулирование <i>Параметры:</i> • messageCards — карточки сообщений • revocationText — текст отказа • certificate — сертификат • sendErrors — отправлять ошибки
<pre>public IEnumerable<EdiMessage> CreateOutgoingRevRejectionReplies(Guid, string, X509Certificate2)</pre>	Создаёт исходящие ответы на аннулирование подписи <i>Параметры:</i> • cardId — ID карточки документа • rejectionText — текст комментария с отказом • certificate — сертификат
<pre>public IEnumerable<EdiMessage> CreateOutgoingRevSignatureReplies(Guid, X509Certificate2)</pre>	Создаёт исходящие ответы на аннулирование подписи <i>Параметры:</i> • cardId — ID карточки

Имя	Описание
<code>public IEnumerable<EdiMessage> CreateOutgoingReceipts(Guid, X509Certificate2)</code>	Отправляет исходящие извещения <i>Параметры:</i> • <code>cardId</code> — карточка • <code>certificate</code> — сертификат
<code>public IEnumerable<EdiMessage> CreateCorrectionRequests(IEnumerable<EdiMe ssage>, string, X509Certificate2, out string sendErrors)</code>	Отправляет исходящие запросы на исправление <i>Параметры:</i> • <code>messageCards</code> — карточки сообщений • <code>comment</code> — комментарий • <code>certificate</code> — сертификат • <code>sendErrors</code> — отправлять ошибки
<code>public IEnumerable<EdiMessage> FindSentMessageCards(Guid)</code>	Находит отправленные карточки сообщений по ID карточки
<code>public IEnumerable<EdiMessage> FindReceivedMessageCards(Guid)</code>	Находит полученные карточки сообщений по ID карточки
<code>public IEnumerable<EdiMessage> FindIncomingMessageCards(Guid)</code>	Находит входящие карточки сообщений по ID карточки
<code>public IEnumerable<EdiMessage> FindCorrectionMessageCards(Guid)</code>	Находит карточки сообщений для исправления по ID карточки
<code>public IEnumerable<EdiMessage> FindRevocationMessageCards(Guid)</code>	Находит карточки сообщений для аннулирования по ID карточки
<code>public IEnumerable<EdiMessage> FindReceiptMessageCards(Guid)</code>	Находит карточки сообщений, по которым требуется отправить извещения по ID карточки

FileContentType — перечисление

Определяет возможные типы содержимого в файле карточки *Документ*.

- **Пространство имен:** `DocsVision.Edi.Entities`
- **Сборка:** `DocsVision.Edi.Interfaces.dll`

Синтаксис

```
public enum FileContentType
```

Члены

Имя члена	Описание
<code>NonFormalized</code>	Неформализованный документ.
<code>SellerInvoice</code>	УПД продавца.

EdiMessageType — перечисление

Определяет возможные типы событий *журнала обмена* в *Карточки обмена сообщениями*.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public enum EdiMessageType
```

Члены

Имя члена	Описание
<code>Information</code>	Информация.
<code>Error</code>	Ошибка.
<code>Reply</code>	Ответ.
<code>FromDocsvision</code>	Из Docsvision.
<code>FromPartner</code>	От контрагента.
<code>FromOperator</code>	От оператора обмена.

EdiMessageFileState — перечисление

Определяет возможные типы содержимого в `MessageFile`.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`

- **Сборка:** DocsVision.Edi.ObjectModel.dll

Синтаксис

```
public enum EdiMessageFileState
```

Члены

Имя члена	Описание
Creating	Создаётся.
SignatureRequired	Получен на подпись от контрагента.
SentToPartner	Отправлен на подпись контрагенту.
PartnerRejected	Получен отказ от контрагента.
PartnerSigned	Получена подпись от контрагента.
Error	Ошибка.
ReplySignature	Ответная подпись.
ReplyRejection	Ответный отказ от подписи.
RevocationRequested	Запрошен отзыв.
RevocationSent	Отзыв отправлен.
Revoked	Отозвано.
Received	Получено.

EdiMessageFileContentType — перечисление

Определяет возможные типы содержимого в файле сообщения электронного обмена.

- **Пространство имен:** DocsVision.Edi.ObjectModel
- **Сборка:** DocsVision.Edi.ObjectModel.dll

Синтаксис

```
public enum EdiMessageFileContentType
```

Члены

Имя члена	Описание
<code>NonFormalized</code>	Неформализованный документ.
<code>SellerInvoice</code>	УПД продавца.

EdiMessageFileType — перечисление

Определяет возможные типы файлов в *Карточки обмена сообщениями*.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public enum EdiMessageFileType
```

Члены

Имя члена	Описание
<code>Main</code>	Основной.
<code>Additional</code>	Дополнительный.

EdiMessageReplyFileType — перечисление

Определяет возможные типы файлов в *Карточки обмена сообщениями*.

- **Пространство имен:** `DocsVision.Edi.ObjectModel`
- **Сборка:** `DocsVision.Edi.ObjectModel.dll`

Синтаксис

```
public enum EdiMessageReplyFileType
```

Члены

Имя члена	Описание
<code>InvoiceConfirmation</code>	Подтверждение получения УПД.

Имя члена	Описание
<code>InvoiceReceipt</code>	Извещение о получении УПД.
<code>InvoiceReply</code>	Ответный титул УПД.
<code>SignatureRejection</code>	Отказ от подписи.
<code>InvoiceCorrectionRequest</code>	Запрос на уточнение УПД.

Методы расширения для Web-клиента

Для работы с модулем в web-интерфейсе Web-клиент приносит расширения:

- Клиентское расширение, устанавливается в каталог `Docsvision\WebClient\Site\Content\Modules\EdiWebExtension\` и собирается в папку `\edi-module\WebClient\Edi.WebExtension\dist`
- Серверное расширение и сборки EDI устанавливаются в каталог `Docsvision\WebClient\Site\Extensions\DocsVision.Edi.ServerExtension\`.

Расширения `EdiWebServerExtension` и `EdiWebClientExtension` добавляют в систему следующие методы:

Скрипты	Операция	Используется в разметках	Когда использовать
<code>approveRevocation</code>	Подтвердить аннулирование	Исходящий УПД (просмотр), Входящий УПД (просмотр)	Для подтверждения аннулирования. При этом нужно будет выбрать сертификат квалифицированно й подписи для формирования ответной подписи. Для неформализованны х документов

Скрипты	Операция	Используется в разметках	Когда использовать
<code>rejectRevocation</code>	Отказать в аннулировании	Исходящий УПД (просмотр), Входящий УПД (просмотр)	Для отказа в аннулировании. При этом нужно будет указать причину отказа. Для неформализованных документов
<code>printInvoiceUniversalTransferDocument</code> Расширение из папки в модулях <code>CustomPrintInvoice</code> .	Печать	Исходящий УПД (просмотр), Входящий УПД (просмотр)	Выводит на печать печатную форму формализованного документа
<code>signAndSendOutgoing</code>	Подписать и отправить	Исходящий УПД (просмотр)	При нажатии будет вызвана функция подписания документа с отправкой контрагенту. Для неформализованных исходящих
<code>signIncoming</code>	Подписать	-	Для неформализованных входящих
<code>signAndSendInvoice</code>	Подписать	Входящий УПД (просмотр)	При нажатии будет вызвана функция подписания документа с отправкой контрагенту

Скрипты	Операция	Используется в разметках	Когда использовать
<code>rejectAndSendInvoice</code>	Отказать	Входящий УПД (просмотр)	При нажатии будет вызвана функция отказа в подписании с указанием причины отказа. Для неформализованных входящих
<code>receipt</code>	Отправить извещение	Входящий УПД (просмотр)	Отправляет извещение-подтверждение о получении оператора ЭДО
<code>revoke</code>	Аннулировать	Входящий УПД (просмотр)	При нажатии будет вызвана функция аннулирования документа вне зависимости от его типа

Описание схем карточек модуля

Модуль интеграции с операторами ЭДО

Описание схем карточек модуля

Описание схем системных карточек Модуля интеграции с операторами ЭДО.

- [Схема карточки "Карточка обмена сообщениями"](#)
- [Схема карточки "Расширение навигатора ЮЗДО"](#)
- [Схема карточки "Справочник настроек операторов ЮЗДО"](#)

Схема карточки "Карточка обмена сообщениями"

- Идентификатор: F2C0CE31-F629-49DF-9BCB-89BC07C60CA5
- Псевдоним: CardEdiMessage
- Название: Карточка обмена сообщениями

Секции карточки

Секция "Основная информация"

- Идентификатор: C03530CF-F7A1-4BE5-AD5E-8328FDB9D614
- Псевдоним: MainInfo
- Название: Основная информация
- Тип секции: struct

Таблица 3. Поля секции "Основная информация"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
CardId	Карточка	refcardid		

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
MessageState	Состояние	enum		Значения: Создается = 0, Получен на подпись от контрагента = 1, Отправлен на подпись контрагенту = 2, Получен отказ от контрагента = 3, Получена подпись от контрагента = 4, Ошибка = 5, Готово к отправке = 6, Отправлена ответная подпись контрагенту = 7, Требуется подтверждение получения = 8, Отправлено извещение о получении = 9, Требуется уточнение = 10, Получен запрос на аннулирование = 11, Отправлен запрос на аннулирование = 12,
322				Аннулирован = 13, Получен от

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Partner	Контрагент	refid		
OrganizationBox	Ящик организации	refid		
OrganizationDepartment	Подразделение организации	refid		
PartnerBox	Ящик контрагента	refid		
PartnerDepartment	Подразделение контрагента	refid		
OperatorMessageId	Идентификатор сообщения у оператора	unistring	1024	
DocumentType	Тип документа	unistring	1024	
DocumentVersion	Версия документа	unistring	1024	
IncomingMessage	Входящее сообщение	bool		

Секция "Файлы"

- Идентификатор: 7D729392-7A2A-47A8-8865-14521CB2B99D
- Псевдоним: Files
- Название: Файлы
- Тип секции: coll

Таблица 4. Поля секции "Файлы"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
FileId	Файл	uniqueid		
FileName	Имя файла	unistring	1024	

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
FileType	Тип файла	enum		Значения: Неформализованный основной = 0, Неформализованный дополнительный = 1
SignatureId	Подпись	uniqueid		
OperatorFileId	Идентификатор файла у оператора	unistring	1024	
OperatorSignatureId	Идентификатор подписи у оператора	unistring	1024	

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
FileState	Состояние файла	enum		Значения: Создается = 0, Получен на подпись от контрагента = 1, Отправлен на подпись контрагенту = 2, Получен отказ от контрагента = 3, Получена подпись от контрагента = 4, Ошибка = 5, Отправлена ответная подпись контрагенту = 6, Отправлен отказ от подписи контрагенту = 7, Получен запрос на аннулирование = 8, Отправлен запрос на аннулирование = 9, Аннулирован = 10, Получен от контрагента = 11

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
RejectionComment	Комментарий при отказе от подписи	refid		
ReplySignatureId	Ответная подпись	uniqueid		
ContentType	Тип содержимого	enum		Значения: Не формализованный = 0, УПД продавца = 1
RevRejectionFile	Комментарий при отказе от аннулирования	refid		
RevReplySignatureId	Ответная подпись об аннулировании	uniqueid		

Подчиненные секции

Секция "Ответные файлы"

- Идентификатор: 3BA8BF11-9A51-4537-9E91-A90F62C4288E
- Псевдоним: ReplyFiles
- Название: Ответные файлы
- Тип секции: coll

Таблица 5. Поля секции "Ответные файлы"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
ReplyFile	Ответный файл	refid		

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
ReplyFileType	Тип ответного файла	enum		Значения: Подтверждение получения = 0, Извещение о получении = 1, Ответный титул = 2, Отказ от подписи = 3, Запрос на уточнение = 4, Запрос на аннулирование = 5, Ответная подпись аннулирования = 6, Отказ от аннулирования = 7
NeedReceipt	Требуется извещение	bool		
ReplyReceipt	Ответная квитанция	refid		

Секция "Журнал обмена"

- Идентификатор: 7242BC7E-1C4A-4EE0-9B18-D6A922406376
- Псевдоним: Log
- Название: Журнал обмена
- Тип секции: coll

Таблица 6. Поля секции "Журнал обмена"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
EventType	Тип события	enum		Значения: Информация = 0, Ошибка = 1, Ответ = 2, Из Docsvision = 3, От контрагента = 4, От оператора обмена = 5
Date	Дата	datetime		
Description	Описание	unitext		
Comment	Комментарий	unitext		
FileName	Имя файла	unistring	1024	
SignedFile	Подписанный файл	refid		

Секция "Подписанные комментарии"

- Идентификатор: BFD0947B-ECCE-4C6C-B777-0AF149D9CDA8
- Псевдоним: SignedComments
- Название: Подписанные комментарии
- Тип секции: coll

Таблица 7. Поля секции "Подписанные комментарии"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Comment	Комментарий	unitext		
SignatureId	Подпись	uniqueid		

Секция "Подписанные файлы"

- Идентификатор: 4A8B3515-6E35-4FC7-89D4-9898021C074E
- Псевдоним: SignedFiles
- Название: Подписанные файлы
- Тип секции: coll

Таблица 8. Поля секции "Подписанные файлы"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
FileId	Файл	refcardid		Идентификатор типа: 6E39AD2B-E930-4D20-AAFA-C2ECF812C2B3 Идентификатор секции: 2FDE03C2-FF87-4E42-A8C2-7CED181977FB
FileName	Имя файла	unistring	1024	
SignatureId	Подпись	uniqueid		
OperatorFileId	Идентификатор файла у оператора	unistring	1024	
OperatorSignatureId	Идентификатор подписи у оператора	unistring	1024	
SourceFileId	Исходный файл	uniqueid		

Секция "Данные для переноса"

- Идентификатор: 07119738-4108-4766-AF63-F80E04797FA4

- Псевдоним: TransferData
- Название: Данные для переноса
- Тип секции: coll

Таблица 9. Поля секции "Данные для переноса"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Date	Дата	datetime		
NewStateId	Новое состояние карточки	uniqueid		

Подчиненные секции

Секция "Файлы для переноса"

- Идентификатор: C5ED1494-BF06-4E4C-825F-CDAB99E3B502
- Псевдоним: TransferFiles
- Название: Файлы для переноса
- Тип секции: coll

Таблица 10. Поля секции "Файлы для переноса"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
SignedFile	Подписанный файл	refid		

Схема карточки "Расширение навигатора ЮЗДО"

- Идентификатор: 17136E0A-B366-462B-82B8-E323309FDA42
- Псевдоним: EdiNavExtension
- Название: Расширение навигатора ЮЗДО

Секции карточки

Секция "Настройки"

- Идентификатор: 30566216-D19A-4ED3-9767-5679FDC08377
- Псевдоним: Settings
- Название: Настройки
- Тип секции: struct

Таблица 11. Поля секции "Настройки"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
EmptyField	Пустое поле	int		

Схема карточки "Справочник настроек операторов ЮЗДО"

- Идентификатор: 32E379B1-000C-4587-B503-8A113CB45DA1
- Псевдоним: RefOperators
- Название: Справочник настроек операторов ЮЗДО

Секции карточки

Секция "Операторы"

- Идентификатор: CB39C02E-8664-4729-A95D-145C2C0C5FB1
- Псевдоним: Operators
- Название: Операторы
- Тип секции: coll

Таблица 12. Поля секции "Операторы"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Name	Название	unistring	256	

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
ServiceClassName	Имя класса сервиса сообщений	unistring	1024	
SignaturesClassName	Имя класса для работы с подписями	unistring	1024	
ImportClassName	Имя класса импорта	unistring	1024	

Подчиненные секции

Секция "Организации"

- Идентификатор: 9586B49F-94DC-4288-9D9F-5ED7DADC698D
- Псевдоним: Units
- Название: Организации
- Тип секции: coll

Таблица 13. Поля секции "Организации"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Name	Название	unistring	1024	
UnitId	Идентификатор организации	unistring	1024	
INN	ИНН	unistring	1024	
Unit	Организация	refid		
Recipient	Получатель	refid		
RecipientGroup	Группа получателей	refid		

Подчиненные секции

Секция "Ящики организаций"

- Идентификатор: 70DE302E-2FCA-4562-9FDB-083B9A14B135
- Псевдоним: Boxes
- Название: Ящики организаций
- Тип секции: coll

Таблица 14. Поля секции "Ящики организаций"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Name	Название	unistring	1024	
BoxId	Идентификатор ящика	unistring	1024	
Activated	Активирован	bool		
IsDefault	Использовать по умолчанию	bool		

Подчиненные секции

Секция "События ящика"

- Идентификатор: 25301DB3-9EB4-4B5D-A156-43D9BDF15625
- Псевдоним: BoxEvents
- Название: События ящика
- Тип секции: struct

Таблица 15. Поля секции "События ящика"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
LastEventId	Последнее обработанное событие	unistring	1024	

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
ProcessDate	Дата обработки событий	datetime		

Секция "Подразделения"

- Идентификатор: 3E9F0511-649F-4A8F-B1D6-462451AE2F00
- Псевдоним: Departments
- Название: Подразделения
- Тип секции: tree

Таблица 16. Поля секции "Подразделения"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Name	Название	unistring	1024	
DepartmentId	Идентификатор подразделения	unistring	1024	
KPP	КПП	unistring	1024	
Department	Подразделение	refid		
Recipient	Получатель	refid		
RecipientGroup	Группа получателей	refid		

Секция "Контрагенты"

- Идентификатор: 3CDD614B-3D89-4579-8C6B-02C51E9FAB77
- Псевдоним: Partners
- Название: Контрагенты
- Тип секции: coll

Таблица 17. Поля секции "Контрагенты"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Name	Название	unistring	1024	
PartnerId	Идентификатор контрагента	unistring	1024	
INN	ИНН	unistring	1024	
Partner	Контрагент	refid		
Contract	Контракт	bool		

Подчиненные секции

Секция "Ящики контрагента"

- Идентификатор: F439CB17-EDDB-407D-9161-99613048C60E
- Псевдоним: PartnerBoxes
- Название: Ящики контрагента
- Тип секции: coll

Таблица 18. Поля секции "Ящики контрагента"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Name	Название	unistring	2048	
BoxId	Идентификатор	unistring	1024	
IsDefault	Использовать по умолчанию	bool		

Секция "Связанные ящики организаций"

- Идентификатор: 8D527958-738D-4F29-8023-F0AC06127966
- Псевдоним: LinkedBoxes
- Название: Связанные ящики организаций
- Тип секции: coll

Таблица 19. Поля секции "Связанные ящики организаций"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
LinkedException	Связанный ящик	refid		Поля ссылки: LinkedException_Name > Name

Секция "Подразделения контрагента"

- Идентификатор: BBEA60A0-5803-421C-B82E-2A28480D3AEB
- Псевдоним: PartnerDepartments
- Название: Подразделения контрагента
- Тип секции: tree

Таблица 20. Поля секции "Подразделения контрагента"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Name	Название	unistring	1024	
DepartmentId	Идентификатор подразделения	unistring	1024	
KPP	КПП	unistring	1024	
PartnerDepartment	Подразделение контрагента	refid		

Секция "Настройки"

- Идентификатор: 5A18EF80-9384-4BFF-B57A-29F0239C982B
- Псевдоним: Settings
- Название: Настройки
- Тип секции: coll

Таблица 21. Поля секции "Настройки"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
Name	Название	unistring	256	
Value	Значение	unitext		
Encrypted	Зашифровано	bool		

Секция "Типы документов"

- Идентификатор: BF97F2FE-E13A-42BE-B4A1-F8E06FCD93B1
- Псевдоним: DocumentTypes
- Название: Типы документов
- Тип секции: coll

Таблица 22. Поля секции "Типы документов"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
TypeId	Тип	uniqueid		
ReaderClassName	Имя класса чтения данных	unistring	1024	
UpdaterClassName	Имя класса изменения данных	unistring	1024	

Секция "Типы полученных документов"

- Идентификатор: 88D45B13-E836-41E1-BDBD-AFFD2436EFD4
- Псевдоним: ReceivedTypes
- Название: Типы полученных документов
- Тип секции: coll

Таблица 23. Поля секции "Типы полученных документов"

Псевдоним	Название	Тип поля	Размер	Дополнительная информация
TypeName	Тип полученного документа	unistring	1024	
Version	Версия полученного документа	unistring	1024	
KindId	Вид карточки	uniqueid		
CreatorClassName	Имя класса для создания карточек	unistring	1024	

Администрирование коннектора к Диадок

Общие сведения о коннекторе

Назначение и функции Коннектора

Коннектор к *Диадок* является дополнительным компонентом *Модуля интеграции с операторами ЭДО*, который добавляет поддержку работы с оператором *Диадок*.

Коннектор предоставляет следующие возможности:

- Отправка и получение электронных документов через оператора *Диадок*.
- Получение списка организаций и контрагентов, зарегистрированных у оператора ЭДО *Диадок*.

Необходимые ресурсы

Требование	Описание требования
Необходимое техническое обеспечение	Специальные требования к аппаратному обеспечению не предъявляются.
Необходимое программное обеспечение	Серверная и клиентская части <i>Модуля интеграции с операторами ЭДО</i> версии 5.5.4 и выше.
Требования к каналам связи	Администратор локальной сети должен обеспечить возможность подключения с сервера Docsvision и серверов СУБП к сервису оператора <i>Диадок</i> . • Стандартная точка подключения: https://diadoc-api.kontur.ru. • Адрес тестового сервера <i>Диадок</i>: https://diadoc-api-staging.kontur.ru
Лицензия	Для работы с Коннектором лицензия на систему Docsvision должна содержать дополнительную опцию <i>Docsvision Коннектор к Диадок</i> .

Требование	Описание требования
Учётная запись в системе Диадок	Для передачи электронных документов через оператора ЭДО "Диадок" необходимо быть зарегистрированным пользователем данной системы. Регистрация организации в системе "Диадок" выполняется самостоятельно. Для загрузки контрагентов требуется подключение к ЭДО "Диадок" с тарифом, включающим оплату доступа к API. В противном случае при загрузке контрагентов в журнале сервера будет появляться ошибка "Необходима оплата тарифа АПИ". Если учетная запись используется в нескольких независимых организациях с разными тарифами, конкретно для коннектора можно сделать отдельную учетную запись и добавить её только в организации с доступом к API.  Информация о подключении к оператору Диадок приведена на сайте https://www.diadoc.ru/.

Установка и удаление Коннектора

Состав установочного комплекта

1. Пакет установки серверной части Коннектора: **DiadocConnectorServer.msi**.
2. Пакет установки клиентской части Коннектора: **DiadocConnectorClient.msi**.

Установка серверной части Коннектора

Серверная часть *Коннектора к Диадок* устанавливается на компьютеры с установленной серверной частью *Модуля интеграции с операторами ЭДО*.

Установка должна выполняться от имени пользователя, являющегося локальным администратором и администратором Docsvision.



Администратор Docsvision — пользователь, включенный в группу безопасности "Docsvision Administrators" на компьютере с сервером Docsvision.

- 1. Запустите пакет установки **DiadocConnectorServer.msi**.

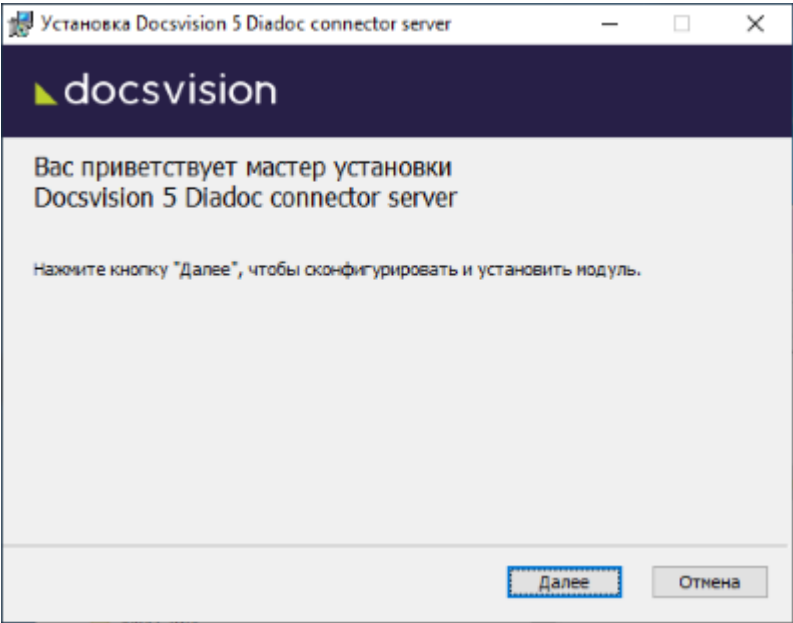


Рисунок 125. Мастер установки серверной части Коннектора

- 2. Примите условия лицензионного соглашения для продолжения установки.

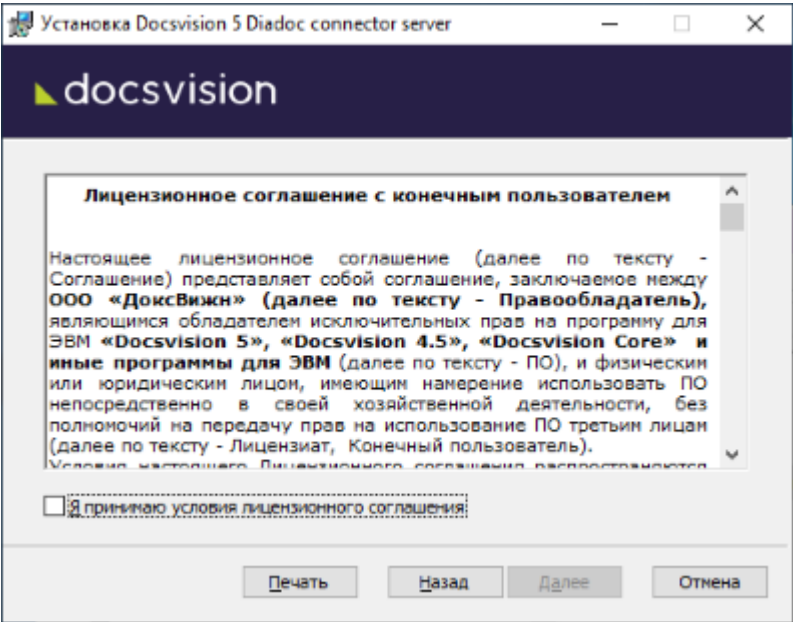


Рисунок 126. Условия лицензионного соглашения

3. В следующем окне нажмите **Установить** и дождитесь завершения установки Коннектора.
4. Нажмите на кнопку **Готово**.
5. Если коннектор устанавливается последним, запустите Консоль настройки Docsvision и выполните настройку системы следуя инструкции, данной в документации модуля Платформа `undefined/platform/5.5.5/admin/config-master/[здесь]` и `undefined/platform/5.5.5/admin/post-config-server/[здесь]`.

Установка клиентской части Коннектора

Клиентская часть *Коннектора к Диадок* устанавливается на компьютеры с установленной клиентской частью *Модуля интеграции с операторами ЭДО*. Если *Windows-клиент* и клиентская часть *Модуля интеграции с операторами ЭДО* установлены для всех пользователей, клиентский компонент *Коннектора* также должен быть установлен для всех пользователей.

1. Запустите мастер установки:

- `DiadocConnectorClient.msi`, чтобы установить *Коннектор* с областью установки *Для текущего пользователя*.
- `msiexec /i "DiadocConnectorClient.msi" ALLUSERS=1`, чтобы установить *Коннектор* с областью установки *Для всех пользователей*. Для данного варианта установки требуются права администратора.

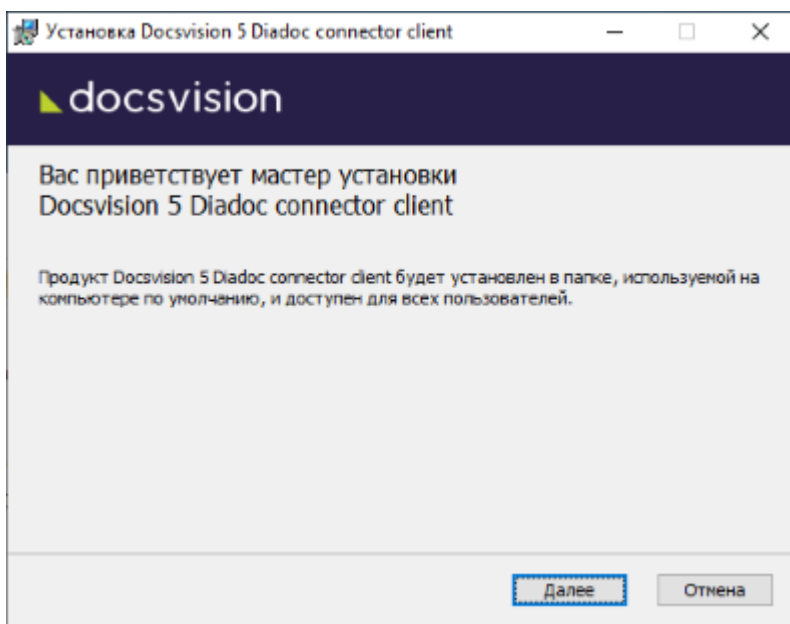
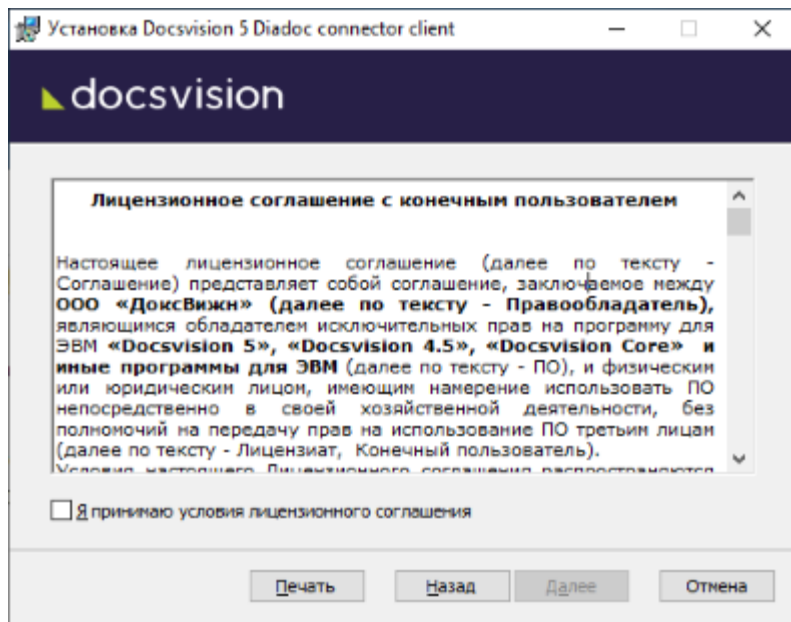


Рисунок 127. Мастер установки клиентской части Коннектора к Диадок

2. Примите условия лицензионного соглашения для продолжения установки.



3. В следующем окне нажмите **Установить** и дождитесь завершения установки Коннектора.
4. Нажмите на кнопку **Готово**.

Удаление Коннектора

1. Удалите настройки оператора Диадок из *Справочника настроек операторов ЮЗДО*.
 - а. Откройте карточку *Справочник настроек операторов ЮЗДО* в утилите *Docsvision Explorer* (входит в *Resource Kit*). Инструкция по работе с утилитой *Docsvision Explorer* приведена в документации *Resource Kit*, разделе с описанием утилиты "DVExplorer".
 - б. Удалите из секции *Операторы* строку оператора "Диадок".
2. Удалите компоненты модуля:
 - Клиентский компонент модуля: "Коннектор к Диадок. Клиент".
 - Серверный компонент модуля: "Коннектор к Диадок. Сервер".

Функции администратора

Настройка подключения к оператору Диадок

Администратор должен настроить подключение к оператору ЭДО "Диадок" в *Справочнике настроек операторов ЮЗДО Docsvision*, а также загрузить справочную информацию из системы Диадок.

1. Запустите Windows-клиент от имени администратора Docsvision.

Если Windows-клиент был открыт при установке Коннектора — перезапустите.

2. Откройте *Справочник настроек операторов ЮЗДО*.

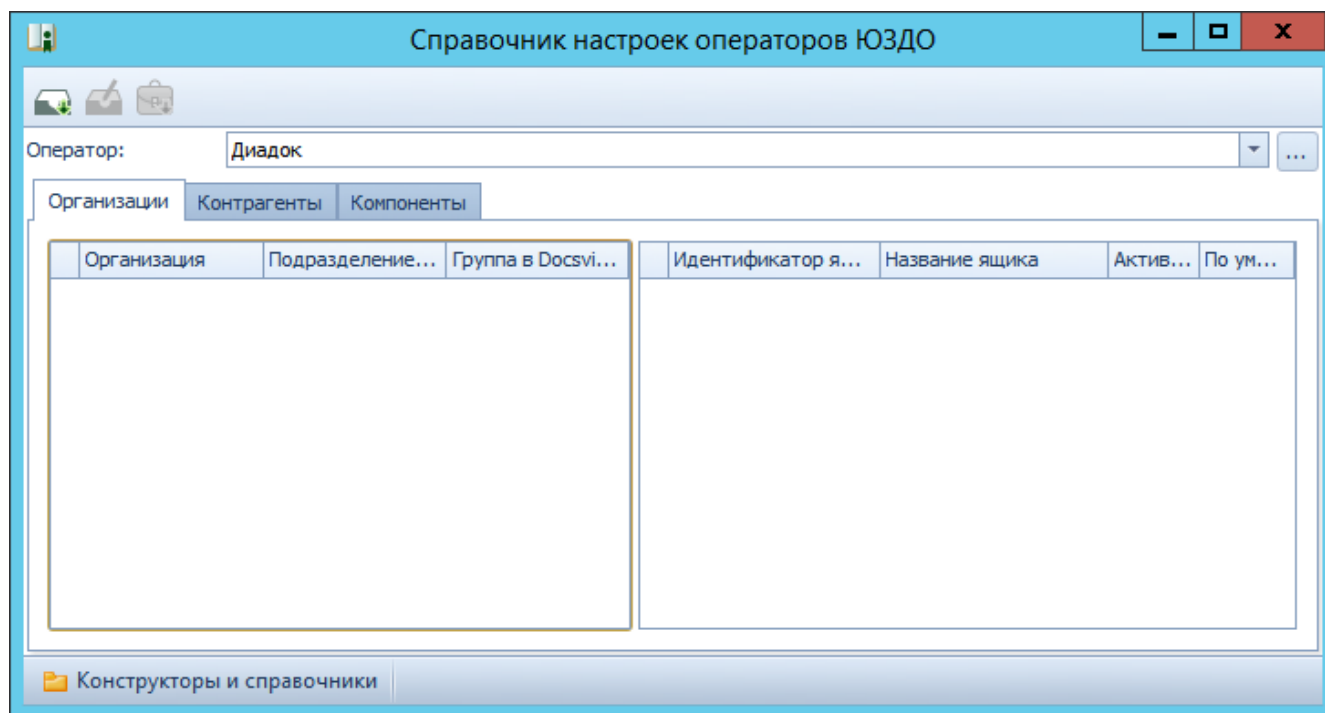


Рисунок 128. Справочник настроек операторов ЮЗДО

3. В раскрывающемся списке **Оператор** выберите элемент "Диадок".

4. Нажмите кнопку  в строке *Оператор*. Будет открыто окно настройки подключения к оператору ЭДО.

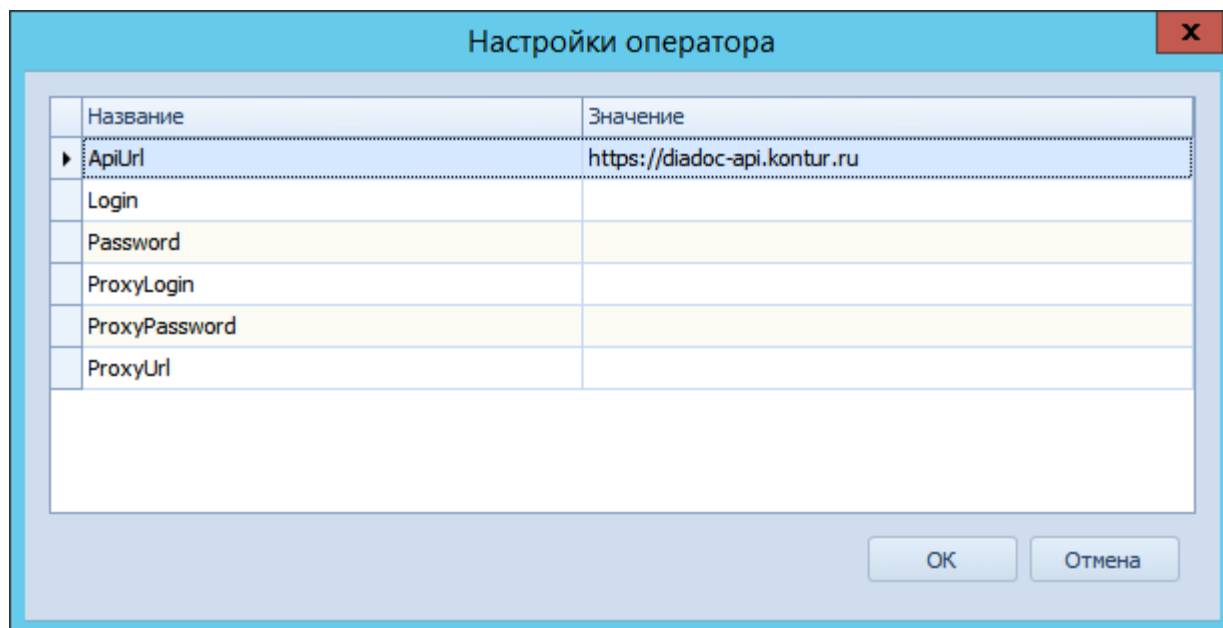


Рисунок 129. Параметры подключения к оператору ЭДО

5. Укажите данные для подключения к системе Диадок:

- *Login* — имя учетной записи в Диадок.
- *Password* — пароль.

Учетная запись должна обладать доступом **ко всем документам организации** в системе Диадок.

Следующие параметры настраиваются, если подключение к оператору ЭДО "Диадок" будет осуществляться через прокси-сервер:

- *ProxyUrl* — адрес прокси-сервера.
- *ProxyLogin* / *ProxyPassword* — учетная запись и пароль для подключения к прокси-серверу.

Для анонимного подключения к прокси-серверу *ProxyLogin* / *ProxyPassword* оставьте пустыми.

Точку подключения к API Диадок — *ApiUrl* — изменять не требуется.

6. Нажмите **ОК** для сохранения настроек.

7. В основном окне *Справочника настроек операторов ЮЗДО*:

- а. Настройте справочник организаций.

б. Настройте справочник контрагентов.

Инструкция по выполнению данных действий приведена в руководстве по администрированию модуля *Модуль интеграции с операторами ЭДО* в разделе "[Настройка справочника организаций](#)".