
diadoc1cpro-docs Documentation

Выпуск 1

Diadoc

мар. 17, 2020

1	Работа подключаемого модуля	3
1.1	Краткое описание	3
1.1.1	События подключаемого модуля	3
1.1.2	Возможности подключаемого модуля	4
1.1.3	Расположение подключаемого модуля	4
1.2	Примеры использования подключаемого модуля	4
1.2.1	Как подготовить пакет документов для отправки	4
1.2.1.1	Названия пакетов	4
1.2.1.2	Форматы документов	5
1.2.1.3	Заполнение списка пакетов на отправку	5
1.2.1.4	Состав пакетов	6
1.2.1.5	Формирование электронного документа	7
1.2.1.5.1	Формализованный	7
1.2.1.5.2	Не формализованные документы	9
1.2.1.6	Отправка пакета-шаблона	10
1.2.2	Дополнительная обработка исходящих пакетов	10
1.2.3	Как обработать входящий пакет документов	11
1.2.3.1	Сопоставление документов	11
1.2.3.2	Правила ручного сопоставления документа	12
1.2.3.3	Создание документов	13
1.2.3.4	Поиск документов, созданных на основании шаблона	13
1.2.3.5	Сопоставление номенклатуры	14
1.2.3.6	Анализ документов	15
1.2.4	Как доработать таблицу списка документов	16
1.2.4.1	Дополнительные колонки	17
1.2.4.2	Доработка формы основного модуля	18
1.2.5	Как добавить кнопки в действия	20
1.2.6	Регламентное задание	20
1.2.7	Дополнительные возможности	23
1.2.7.1	После обработки ленты событий	23
1.2.7.2	При изменении статуса документа	24
1.2.7.3	Переопределение типов Организации и Контрагентов	24
1.2.7.4	Фильтр контрагентов для приглашения	25
2	Функции подключаемого модуля	27
2.1	Исходящие документы	27

2.1.1	ПодготовитьПакет	27
2.1.2	ПодготовитьЭлектронныйДокумент	27
2.1.3	ПолучитьТаблицуИспользуемыхВидовДокументов	28
2.1.4	ПолучитьТаблицуИспользуемыхПакетов	28
2.1.5	Получить текст запроса для списка пакетов на отправку по массиву видов пакетов	29
2.1.6	ПослеОтправкиПакета	30
2.1.7	ПослеПодготовкиПакета	30
2.2	Входящие документы	30
2.2.1	ВыполнитьТребуемоеДействие	30
2.2.2	НайтиСопоставлениеДокумента	31
2.2.3	ПолучитьЗаполненныйОбъектДокумента1С	31
2.2.4	ПослеАнализаПакета	32
2.2.5	ПравилаСопоставленияДокумента	32
2.2.6	ПровестиАнализДокумента	32
2.2.7	СопоставитьНоменклатуруПоставщика	33
2.3	Дополнение формы модуля	34
2.3.1	ЗаполнитьСписокДополнительныхКнопок	34
2.3.2	НажатиеДополнительнойКнопки	34
2.3.3	Получить настройки дополнительных колонок списка документов	35
2.3.4	ПослеОбновленияСпискаДокументов	35
2.3.5	ПриОткрытииФормы	36
2.3.6	ПриПолученииДанныхТабличнойЧасти	36
2.3.7	УстановитьУсловноеОформление	37
2.4	Прочие функции	37
2.4.1	ВыполнитьРегламентныеДействия	37
2.4.2	ОбработатьСобытие	37
2.4.3	ОпределитьТипЗначенияОбъекта	38
2.4.4	ПолучитьКоличествоДокументовЗаПериод	38
2.4.5	ПослеОбработкиЛентыСобытий	39
2.4.6	ПослеУстановкиСтатусаДокумента	39
3	Основные структуры	41
3.1	Структура «Пакет»	41
3.1.1	Пакет (начиная с версии 4.0)	41
3.1.2	Пакет (до версии 4.0)	43
3.2	Структура «Документ»	44
4	Вспомогательные структуры	49
4.1	Типы документов	49
4.2	Типы контента	50
5	Структуры для заполнения контента УПД в формате приказа ММВ-7-15/820@	51
5.1	Титул продавца Utd820SellerContent	51
5.2	Титул покупателя Utd820BuyerContent	52
5.3	AdditionalInfoId	53
5.4	AddressInfo	53
5.5	BuyerInfoCircumPublicProc	54
5.6	ContentOperCode	55
5.7	CustomDeclaration	55
5.8	DocumentShipment	55
5.9	Employee	56
5.10	ExtendedOrganizationInfo	56
5.11	ExtendedSigner	57
5.12	ExtendedSignerDetails	57

5.13	InfoFundsLiab	58
5.14	Item	59
5.15	ItemTracingInfo	60
5.16	ItemIdentificationNumber	60
5.17	OtherIssuer	60
5.18	PaymentDocumentInfo	61
5.19	SellerInfoCircumPublicProc	61
5.20	Shipper	62
5.21	StructedAdditionalInfo	62
5.22	Table	62
5.23	TransferBase	63
5.24	TransferInfo	63
5.25	Waybill	64

Модуль предназначен для отправки, получения и просмотра электронных документов, передаваемых через систему электронного документооборота Диадок.

Модуль Диадок позволяет компаниям мгновенно отправлять документы своим клиентам, поставщикам, партнерам, удаленным подразделениям прямо из 1С. Это могут быть как документы в формате ФНС, так и произвольные файлы.

Документы в формате ФНС в сервисе подписываются электронной подписью, и обладает юридической силой. Подписание прочих документов настраивается в соответствии с бизнес-процессами компании.

Работа подключаемого модуля

Подключаемый модуль (ПМ) - это вспомогательная внешняя обработка, содержащая функции для обработки событий, зависящих от конфигурации 1С и бизнес-процессов организации.

1.1 Краткое описание

1.1.1 События подключаемого модуля

К событиям подключаемого модуля относятся действия, которые:

- собирают данные из конфигурации 1С для их отправки на сервер Диадока
- обрабатывают полученные с сервера Диадок данные для их регистрации в базе 1С. Например, создание документа, регистрирующего поступление товаров на основании входящего УПД.

Точкой входа из основного модуля в подключаемый модуль является функция *ОбработатьСобытие*.

Обращение к необходимой функции происходит через условие «Если»

```
Если ИмяСобытия = "ПолучитьТекстЗапросаДляСпискаПакетовНаОтправкуПоМассивуВидовПакетов" Тогда
```

```
    Возврат ПолучитьТекстЗапросаДляСпискаПакетовНаОтправкуПоМассивуВидовПакетов(Параметры,  
    ↪МассивВидовПакетов);
```

Вызов в основном модуле происходит с использованием процедуры ЭДО_ПодключаемыйМодуль_ОбработатьСобытие.

```
Параметры = Новый Структура("ВидПакетаРазвернутый,СтрокаСписка,Пакет", ВидПакетаРазвернутый,  
    ↪СтрокаСписка, Пакет);  
ЭДО_ПодключаемыйМодуль_ОбработатьСобытие("ПодготовитьПакет", Параметры);
```

1.1.2 Возможности подключаемого модуля

Для типовых конфигураций в модуле описаны основные действия для работы с событиями пакетов документов:

- сбор документов 1С в один вид пакета
- заполнение контента на основании данных из 1С для отправки пакета
- логика сопоставления и создания документов в 1С

Кроме того, в ПМ можно описать функции для прочих действий, в том числе связанных со списком документов в основной форме:

- логика необходимого анализа входящих документов согласно бизнес-процессам организации
- действия после обновления списка документов
- выполнение регламентного задания
- сопоставление номенклатуры
- действия дополнительных кнопок и т.д.

1.1.3 Расположение подключаемого модуля

Типовые ПМ для основных конфигураций 1С располагаются в макетах основного модуля.

В настройках основной обработки модуля Диадок есть возможность указать путь к доработанному ПМ, что упрощает настройку системы под бизнес-процессы каждого клиента.

1.2 Примеры использования подключаемого модуля

1.2.1 Как подготовить пакет документов для отправки

Модуль «Диадок Про» умеет отправлять документы контрагентам. По одной сделке часто требуется отправить не один документ, а минимум два.

Поэтому документы объединяются в *Пакет*. Состоять такой пакета может из нескольких доступных *типов документов*.

Аналогично веб-версии, одновременно нельзя отправить больше 30 документов.

Предварительно следует пакет назвать, определить состав и логику сбора документов. Каждый этап подготовки описывается в событии подключаемого модуля.

1.2.1.1 Названия пакетов

Документы, которые будут отправлены вместе, отображаются в списке для отправки в одну строку.

Стоит назвать каждую группу, чтобы пользователю понимать состав пакета. Например, согласно бизнес-логике («Возмещение») или типу документа («УПД»).

Наименование пакета, которое будет отображаться в интерфейсе, определяется в функции *ПолучитьТаблицуИспользуемыхПакетов*.

Пример определения пакетов «УПД» и «Возмещение»:

```
ТЗ = Новый ТаблицаЗначений;
ОсновнойМодуль.ЭДО_Службные_ДобавитьКолонкиВТаблицуЗначений(ТЗ, "ID, Наименование, УПД");
```

```
ОсновнойМодуль.ЭДО_Службные_ДобавитьСтрокуВТаблицуЗначений(ТЗ, "ID_УПД", "УПД", Истина);
ОсновнойМодуль.ЭДО_Службные_ДобавитьСтрокуВТаблицуЗначений(ТЗ, "ID_Возмещение", "Возмещение");
```

1.2.1.2 Форматы документов

При формировании документа важно знать какие сведения документа передавать, учитывать ограничения, накладываемые ФНС, и т.п.

Для объединения похожих по структуре документов используются такие характеристики: *тип контента*, функция и *тип документа API*.

Определение комбинаций этих характеристик («видов документов»), которые будут использованы в пакетах на отправку, происходит в функции *ПолучитьТаблицуИспользуемыхВидовДокументов*

Пример определения УПД и Счета на оплату:

```
ТЗ = Новый ТаблицаЗначений;
ОсновнойМодуль.ЭДО_Службные_ДобавитьКолонкиВТаблицуЗначений(ТЗ, "ID, Наименование, ↵
↵ТипДокументаAPI, ТипКонтентаAPI, ФункцияДокументаAPI");
```

```
ОсновнойМодуль.ЭДО_Службные_ДобавитьСтрокуВТаблицуЗначений(ТЗ, "ID_УПД", "УПД",
↵"UniversalTransferDocument", "utd", "СЧФДОП");
ОсновнойМодуль.ЭДО_Службные_ДобавитьСтрокуВТаблицуЗначений(ТЗ, "ID_Счет", "Счет",
↵"ProformaInvoice", "");
```

1.2.1.3 Заполнение списка пакетов на отправку

Построение запроса для заполнения списка пакетов на отправку происходит в функции *ПолучитьТекстЗапросаДляСпискаПакетовНаОтправкуПоМассивуВидовПакетов*.

Логика заполнения пакета предусматривает наличие одного главного документа, без которого отправлять контрагенту остальные не имеет смысла.

Сколько таких документов будет выбрано в запросе, столько пакетов и будет сформировано на отправку.

Пример выбора счетов-фактур из 1С для формирования вида пакета «УПД»:

Если ВидПакетаРазвернутый.ID = "ID_УПД" Тогда

Результат = ДобавитьОбъединениеВТекстЗапроса(Результат) +

"ВЫБРАТЬ

| СчетФактураВыданный.Ссылка КАК Документ,
 | СчетФактураВыданный.Номер КАК НомерДокумента,
 | СчетФактураВыданный.Дата КАК ДатаДокумента,
 | СчетФактураВыданный.Контрагент КАК Контрагент,
 | СчетФактураВыданный.Организация КАК Организация,
 | СчетФактураВыданный.СуммаДокумента КАК СуммаДокумента,
 | ""ID_УПД"" КАК ВидПакетаID
 |ИЗ

| Документ.СчетФактураВыданный КАК СчетФактураВыданный
 | ВНУТРЕННЕЕ СОЕДИНЕНИЕ ВТ_КэшКонтрагентовПоВидамУПД КАК ВТ_КэшКонтрагентовПоВидамУПД

(continues on next page)

(продолжение с предыдущей страницы)

```

|   ПО СчетФактураВыданный.Организация = ВТ_КэшКонтрагентовПоВидамУПД.Организация1С
|   И СчетФактураВыданный.Контрагент = ВТ_КэшКонтрагентовПоВидамУПД.Контрагент1С
|   ЛЕВОЕ СОЕДИНЕНИЕ Документ.ПакетЭД.ЭлектронныеДокументы КАК ОтправленныеДокументы
|   ПО ТипЗначения(ОтправленныеДокументы.ОбъектВладелец) = Тип(Документ.СчетФактураВыданный)
|   И СчетФактураВыданный.Ссылка = ОтправленныеДокументы.ОбъектВладелец
|   И (ОтправленныеДокументы.Ссылка.ВнешнийУИД = ""ID_УПД"")
|ГДЕ
|   ОтправленныеДокументы.Ссылка ЕСТЬ NULL
|   И СчетФактураВыданный.Дата МЕЖДУ &НачалоПериода И &КонецПериода
|   И СчетФактураВыданный.Проведен
|   И СчетФактураВыданный.ВидСчетаФактуры = ЗНАЧЕНИЕ(Перечисление.ВидСчетаФактурыВыставленного.
↪НаРеализацию)
|   И СчетФактураВыданный.ДоговорКонтрагента.ВидДоговора = ЗНАЧЕНИЕ(Перечисление.
↪ВидыДоговоровКонтрагентов.СПокупателем)
|   И НЕ СчетФактураВыданный.Исправление
|   И ВТ_КэшКонтрагентовПоВидамУПД.ИспользоватьУПД_МеткаОсновногоПакета";
КонецЕсли;

```

1.2.1.4 Состав пакетов

Наполнение документами происходит в функции *ПодготовитьПакет*.

Для каждого вида пакета вычисляются ссылки на документы 1С, которые стоит подготовить и добавить в электронном виде.

Это может быть только главный документ, либо связанные с ним документы и т.п.

Добавление документа в пакет происходит с помощью метода основного модуля «ЭДО_ДокументМенеджер_ПодготовитьИДобавитьДокументВПакет».

В качестве параметров передаются:

- ссылка на пакет, в который добавляется документ
- ссылка на документ 1С, по сведениям которого будет сформирован электронный документ
- внутренний идентификатор вида документов, в формате которого будет сформирован электронный документ
- дополнительный параметр, который будет передан в функцию по формированию электронного документа (необязательный)

Пользователю видны подготовленные документы в форме пакета на отправку.

Пример определения состава пакетов вида «УПД» (формируется из одного документа Счет-фактуры) и «Возмещение» (зависит от реквизитов списка):

```

Если ВидПакетаРазвернутый.ID = "ID_УПД" Тогда
    ОсновнойМодуль.ЭДО_ДокументМенеджер_ПодготовитьИДобавитьДокументВПакет(Пакет, СтрокаСписка.
↪Документ, "ID_УПД");
ИначеЕсли ВидПакетаРазвернутый.ID = "ID_Возмещение" Тогда
    ОсновнойМодуль.ЭДО_ДокументМенеджер_ПодготовитьИДобавитьДокументВПакет(Пакет, СтрокаСписка.
↪Документ, "ID_ОтчетИсполнителя");
    ОсновнойМодуль.ЭДО_ДокументМенеджер_ПодготовитьИДобавитьДокументВПакет(Пакет, СтрокаСписка.
↪ДопРеквизит1, "ID_Счет");

```

(continues on next page)

(продолжение с предыдущей страницы)

```

Если ЗначениеЗаполнено (СтрокаСписка.ДопРеквизит2) Тогда
    ОсновнойМодуль.ЭДО_ДокументМенеджер_ПодготовитьИДобавитьДокументВПакет (Пакет, СтрокаСписка.
↪ДопРеквизит2, "ID_СчетФактураВозмещение");
    КонецЕсли;
КонецЕсли;

```

1.2.1.5 Формирование электронного документа

Электронный документ состоит из метаданных и контента, заполнение которых происходит в функции *ПодготовитьЭлектронныйДокумент*.

Описание требуется для каждого вида документа или типа контента, которые используются при отправке.

1.2.1.5.1 Формализованный

Сбор контента для формализованных документов проще всего сделать с помощью метода основного модуля «ТиповойМодуль_ПолучитьКонтент».

В параметрах передается ссылка на документ 1С и тип контента, который надо собрать. Если это УПД или УКД, то в дополнительных параметрах стоит указать функцию.

Полученную структуру переводим в объект XDTO с помощью метода основного модуля «ЗаполнитьКонтентXDTOПоСтруктуре».

Пример типового сбора контента для документов типа «УПД»:

```

Результат = Параметры.Результат;
ВидДокументаРазвернутый = Параметры.ВидДокументаРазвернутый;
ДополнительныеПараметры = Параметры.ДополнительныеПараметры;

Документ1С = Результат.Документ1С;
ВидДокументаНаименование = ВидДокументаРазвернутый.Наименование;

ID = ВидДокументаРазвернутый.ID;
ТипКонтента_XDTO = ВидДокументаРазвернутый.ТипКонтента;
TypeNamedId = ВидДокументаРазвернутый.TypeNamedId;
FormatVersion = ВидДокументаРазвернутый.FormatVersion;
FunctionName = ВидДокументаРазвернутый.FunctionName;

Если ВРЕГ(ТипКонтента_XDTO) = ВРЕГ("UtdSellerContent") ИЛИ ВРЕГ(ТипКонтента_XDTO) = ВРЕГ(
↪"UcdSellerContent") Тогда

    //Получим функцию документа, которая используется в типовом модуле
    Если TypeNamedId = "Invoice" ИЛИ TypeNamedId = "InvoiceRevision" Тогда
        функцияУПД = "СЧФ";
    ИначеЕсли TypeNamedId = "XmlTorg12" ИЛИ TypeNamedId = "XmlAcceptanceCertificate" Тогда
        функцияУПД = "ДОП";
    Иначе
        функцияУПД = ОсновнойМодуль.ФункцияДокументаДляXML(TypeNamedId, FunctionName);
    КонецЕсли;

    Если ЗначениеЗаполнено(функцияУПД) Тогда

```

(continues on next page)

(продолжение с предыдущей страницы)

```

        ДополнительныеПараметры = Новый Структура("ПараметрыСогласования, ДопСведения,
↪ФИОПодписанта, ФункцияУПД", Неопределено, Неопределено, " ", ФункцияУПД);
        Content = ОсновнойМодуль.ТиповойМодуль_ПолучитьКонтент(ТиповойМодульДиадока(),
↪РежимУправляемыхФорм, Документ1С, ТипКонтента_XDTO, ДополнительныеПараметры);
        ОсновнойМодуль.ЗаполнитьКонтентXDTOПоСтруктуре(Результат.Content, Content);
        КонецЕсли;
КонецЕсли;

```

Если документ 1С доработан, или сбор контента для его типа не определен в модуле, тогда можно воспользоваться шаблоном кода.

В настройке модуля на закладке «Подключаемый модуль» при нажатии на кнопку «Генерировать пример подключаемого модуля» откроется вспомогательная форма.

В этой форме надо отметить галочкой нужный формат документа и нажать «Сгенерировать шаблон кода ПМ».

Определение значений полей формата есть в документации.

Пример заполнения UtdSellerContent:

```

.....
Результат = Параметры.Результат;

Если ВРЕГ(ТипКонтента_XDTO) = ВРЕГ("UtdSellerContent") Тогда

    Если ТипЗнч(Основание) = Тип("ДокументСсылка.АР_НачислениеАренднойПлаты") Тогда
        Заполнить_UtdSellerContent(Результат.Content, Документ1С, ФункцияУПД);
        КонецЕсли;

КонецЕсли;

.....

Процедура Заполнить_UtdSellerContent(Контент, Параметры, ФункцияУПД)

    УстановитьЗначениеXDTO(Контент, "Function",           ФункцияУПД);
    УстановитьЗначениеXDTO(Контент, "Date",             Параметры.Дата);
    УстановитьЗначениеXDTO(Контент, "Number",          ПрефиксацияОбъектовКлиентСервер.
↪ПолучитьНомерНаПечать(Параметры.Номер, Истина, Ложь));
    УстановитьЗначениеXDTO(Контент, "Currency",        Параметры.ВалютаДокумента.Код);
    УстановитьЗначениеXDTO(Контент, "Creator",         Параметры.Организация.Наименование);
    УстановитьЗначениеXDTO(Контент, "GovernmentContractInfo",
↪Параметры.ИдентификаторГосКонтракта);
    УстановитьЗначениеXDTO(Контент, "ВалютаСсылка",    Параметры.ВалютаДокумента, Истина);

    Заполнить_ExtendedOrganizationInfo (Контент.Seller, Параметры.Организация);
    Заполнить_ExtendedOrganizationInfo (Контент.Buyer, Параметры.Контрагент);
    Заполнить_InvoiceTable              (Контент.InvoiceTable, Параметры, ФункцияУПД);
    Заполнить_AdditionalInfoId          (Контент.AdditionalInfoId, Неопределено);
    Заполнить_TransferInfo              (Контент.TransferInfo, Параметры);

КонецПроцедуры

```

1.2.1.5.2 Не формализованные документы

Для не формализованных документов заполняется мета и файл.

Мета содержит сведения о номере, дате документа, имени файла и т.д. в зависимости от типа документа.

Получить двоичные данные файла можно несколькими способами:

- с помощью метода основного модуля «ТиповойМодуль_СформироватьПечатнуюФорму» для типовых печатных форм документов «Счет на оплату» и «Акта сверки»

```
Если ID = "ID_Счет" Тогда

    ДополнительныеПараметры = Новый Структура("ПараметрыСогласования, ФИОПодписанта", Неопределено,
↪ " ");
    ДанныеПечатнойФормы = ОсновнойМодуль.ТиповойМодуль_
↪СформироватьПечатнуюФорму(ТиповойМодульДиалога(), РежимУправляемыхФорм, Документ1С, "СчетНаОплату
↪", ДополнительныеПараметры);

    Результат.Метаданные.DocumentDate = Документ1С.Дата;
    Результат.Метаданные.DocumentNumber = СокрЛП(Документ1С.Номер);
    Результат.Метаданные.FileName = ДанныеПечатнойФормы.ИмяФайла;

    Если TypeNameId = "ProformaInvoice" Тогда
        Результат.Метаданные.TotalSum = Документ1С.СуммаДокумента;
    ИначеЕсли TypeNameId = "Nonformalized" Тогда
        //Результат.ЗапрашиватьОтветнуюПодпись = Истина;
    КонецЕсли;

    Результат.ДвоичныеДанные = Новый ДвоичныеДанные(ДанныеПечатнойФормы.ИмяВременногоФайла);

    УдалитьФайлы(ДанныеПечатнойФормы.ИмяВременногоФайла);

КонецЕсли;
```

- с помощью метода «СформироватьВнешнююПечатнуюФорму» для внешних печатных форм

```
Если ID = "ID_ИмяВнешнейПечатнойФормы" Тогда

    СсылкаВПФ = Справочники.ДополнительныеОтчетыИОбработки.НайтиПоНаименованию(
↪"ИмяВнешнейПечатнойФормы", Истина);
    ИдентификаторКомандыПечатнойФормы = "ПротоколСогласованияЦен"; // идентификатор команды, должен_
↪соответствовать внешней ПФ
    ДанныеПечатнойФормы = СформироватьВнешнююПечатнуюФорму(Документ1С, СсылкаВПФ,
↪ИдентификаторКомандыПечатнойФормы);

    Результат.Метаданные.DocumentDate = Документ1С.Дата;
    Результат.Метаданные.DocumentNumber = СокрЛП(Документ1С.Номер);
    Результат.Метаданные.FileName = ДанныеПечатнойФормы.ИмяФайла;

    Результат.ЗапрашиватьОтветнуюПодпись = Истина;
    Результат.ДвоичныеДанные = Новый ДвоичныеДанные(ДанныеПечатнойФормы.ИмяВременногоФайла);

    УдалитьФайлы(ДанныеПечатнойФормы.ИмяВременногоФайла);

КонецЕсли;
```

- стандартными функциями из объектов 1С, например «ПрисоединенныеФай-

лы.ПолучитьДвоичныеДанныеФайла».

1.2.1.6 Отправка пакета-шаблона

Пакет-шаблон - это сообщение с шаблонами, на основании которых можно создать юридически значимые документы.

Для отправки таких пакетов необходимо выполнить следующее:

- Добавить необходимый вид пакета-шаблона в функции *ПолучитьТаблицуИспользуемыхПакетов*.

```
ТЗ = Новый ТаблицаЗначений;
ОсновнойМодуль.ЭДО_Службные_ДобавитьКолонкиВТаблицуЗначений(ТЗ, "ID, Наименование, Шаблон");
ОсновнойМодуль.ЭДО_Службные_ДобавитьСтрокуВТаблицуЗначений(ТЗ, "ID_Шаблоны_КС2_КС3", "Шаблоны КС2,
↔ КС3", Истина);
```

- Заполнить **ПараметрыШаблона** структуры *Пакет* в функции *ПодготовитьПакет*

```
Если ВидПакетаРазвернутый.ID = "ID_Шаблоны_КС2_КС3" Тогда

// Документы должен подготовить получатель шаблона
Пакет.ПараметрыШаблона.MessageFromBoxId = Пакет.ДанныеДД.CounterAgentId;
Пакет.ПараметрыШаблона.MessageToBoxId      = Пакет.ДанныеДД.OrganizationId;

// Документы должны быть отправлены все вместе в одном сообщении,
// подписать или отклонить документы можно разными сообщениями
Пакет.ПараметрыШаблона.LockMode = "Send";

КонецЕсли
```

- Если поля документа созданного на основании шаблона требуется дозаполнить, то необходимо указать соответствующий идентификатор настройки редактирования **ПараметрыШаблона.EditingSettingId** в функции *ПодготовитьЭлектронныйДокумент*.

```
Если ВидДокументаРазвернутый.ID = "ID_Шаблон_УПД_БезДатыНомера" Тогда
    Результат.ПараметрыШаблона.EditingSettingId = "ab30afed-e04f-4309-9b15-51a77f55e183";
КонецЕсли;
```

Если идентификатор настройки редактирования заполнен, то модуль не будет выполнять валидацию контента шаблона.

1.2.2 Дополнительная обработка исходящих пакетов

Когда стоит задача изменить поля пакета до отправки на сервер, поможет событие *ПослеПодготовкиПакета*. В этой функции получится переопределить:

- отправителя
- получателя
- подразделения
- признак заблокированного пакета и т.п.

Пример изменения получателя до отправки пакета:


```

Запрос = Новый Запрос;
Запрос.Текст =
    "ВЫБРАТЬ РАЗРЕШЕННЫЕ ПЕРВЫЕ 1
    |   ВЫБОР
    |       КОГДА Диадок_ЮрФизЛица.ID_ВладелецПодразделения <> """"
    |           ТОГДА Диадок_ЮрФизЛица.ID_ВладелецПодразделения
    |           ИНАЧЕ Диадок_ЮрФизЛица.ID
    |   КОНЕЦ КАК ID
    |ИЗ
    |   Справочник.Диадок_ЮрФизЛица КАК Диадок_ЮрФизЛица
    |ГДЕ
    |   Диадок_ЮрФизЛица.СвязанныйСправочник1 = &СвязанныйСправочник1";

Запрос.УстановитьПараметр("СвязанныйСправочник1", Пакет.Данные1С.Контрагент);
Пакет.ДанныеДД.CounteragentId = ОсновнойМодуль.ЭДО_Службные_ПолучитьРезультатЗапроса(Запрос, "ID
↪");

```

После отправки пакета документов, можно дополнить или изменить данные в документах 1С с помощью функции *ПослеОтправкиПакета*.

Пример заполнения комментария в счете-фактуре 1С сведениями о дате отправки его в пакете УПД:

```

Если Пакет.Данные1С.ВидПакета.Наименование = "УПД" Тогда

    СчетФактура1С = Пакет.Данные1С.Документ.ПолучитьОбъект();

    СчетФактура1С.Комментарий = "Отправлен " + Пакет.Данные1С.Ссылка.ДатаЗагрузки;
    СчетФактура1С.ОбменДанными.Загрузка = Истина;

    СчетФактура1С.Записать();

КонецЕсли;

```

1.2.3 Как обработать входящий пакет документов

Для сокращения времени сотрудников, потраченного на создание и контроль первичной документации, в подключаемом модуле существуют следующие механизмы:

- сопоставления полученных электронных документов с документами в базе 1С
- создания новых документов в 1С на основании полученных электронных документов
- дополнительный анализ входящих документов

1.2.3.1 Сопоставление документов

Для типовых конфигураций предусмотрено сопоставление входящих электронных документов в формате xml, установленном ФНС, с типовыми документами 1С.

Например, входящий документ формата «УПД» сопоставляется с документом «Счет-фактура полученный», для которого существует документ-основание «Поступление товаров и услуг». Параметрами для сопоставления являются следующие поля документа 1С:

- Организация
- Контрагент
- Дата входящего документа

- Номер входящего документа

Описание правил сопоставления электронных документов, в зависимости от их типа, происходит в процедуре *НайтиСопоставлениеДокумента*.

Вызов события происходит при нажатии на кнопку «Сопоставить» в карточке входящего документа или кнопки «Сопоставить документ» в меню основной формы «Действия».

Пример сопоставления с документом типа «СчетФактураПолученный»:

```
Если СтруктураДанных.ТипДокумента = "Invoice"
    ИЛИ (СтруктураДанных.ТипДокумента = "UniversalTransferDocument" И СтруктураДанных.
    ↪ФункцияДокумента = "Invoice") Тогда

    Результат = ПолучитьДокументы1С(ПараметрыОтбора, "СчетФактураПолученный");

КонецЕсли;
```

1.2.3.2 Правила ручного сопоставления документа

Автоматическое сопоставление может не определить подходящий документ. В этом случае поможет ручное сопоставление.

Для этого пользователь нажимает «Выбрать» в поле «Документ 1С», выбирает сначала тип, а затем и сам документ.

Для сокращения списка типов документов, которые подходят для сопоставления, используется событие *ПравилаСопоставленияДокумента*.

Кроме того, в этом методе задается фильтр на список документов. В типовом событии устанавливается отбор по организации и контрагенту.

Пример правила для сопоставления накладной:

```
Результат = Новый Структура;

РеквизитыДокумента = Параметры.РеквизитыДокумента;

ТипДокумента = РеквизитыДокумента.ТипДокумента;
ФункцияДокумента = РеквизитыДокумента.ФункцияДокумента;
Организация = РеквизитыДокумента.Организация;
Контрагент = РеквизитыДокумента.Контрагент;

ВозможныеВидыЭДО = ОсновнойМодуль.ЭДО_ВидыЭлектронныхДокументовПоФорматуЭДО(ТипДокумента,
    ↪ФункцияДокумента);
ВидыЭДО = ОсновнойМодуль.ЭДО_Перечисление_ВидыЭДО();

Отбор = Новый Структура("Организация, Контрагент", Организация, Контрагент);

Для Каждого ВидЭДО Из ВозможныеВидыЭДО Цикл

    Если ВидЭДО = ВидыЭДО.Накладная Тогда

        ОсновнойМодуль.ЭДО_ДобавитьПравилоСопоставленияДокумента(Результат, "Документы",
        ↪"ПоступлениеТоваровУслуг", Отбор);
        ОсновнойМодуль.ЭДО_ДобавитьПравилоСопоставленияДокумента(Результат, "Документы",
        ↪"ПоступлениеТоваровУслугВНТТ", Отбор);
        ОсновнойМодуль.ЭДО_ДобавитьПравилоСопоставленияДокумента(Результат, "Документы",
        ↪"ПоступлениеИзПереработки", Отбор);
```

(continues on next page)

(продолжение с предыдущей страницы)

```

        ОсновнойМодуль.ЭДО_ДобавитьПравилоСопоставленияДокумента(Результат, "Документы",
↪ "ВозвратТоваровОтПокупателя", Отбор);
        ОсновнойМодуль.ЭДО_ДобавитьПравилоСопоставленияДокумента(Результат, "Документы",
↪ "ПолучениеУслугПоПереработке", Отбор);

        КонецЕсли;

КонецЦикла;

Возврат Результат;

```

1.2.3.3 Создание документов

Для возможности быстро формировать документы в 1С на основании полученного формализованного документа существует процедура *ПолучитьЗаполненныйОбъектДокумента1С*.

Полученные данные в формате XML преобразуются в XDTO структуру. Затем каждое поле в создаваемом документе 1С заполняется соответствующим реквизитом из получившейся стурктуры XDTO.

Вызов события происходит при нажатии на кнопку «Создать» в карточке входящего документа или кнопки «Создать документ» в меню основной формы «Действия».

```

Если СтруктураДанных.ТипДокумента = "XmlTorg12"
    ИЛИ СтруктураДанных.ТипДокумента = "XmlAcceptanceCertificate"
    ИЛИ (СтруктураДанных.ТипДокумента = "UniversalTransferDocument" И СтруктураДанных.
↪ФункцияДокумента <> "Invoice")
    ИЛИ (СтруктураДанных.ТипДокумента = "UniversalCorrectionDocument" И СтруктураДанных.
↪ФункцияДокумента <> "Invoice") Тогда

        ЭтоКорректировка = (СтруктураДанных.ТипДокумента = "UniversalCorrectionDocument");

        ТекстXML = ОсновнойМодуль.ДвоичныеДанные_В_Текст(ОсновнойМодуль.ЭДО_
↪ПолучитьХранилищеКонтента(ДокументДД));
        ДеревоКонтента = ОсновнойМодуль.XML_В_XDTO(ТекстXML);
        ТабличнаяЧасть = ОсновнойМодуль.ЭДО_ПолучитьТабличнуюЧастьИзКонтентаXML(ДеревоКонтента,
↪ЭтоКорректировка);

        Если СтруктураДанных.ТипДокумента = "UniversalCorrectionDocument" Тогда
            СоздатьДокументКорректировки(ДокументДД, ПакетДД, СтруктураДанных, ТабличнаяЧасть,
↪ДеревоКонтента, Результат);
        Иначе
            СоздатьДокументПоступления(ДокументДД, ПакетДД, СтруктураДанных, ТабличнаяЧасть,
↪ДеревоКонтента, Результат);
        КонецЕсли;

КонецЕсли;

```

1.2.3.4 Поиск документов, созданных на основании шаблона

Идентификаторы всех документов созданных на основании шаблона содержатся в коллекции **Связи.TransformedToLetterIds** структуры шаблона документа.

Эту коллекцию можно использовать для анализа в функции *ПровестиАнализДокумента*.

```

Документ = ОсновнойМодуль.Документы_СохраненныйДокумент(ДокументДД);
Шаблон   = ОсновнойМодуль.Документы_ШаблонДокумента(Документ);

Если Не ЗначениеЗаполнено(Шаблон) Тогда
    Возврат Результат;
КонецЕсли;

LetterId  = Документ.Идентификаторы.LetterId;
DocumentId = Документ.Идентификаторы.DocumentId;

Документ1С      = Неопределено;
ТекстОшибки     = Неопределено;
КонтентОтвета   = Неопределено;
ДокументУжеПолучен = Ложь;

Для каждого Элемент Из Шаблон.Связи.TransformedToLetterIds Цикл

    Если Элемент.EntityId = DocumentId И Элемент.MessageId <> LetterId Тогда
        ДокументУжеПолучен = Истина;
        ТекстОшибки = НСтр("ru = 'Документ по шаблону уже получен'");
        Прервать;
    КонецЕсли;

КонецЦикла;

Если ДокументУжеПолучен Тогда
    ДействиеВДД = "ОтказатьВПодписи";
    КонтентОтвета = ПолучитьКонтентОтвета(ДокументДД, Документ1С, ДействиеВДД, ТекстОшибки);
Иначе
    ДействиеВДД = "Подписать";
    Документ1С = Шаблон.ДокументУчета;
    КонтентОтвета = ПолучитьКонтентОтвета(ДокументДД, Документ1С, ДействиеВДД, ТекстОшибки);
КонецЕсли;

Если ЗначениеЗаполнено(ТекстОшибки) Тогда
    Результат.Ошибки = Новый Массив;
    Результат.Ошибки.Добавить(ТекстОшибки);
КонецЕсли;

Если ЗначениеЗаполнено(ДействиеВДД) Тогда
    Результат.ДействиеВДД = ДействиеВДД;
КонецЕсли;

Если ЗначениеЗаполнено(Документ1С) Тогда
    Результат.ДокументДляСопоставления = Документ1С;
КонецЕсли;

Если КонтентОтвета <> Неопределено Тогда
    Результат.КонтентОтвета = КонтентОтвета;
КонецЕсли;

```

1.2.3.5 Сопоставление номенклатуры

При создании документа происходит поиск номенклатуры для сопоставления.

Логика поиска описана в служебной функции подключаемого модуля «**НайтиНоменклатуруПоставщика**». Типовые поля поиска: наименование и/или код из входящего документа.

Если найти подходящую номенклатуру не удалось, то происходит ручное сопоставление с помощью таблицы значений.

После окончания сопоставления происходит запись изменений в базу.

Определение места хранения и правил заполнения происходит в функции *СопоставитьНоменклатуруПоставщика*.

```

ДлинаНаименования = Метаданные.Справочники.НоменклатураПоставщиков.ДлинаНаименования;

Для каждого СтрокаТЗ Из ТаблицаНоменклатурыДляСопоставления Цикл

    НовыйОбъект = Справочники.НоменклатураПоставщиков.СоздатьЭлемент();

    НовыйОбъект.Владелец = СтрокаТЗ.Контрагент;
    НовыйОбъект.Наименование = Прав(СтрокаТЗ.Наименование, ДлинаНаименования); // Как правило
    ↪ окончание наименования является уникальным, поэтому пишем последние символы.
    НовыйОбъект.Идентификатор = СтрокаТЗ.Код;
    НовыйОбъект.Артикул = СтрокаТЗ.Артикул;
    НовыйОбъект.Номенклатура = СтрокаТЗ.Номенклатура;

    НовыйОбъект.Записать();

КонецЦикла;

```

1.2.3.6 Анализ документов

Выполнение анализа вызывается при нажатии на кнопку **Анализ выбранных документов** в меню «Действия».

Для описания какого-либо процесса проверки входящих электронных документов используется процедура *ПровестиАнализДокумента*.

Например, можно проверить наличие сопоставленных документов 1С и получить массив выявленных ошибок.

```

Если ТипДокумента = "xmltorg12"
    ИЛИ (ТипДокумента = "universaltransferdocument" И (ФункцияДокумента = "invoiceandbasic" ИЛИ
    ↪ ФункцияДокумента = "basic")) Тогда

    Если НЕ ЗначениеЗаполнено(Документ1С) Тогда
        Ошибки.Добавить("Не найдена накладная №: " + СтруктураДанных.НомерДокумента + " от " +
        ↪ Формат(СтруктураДанных.ДатаДокумента, "ДФ=dd.MM.yyyy") + " на сумму " + Формат(СтруктураДанных.
        ↪ СуммаДокумента, "ЧДЦ=2"));
        КонецЕсли;

КонецЕсли;

```

После выполненного анализа каждого документа, в процедуре *ПослеАнализаПакета* выполняется основная логика обработки действий, определенных в предыдущей процедуре.

Вызов анализа по пакетам происходит при нажатии на кнопку **Анализ выбранных пакетов** в меню «Действия».

Например, заполнение структуры ответа на входящий документ в зависимости от определенного действия (подписать, отказать в подписи и т.п.).

Для Каждого ТекДокумент Из ДокументыПакета Цикл

```

ДанныеДокумента = ТекДокумент.ДанныеДокумента;

ДокументДД = ТекДокумент.Ссылка;
Документ1С = ДанныеДокумента.ДокументВ1С;
ТекстОшибки = ДанныеДокумента.ТекстОшибки;

РезультатАнализа = ОсновнойМодуль.ЭДО_НовыйРезультатАнализаВходящегоДокумента();

СписокДействийТекДокумента = ОсновнойМодуль.ЭДО_ВариантыОтветныхДействийПоДокументу(ДокументДД,
↪ ОшибокНет);
РезультатАнализа.ДействиеВДД = ?(СписокДействийТекДокумента.Количество()=0, "", ↪
↪СписокДействийТекДокумента[0].Значение);

Если ЗначениеЗаполнено(РезультатАнализа.ДействиеВДД) Тогда
    РезультатАнализа.Вставить("КонтентОтвета", ПолучитьКонтентОтвета(ДокументДД, Документ1С, ↪
↪РезультатАнализа.ДействиеВДД, ТекстВсехОшибокПакета));
    КонецЕсли;

РезультатАнализа.Вставить("ДокументДляСопоставления", Документ1С);
ОсновнойМодуль.СохранитьРезультатАнализа(ДокументДД, РезультатАнализа);

КонецЦикла;

```

Завершающий этап анализа - выполнение определенных действий по подготовленным данным, определяется в процедуре *ВыполнитьТребуемоеДействие*.

Например, проводим документ 1С, если требуемое действие, сохраненное в 1С в документе подсистемы Диадока, называется «ПровестиДокумент».

```

ДокументДД = Параметры.ДокументДД;
ПакетДД = Параметры.ПакетДД;

ДанныеДокумента = ОсновнойМодуль.ЭДО_ПолучитьРеквизитыДокумента(ДокументДД);

Если ДанныеДокумента.ТребуемоеДействиеВ1С = "ПровестиДокумент" И ЗначениеЗаполнено(ДанныеДокумента.
↪ДокументВ1С) Тогда

    Документ1СОбъект = ДанныеДокумента.ДокументВ1С.ПолучитьОбъект();
    Документ1СОбъект.Записать(РежимЗаписиДокумента.Проведение);

    ОсновнойМодуль.ЭДО_ЗаписатьРеквизитыДокумента(ДокументДД, Новый Структура("ТребуемоеДействиеВ1С
↪", "")); // отработали действие

КонецЕсли;

```

1.2.4 Как доработать таблицу списка документов

В списке документов основного модуля выводятся наиболее часто требуемые поля, характеризующие полученный пакет или документ.

1.2.4.1 Дополнительные колонки

Помимо основных полей списка, которые заполняются в модуле, существует 5 колонок для вывода дополнительных сведений. Они называются «ДопРеквизит1», «ДопРеквизит2» и т.д.

Для изменения заголовков, типов и допустимых данных для каждой из них используется событие *ПолучитьНастройкиДополнительныхКолонокСпискаДокументов*.

Формирование списка колонок происходит в основной форме модуля при изменении режима отображения документов.

Пример переименования 3 колонок: «ДопРеквизит1», «ДопРеквизит2», «ДопРеквизит3»:

```
Режим = Параметры.Режим;

Если Режим = "ПолученныеДокументы" Тогда

    ДопКолонки = Новый Массив; // не более 5. Индекс элемента в массиве, увеличенный на единицу,
    ↳ совпадает с порядковым номером колонки
    ДопКолонки.Добавить(Новый Структура("Заголовок, Тип, ДопустимыеЗначения", "Состояние документа,
    ↳ 1С", Новый ОписаниеТипов("Строка")));
    ДопКолонки.Добавить(Новый Структура("Заголовок, Тип, ДопустимыеЗначения", "Сумма в 1С", Новый
    ↳ ОписаниеТипов("Число")));
    ДопКолонки.Добавить(Новый Структура("Заголовок, Тип, ДопустимыеЗначения", "Расхождение", Новый
    ↳ ОписаниеТипов("Число")));

    Возврат ДопКолонки;

КонецЕсли;
```

Заполнение данными колонок «ДопРеквизит» происходит:

- в функции *ПолучитьТекстЗапросаДляСпискаПакетовНаОтправкуПоМассивуВидовПакетов*
- в функции *ПослеОбновленияСпискаДокументов*

В первом варианте необходимо учитывать, что поля запроса с дополнительными данными должны иметь названия «ДопРеквизит1», «ДопРеквизит2» и т.д.

Во втором варианте заполнение происходит через непосредственное обращение к реквизиты таблицы СписокДокументов.

Обращение к событию происходит при каждом обновлении списка на форме модуля.

Пример заполнения значений дополнительных реквизитов с использованием события *ПослеОбновленияСпискаДокументов*:

```
СписокДокументов = Параметры.СписокДокументов;
Режим = Параметры.Режим;

Если Режим = "ПолученныеДокументы" Тогда

    Запрос = Новый Запрос(
        "ВЫБРАТЬ
        |   Парам.НомерСтроки,
        |   ВЫРАЗИТЬ(Парам.СуммаДокумента КАК ЧИСЛО) КАК СуммаДокументаДД,
        |   ВЫРАЗИТЬ(Парам.Документ КАК Документ.ПоступлениеТоваровУслуг) КАК Документ1С
        |ПОМЕСТИТЬ ВТ
        |ИЗ
        |   &СписокДокументов КАК Парам
```

(continues on next page)

(продолжение с предыдущей страницы)

```

|;
|
|////////////////////////////////////
|ВЫБРАТЬ
|    VT.НомерСтроки,
|    ВЫБОР
|        КОГДА VT.Документ1С ССЫЛКА Документ.ПоступлениеТоваровУслуг
|            ТОГДА ВЫБОР
|                КОГДА ВЫРАЗИТЬ(VT.Документ1С КАК Документ.ПоступлениеТоваровУслуг).
↪ПометкаУдаления
|                    ТОГДА ""Удален""
|                    КОГДА НЕ ВЫРАЗИТЬ(VT.Документ1С КАК Документ.ПоступлениеТоваровУслуг).
↪Проведен
|                    ТОГДА ""Не проведен""
|                    ИНАЧЕ """"
|                    КОНЕЦ
|                    ИНАЧЕ """"
|    КОНЕЦ КАК СостояниеДокумента,
|    ЕСТЬNULL(VT.Документ1С.СуммаДокумента, 0) КАК СуммаДокумента1С,
|    VT.СуммаДокументаДД - ЕСТЬNULL(VT.Документ1С.СуммаДокумента, 0) КАК Расхождение
|ИЗ
|    VT КАК VT"
);

Запрос.УстановитьПараметр("СписокДокументов", СписокДокументов.Выгрузить(,"НомерСтроки,
↪Документ, СуммаДокумента"));
Выборка = Запрос.Выполнить().Выбрать();

Пока Выборка.Следующий() Цикл
    СписокДокументов[Выборка.НомерСтроки - 1].ДопРеквизит1 = Выборка.СостояниеДокумента;
    СписокДокументов[Выборка.НомерСтроки - 1].ДопРеквизит2 = Выборка.СуммаДокумента1С;
    СписокДокументов[Выборка.НомерСтроки - 1].ДопРеквизит3 = Выборка.Расхождение;
КонецЦикла;

Возврат ПолучитьНастройкиДополнительныхКолонокСпискаДокументов(Новый Структура("Режим",
↪Режим));
КонецЕсли;

```

1.2.4.2 Доработка формы основного модуля

Для возможности редактирования управляемой формы основного модуля, например, для установки условного оформления списка документов, используется событие *УстановитьУсловноеОформление*

Вызывается при создании на сервере основной формы.

Пример изменения фона строки списка входящих документов:

```

Форма = Параметры.Форма;

//входящие документ на подпись
ЭлементУО = Форма.УсловноеОформление.Элементы.Добавить();

ПолеЭлемента = ЭлементУО.Поля.Элементы.Добавить();
ПолеЭлемента.Поле = Новый ПолеКомпоновкиДанных("СписокДокументов");

```

(continues on next page)

(продолжение с предыдущей страницы)

```

ОтборЭлемента = ЭлементУО.Отбор.Элементы.Добавить(Тип("ЭлементОтбораКомпоновкиДанных"));
ОтборЭлемента.ЛевоеЗначение = Новый ПолеКомпоновкиДанных("Объект.СписокДокументов.Направление");
ОтборЭлемента.ВидСравнения = ВидСравненияКомпоновкиДанных.Равно;
ОтборЭлемента.ПравоеЗначение = "Входящее";

ОтборЭлемента = ЭлементУО.Отбор.Элементы.Добавить(Тип("ЭлементОтбораКомпоновкиДанных"));
ОтборЭлемента.ЛевоеЗначение = Новый ПолеКомпоновкиДанных("Объект.СписокДокументов.Статус");
ОтборЭлемента.ВидСравнения = ВидСравненияКомпоновкиДанных.Содержит;
ОтборЭлемента.ПравоеЗначение = "Требуется подпись";

ЭлементУО.Оформление.УстановитьЗначениеПараметра("ЦветФона", WebЦвета.БледноЗеленый);

Возврат Истина;

```

Для редактирования обычных форм используются несколько событий.

При открытии основной формы модуля вызывается событие *ПриОткрытииФормы*

Пример снятия флажков у всех видов пакетов на отправку:

```

Форма = Параметры.Форма;
ИмяФормы = Параметры.ИмяФормы;

Если ИмяФормы = "ОсновнаяФорма" Тогда

    ДеревоФильтров = Форма.ДеревоФильтров;

    СтрокаОтправкаПакетов = ДеревоФильтров.Строки.Найти("ОтправкаПакетов");

    Если СтрокаОтправкаПакетов <> Неопределено Тогда

        Для Каждого СтрокаВидыПакетов Из СтрокаОтправкаПакетов.Строки Цикл
            СтрокаВидыПакетов.Пометка = Ложь;
        КонечЦикла;

    КонечЕсли;

КонечЕсли;

Возврат Истина;

```

При получении данных списка документов основной формы модуля вызывается событие *ПриПолученииДанныхТабличнойЧастии*

Пример выделения каждой второй строки списка документов:

```

Форма = Параметры.Форма;
ИмяФормы = Параметры.ИмяФормы;
Элемент = Параметры.Элемент;
ОформленияСтрок = Параметры.ОформленияСтрок;

Для Каждого Эл Из ОформленияСтрок Цикл

    Если Эл.ДанныеСтроки.НомерСтроки % 2 = 0 Тогда
        Эл.Ячейки.Документ.ЦветФона = WebЦвета.Аквамарин;
        Эл.ЦветФона = WebЦвета.Древесный;
    КонечЕсли;

```

(continues on next page)

```

КонецЦикла;

Возврат Истина;

```

1.2.5 Как добавить кнопки в действия

Добавления новой кнопки происходит в меню «Действия» списка документа.

Для определения заголовка кнопки используется функция *ЗаполнитьСписокДополнительныхКнопок*.

Формирование списка кнопок происходит в основной форме модуля при изменении режима отображения документов.

Пример добавления кнопки для выполнения сопоставления отправленных документов:

```

РежимОтображенияДокументов = Параметры.РежимОтображенияДокументов;
ТаблицаКнопок = Параметры.ТаблицаКнопок;

Если РежимОтображенияДокументов = "ОтправленныеДокументы" Тогда

    НовСтрока = ТаблицаКнопок.Добавить();
    НовСтрока.Имя = "СопоставитьОтправленныйДокумент";
    НовСтрока.Заголовок = "Сопоставить документ";

КонецЕсли;

```

Алгоритм действия новой кнопки описывается в функции *НажатиеДополнительнойКнопки*.

Вызов происходит из основной формы модуля при нажатии на дополнительную кнопку.

Пример выполнения сопоставления отправленных документов при нажатии на кнопку:

```

РежимОтображенияДокументов = Параметры.РежимОтображенияДокументов;
ВыделенныеСтроки = Параметры.ВыделенныеСтроки;
ИмяКнопки = Параметры.ИмяКнопки;

Если РежимОтображенияДокументов = "ОтправленныеДокументы" И ИмяКнопки =
↳ "СопоставитьОтправленныйДокумент" Тогда

    Для Каждого ВыбраннаяСтрока Из ВыделенныеСтроки Цикл
        Если ЗначениеЗаполнено(ВыбраннаяСтрока.ДокументЭДО) Тогда

            РезультатФункции = ОсновнойМодуль.НайтиИСопоставитьДокументДД(ВыбраннаяСтрока.
↳ ДокументЭДО, ВыбраннаяСтрока.Пакет);
            ОсновнойМодуль.ОбработатьРезультатФункции(РезультатФункции);

        КонецЕсли;
    КонецЦикла;

КонецЕсли;

```

1.2.6 Регламентное задание

При большом объеме обрабатываемых документов часть задач можно переложить на выполнение их в фоновом режиме без участия пользователя.

Алгоритм работы описывается в функции *ВыполнитьРегламентныеДействия*.

Выполнение алгоритма запускается двумя способами:

- при нажатии кнопки «Выполнить регламентное задание» в настройках модуля
- при запуске регламентного задания программно из конфигурации

В первом случае алгоритм запускается на клиенте. Поэтому он может содержать обращения к экспортным функциям, расположенным в формах основного модуля.

Второй вариант имеет несколько ограничений:

- использовать можно только методы, работающие на сервере
- обращение к формам модуля невозможно
- требуется внесение изменений в конфигурацию (добавление регламентного задания в 1С, описание функции вызова *ВыполнитьРегламентныеДействия*)

Чаще всего использование регламентного задания предполагает:

- обновление ленты событий
- проведение анализа документов и запись требуемых действий
- отправка на подпись или согласование
- выполнение сопоставления документов
- подписание и отправка сертификатом установленным на сервере 1С

Пример получения новых событий и отправка документов:

```
// ОБРАБОТКА СОБЫТИЙ ЛЕНТЫ, В Т.Ч. ЗАГРУЗКА НОВЫХ ДОКУМЕНТОВ

КоличествоПорцийСобытий = 10; // Кол-во вычитываемых порций событий (каждая порция содержит не
↳ более 100 событий)
Для Каждого СтрокаОрганизации Из ОсновнойМодуль.ЭДО_Модуль_ТаблицаОрганизацийПользователя() Цикл
    ОсталосьСобытийВЛенте = ОсновнойМодуль.ЭДО_Модуль_
↳ ОбработатьНовыеСобытияДокументов(СтрокаОрганизации, КоличествоПорцийСобытий);
КонецЦикла;

// ВЫПОЛНЕНИЕ MessagePatchToPost

Режим = ""; // Режим исполнения: ПередатьНаСогласование / ПередатьНаПодпись / ПередатьПоМаршруту /
↳ Согласование / ОтказВСогласовании

ПараметрыMessagePatchToPost = Новый Структура;
ПараметрыMessagePatchToPost.Вставить("Действие" , Режим); // вариант режима
↳ MessagePatchToPost
ПараметрыMessagePatchToPost.Вставить("ИдентификаторСотрудника" , Неопределено); // внутренний ID
↳ Диалога
ПараметрыMessagePatchToPost.Вставить("ИдентификаторПодразделения", Неопределено); // внутренний ID
↳ Диалога
ПараметрыMessagePatchToPost.Вставить("ИдентификаторМаршрута" , Неопределено); // внутренний ID
↳ Диалога
ПараметрыMessagePatchToPost.Вставить("Комментарий" , ""); // произвольный
↳ текст

// Произвольная коллекция документов Диалога, которые необходимо пропатчить (необходимо собрать по
↳ нужному алгоритму).
// Элементы этой коллекции должны содержать ключ "ДокументДД".
```

(continues on next page)

```

ТаблицаДокументов = Новый ТаблицаЗначений;

Если ЗначениеЗаполнено(Режим) Тогда
    ОсновнойМодуль.ЭДО_
    ↪ОтправитьMessagePatchToPostДляВыбранныхСтрокСпискаДокументов(ТаблицаДокументов,
    ↪ПараметрыMessagePatchToPost);
КонецЕсли;

```

При большом объеме документов на отправку возникает потребность убрать эту задачу с пользователя. В этом случае тот сертификат, которым будет производиться подписание, устанавливается на сервере 1С.

Отпечаток сертификата указывается в настройках основного модуля для организации: элемент организации → закладка «Прочие настройки» → поле «Отпечаток сертификата на сервере 1С»

Пример подписания документов на сервере:

```

// ПРИМЕР ПОДПИСАНИЯ И ОТПРАВКИ СЕРТИФИКАТОМ УСТАНОВЛЕННЫМ НА СЕРВЕРЕ 1С
// ОГРАНИЧЕНИЯ: сервер под Windows; закрытый ключ сертификата установлен под учеткой агента
↪сервера 1С с сохраненным пин-кодом

// 1. Авторизация под сертификатами сервера 1С
ОсновнойМодуль.ЭДО_АвторизоватьсяПодСертификатомНаСервере1С();
КонтекстСеанса = ОсновнойМодуль.ЭДО_КонтекстСеансаКлиентСервер();

// 2. Параметры для получения списка пакетов на отправку
ТаблицаВидовПакетов = ОсновнойМодуль.ЭДО_СправочникМенеджер_ПолучитьСписокЭлементов("ВидыПакетов
↪");
МассивВидовПакетов = ТаблицаВидовПакетов.ВыгрузитьКолонку("Ссылка");

ПараметрыОбновленияСписка = Новый Структура;
ПараметрыОбновленияСписка.Вставить("Режим", "ОтправкаПакетов");
ПараметрыОбновленияСписка.Вставить("НачалоПериода", ДобавитьМесяц(ТекущаяДата(), -1));
ПараметрыОбновленияСписка.Вставить("КонецПериода", КонецДня(ТекущаяДата()));
ПараметрыОбновленияСписка.Вставить("МассивВыбранныхВидов", МассивВидовПакетов);

//3. Отправка пакетов по организациям, в которых авторизовались под сертом сервера 1С
Для Каждого Элемент Из КонтекстСеанса Цикл

    СтрокаКонтекста = Элемент.Значение;
    Организация = СтрокаКонтекста.ОрганизацияДиадок.СвязанныйСправочник1;

    ПараметрыОбновленияСписка.Вставить("ОтборПоОрганизации", Организация);

    ОсновнойМодуль.ЭДО_Модуль_ОбновитьСписокДокументов(ПараметрыОбновленияСписка);

    //1 вариант: последовательная отправка
    Для Каждого СтрокаТЧ Из ОсновнойМодуль.СписокДокументов Цикл
        ОсновнойМодуль.ЭДО_ПодготовитьИОтправитьПакет(СтрокаТЧ);
    КонецЦикла;

    //2 вариант: фоновая отправка
    ОсновнойМодуль.ЭДО_ПодготовитьИОтправитьПакетыВФоне(ОсновнойМодуль.СписокДокументов);

КонецЦикла;

```

1.2.7 Дополнительные возможности

Существуют общие события, которые либо относятся к документам обоих направлений, либо к работе модуля в целом.

1.2.7.1 После обработки ленты событий

Обновление данных существующих документов (например, изменение статуса), получение информации о новых документах происходит при чтении ленты событий.

Выполняется либо при нажатии на кнопку «Обмен с сервером», либо при выполнении регламентного задания при вызове соответствующего метода основного модуля.

Для получения таблицы идентификаторов документов, которые пришли в текущей партии событий (не более 100), используется *ПослеОбработкиЛентыСобытий*.

Идентификаторы могут пригодиться для выполнения методов API.

Пример сохранения данных о пользователе или подразделении, которым были переданы документы на согласование.

```

Для Каждого СтруктураИдентификаторов Из ИдентификаторыДокументов Цикл
    Если ЗначениеЗаполнено(СтруктураИдентификаторов.ДокументЭДО) Тогда
        ДокументЭДО = СтруктураИдентификаторов.ДокументЭДО;
    Иначе

        BoxId          = СтруктураИдентификаторов.BoxId;
        LetterId       = СтруктураИдентификаторов.LetterId;
        DocumentId     = СтруктураИдентификаторов.DocumentId;

        ДокументЭДО = ОсновнойМодуль.ЭДО_Диадок_Документы_ПолучитьДокумент(BoxId, LetterId,
↵DocumentId);

    КонецЕсли;

    Если Не ЗначениеЗаполнено(ДокументЭДО) Тогда
        Продолжить;
    КонецЕсли;

    РеквизитыДокумента = ОсновнойМодуль.ЭДО_ПолучитьРеквизитыДокумента(ДокументЭДО);
    МаршрутизацияДокумента = ОсновнойМодуль.ЭДО_ПолучитьМаршрутизациюДокумента(ПараметрыСервиса,
↵РеквизитыДокумента);
    Если МаршрутизацияДокумента.Количество() > 0 Тогда

        ДанныеМаршрутизации = МаршрутизацияДокумента[МаршрутизацияДокумента.ВГраница()];
        AttachmentType      = ДанныеМаршрутизации.AttachmentType;
        Если AttachmentType = "ResolutionRequest" Тогда

            TargetResolution      = Неопределено;
            TargetResolutionString = Неопределено;

            Если ЗначениеЗаполнено(ДанныеМаршрутизации.User) И ЗначениеЗаполнено(ДанныеМаршрутизации.
↵UserId) Тогда
                TargetResolution      = ДанныеМаршрутизации.UserId;
                TargetResolutionString = ДанныеМаршрутизации.User;
            ИначеЕсли ЗначениеЗаполнено(ДанныеМаршрутизации.Department) И
↵ЗначениеЗаполнено(ДанныеМаршрутизации.DepartmentId) Тогда

```

(continues on next page)

(продолжение с предыдущей страницы)

```

        TargetResolution                = ДанныеМаршрутизации.DepartmentId;
        TargetResolutionString          = ДанныеМаршрутизации.Department;
    КонечЕсли;

    Если ЗначениеЗаполнено(TargetResolution) Тогда
        ОсновнойМодуль.ЭДО_СправочникМенеджер_УстановитьЗначениеДопРеквизита(ДокументЭДО,
↪"TargetResolution", TargetResolution, TargetResolutionString);
        КонечЕсли;

    ИначеЕсли AttachmentType = "Resolution"
        Или AttachmentType = "ResolutionRequestDenial"
        Или AttachmentType = "ResolutionRouteRemovalAttachment" Тогда

        ОсновнойМодуль.ЭДО_СправочникМенеджер_УдалитьДопРеквизит(ДокументЭДО,
↪"TargetResolution");

    КонечЕсли;

КонечЕсли;

КонечЦикла;

Возврат Истина;

```

1.2.7.2 При изменении статуса документа

Если для документов при смене статуса требуется вносить изменения, например в связанный документ 1С, тогда следует использовать функцию *ПослеУстановкиСтатусаДокумента*.

Функция выполняется после нажатия на кнопку «Обмен с сервером» и при отправке ответа по входящему документу.

Пример записи в комментарий отправленного документа 1С о том, что он подписан контрагентом:

```

ДокументДД = Параметры.ДокументДД;
Статус = Параметры.Статус;

Если Найти(Статус, "Подписан") > 0
    И ДокументДД.Документ <> Неопределено Тогда

        Документ1С = ДокументДД.Документ.ПолучитьОбъект();

        Документ1С.Комментарий = "Подписан";
        Документ1С.ОбменДанными.Загрузка = Истина;

        Документ1С.Записать();

КонечЕсли;

Возврат Истина;

```

1.2.7.3 Переопределение типов Организации и Контрагентов

Для нетиповых конфигураций может потребоваться изменить тип значения Организации и Контрагентов. В том числе для запросов.

Функция *ОпределитьТипЗначенияОбъекта* предназначена для получения нужного типа по синониму.

Обращение к ней происходит в следующих случаях:

- при открытии форм организаций и контрагентов в модуле
- при определении типов значений для отборов СКД и быстрых отборов
- при получении списка контрагентов
- при групповом сопоставлении контрагентов

Типовой вариант функции:

```
Если Параметры.Синоним = "Организации" Тогда
    Результат = "СправочникСсылка.Организации";
ИначеЕсли Параметры.Синоним = "Контрагенты" Тогда
    Результат = "СправочникСсылка.Контрагенты";
Иначе
    Результат = Неопределено;
КонецЕсли;
```

1.2.7.4 Фильтр контрагентов для приглашения

В форме списка контрагентов модуля есть возможность опривить приглашение контрагентам из списка 1С.

В раздел «Поиск и приглашение» попадают те контрагенты, с которыми нет партнерских отношений.

Часто в списке контрагентов 1С есть те, с кем уже нет взаимодействий.

Для возможности фильтровать такие варианты, существует функция *ПолучитьКоличествоДокументовЗаПериод*, которая определяет значение поля «ДокументовЗаПериод».

По умолчанию происходит оценка количества накладных с контрагентами за 3 месяца. Но этот период, а также состав документов для отбора, можно изменять.

Пример запроса для расчета количества документов «Реализация товаров и услуг»:

```
Запрос.Текст =
"ВЫБРАТЬ РАЗРЕШЕННЫЕ
| РеализацияТоваровУслуг.Контрагент КАК Связь1,
| КОЛИЧЕСТВО(РАЗЛИЧНЫЕ РеализацияТоваровУслуг.Ссылка) КАК ДокументовЗаПериод
| ПОМЕСТИТЬ ВТ_Документы
| ИЗ
| Документ.РеализацияТоваровУслуг КАК РеализацияТоваровУслуг
| ГДЕ
| РеализацияТоваровУслуг.Проведен = ИСТИНА
| И РеализацияТоваровУслуг.Организация = &Организация
| И РеализацияТоваровУслуг.Дата >= &ДатаНачала
| И РеализацияТоваровУслуг.Контрагент В(&МассивКонтрагентов)
|
| СГРУППИРОВАТЬ ПО
| РеализацияТоваровУслуг.Контрагент
| ;
|
| //////////////////////////////////////
| ВЫБРАТЬ
| ВТ_Документы.Связь1 КАК Связь1,
| СУММА(ВТ_Документы.ДокументовЗаПериод) КАК ДокументовЗаПериод
```

(continues on next page)

(продолжение с предыдущей страницы)

```
|ИЗ  
| ВТ_Документы КАК ВТ_Документы  
|  
|СГРУППИРОВАТЬ ПО  
| ВТ_Документы.Связь1";
```


2.1 Исходящие документы

2.1.1 ПодготовитьПакет

Синтаксис:

ПодготовитьПакет(<ВидПакетаРазвернутый>, <СтрокаСписка>, <Пакет>)

Описание:

Используется для формирования правил, по которым каждый из пакетов на отправку будет наполняться документами.

Параметры:

- **ВидПакетаРазвернутый** (тип Структура)- реквизиты вида пакета из функции *ПолучитьТаблицуИспользуемыхПакетов*.
- **СтрокаСписка** (тип Структура) - структура, соответствующая строке из интерфейса списка пакетов для отправки.
- **Пакет** (тип Структура Пакет) - структура, которая содержит основные сведения о пакете и его составе.

Пример использования: *Как подготовить пакет документов для отправки*

2.1.2 ПодготовитьЭлектронныйДокумент

Синтаксис:

ПодготовитьЭлектронныйДокумент(<Результат>, <ВидДокументаРазвернутый>, <ДополнительныеПараметры>)

Описание:

Заполнение контента всех документов, которые необходимо включить в пакет для отправки.

Результатом является заполненный ХДТО-объект **Результат.Content** или структура метаданных. Структура зависит от типа контента.

Параметры:

- **Результат** (тип Структура)- структура, описывающая электронный документ. Элементы структуры определяются в функции основного модуля «ЭДО_ДокументМенеджер_РеквизитыДокументаСтрокой».
- **ВидДокументаРазвернутый** (тип Структура)- реквизиты вида документа из функции *ПолучитьТаблицуИспользуемыхВидовДокументов*.
- **ДополнительныеПараметры** (тип Структура) - дополнительные параметры, которые могут потребоваться при сборе контента. Передаются из события *ПодготовитьПакет*.
- **ДанныеСотрудника** (тип Структура) - ФИО, должность, ИНН сертификата пользователя текущего сеанса.

Пример использования: *Как подготовить пакет документов для отправки*

2.1.3 ПолучитьТаблицуИспользуемыхВидовДокументов

Синтаксис:

```
ПолучитьТаблицуИспользуемыхВидовДокументов()
```

Возвращает:

Таблица значений

Описание:

В функции определяется название видов документов на отправку и их характеристики.

Список видов документов хранится в справочнике «Диадок: Дополнительные справочники» (Имя справочника = «ВидыДокументов»).

Состав колонок возвращаемой таблицы:

- **ID** (тип Строка) — внутренний идентификатор для определения вида документа в коде. Не использовать пробелы и другие служебные символы, кроме «_».
- **Наименование** (тип Строка, 150) — пользовательское представление вида документа в списке «Отправленные документы».
- **ТипДокументаAPI** (тип Строка) — один из *типов документов*.
- **ТипКонтентаAPI** (тип Строка) - один из *форматов документов*.
- **ФункцияДокументаAPI** - (тип Строка) - функция в УПД или УКД.
 - Для приказа 155: *СЧФ, ДОП, СЧФДОП*.
 - Для приказа 189: *КСФ, ДИС, КСЧФДИС*.

Пример использования: *Как подготовить пакет документов для отправки*

2.1.4 ПолучитьТаблицуИспользуемыхПакетов

Синтаксис:

```
ПолучитьТаблицуИспользуемыхПакетов()
```

Возвращает:

Таблица значений

Описание:

В функции определяется название пакета на отправку, которое видит пользователь в интерфейсе.

Список пакетов хранится в справочнике «Диадок: Дополнительные справочники» (Имя справочника = «ВидыПакетов»).

Состав колонок возвращаемой таблицы:

- **ID** (тип Строка) — внутренний идентификатор для определения пакета в коде. Не использовать пробелы и другие служебные символы, кроме «_».
- **Наименование** (тип Строка, 150) — пользовательское представление пакета в списке «Пакеты на отправку».
- **УПД** (тип Булево) — признак содержания в пакете документа типа «Универсальный передаточный документ» (UniversalTransferDocument).
- **Шаблон** (тип Булево) — признак того, что пакет является пакетом-шаблоном.

Пример использования: *Как подготовить пакет документов для отправки*

2.1.5 Получить текст запроса для списка пакетов на отправку по массиву видов пакетов

Синтаксис:

ПолучитьТекстЗапросаДляСпискаПакетовНаОтправкуПоМассивуВидовПакетов (<МассивВидовПакетов>)

Возвращает:

Текст запроса

Описание:

В функции формируется текст запроса для выборки главных документов, без которых пакет не готов к отправке.

Параметры:

- **МассивВидовПакетов** (тип Массив) — ссылки на элементы справочника «Диадок_ДополнительныеСправочники» с отбором по имени справочника «ВидыПакетов».
- **ПараметрыЗапроса** (тип Структура) - список параметров, которые передаются в запрос при выполнении. С помощью метода структуры «Вставить», дополняется новыми, если они используются в запросе.

Поля возвращаемого запроса:

- **Документ** — ссылка на документ 1С, который является основным для формирования определяемого пакета.
- **НомерДокумента** — номер документа 1С.
- **ДатаДокумента** — дата документа 1С.
- **Контрагент** — ссылка на контрагента 1С, который является получателем пакета.
- **Организация** — ссылка на организацию 1С, которая является отправителем пакета.
- **СуммаДокумента** — сумма документа 1С.

- **ВидПакетаID** — внутренний идентификатор вида пакета, определенный ранее в функции *ПолучитьТаблицуИспользуемыхПакетов*.

Пример использования: *Как подготовить пакет документов для отправки*

2.1.6 ПослеОтправкиПакета

Синтаксис:

ПослеОтправкиПакета(<Пакет>)

Описание:

Процедура используется в случае необходимости изменения данных в 1С, после отправки пакета на сервер Диадок, когда пакету и документам присвоены ID.

Параметры:

- **Пакет** (тип Структура Пакет) - структура, которая содержит основные сведения об отправленном пакете и его составе.

Пример использования: *Дополнительная обработка исходящих пакетов*

2.1.7 ПослеПодготовкиПакета

Синтаксис:

ПослеПодготовкиПакета(<Пакет>)

Описание:

В процедуре можно изменить любые свойства подготовленного пакета до его отправки. Например, переопределить отправителя, получателя, подразделение, заблокированность, тестовость и т.д.

Параметры:

- **Пакет** (тип Структура Пакет) - структура, которая содержит основные сведения о подготовленном пакете и его составе.

Пример использования: *Дополнительная обработка исходящих пакетов*

2.2 Входящие документы

2.2.1 ВыполнитьТребуемоеДействие

Синтаксис:

ВыполнитьТребуемоеДействие(<Параметры>)

Возвращает:

Ссылка на документ 1С / Неопределено

Описание:

Выполняется требуемое действие, сохраненное в обрабатываемом документе 1С подсистемы Диадок.

Параметры:

Элементы структуры **Параметры:**

- **ДокументДД** - ссылка на справочник «Диадок_Документы», содержащий описание электронного документа, для которого надо выполнить действие
- **ПакетДД** - ссылка на справочник «Диадок_ПакетыДокументов», содержащий описание пакета, к которому относится электронный документ

Пример использования: *Как обработать входящий пакет документов*

2.2.2 НайтиСопоставлениеДокумента

Синтаксис:

НайтиСопоставлениеДокумента(<Параметры>)

Возвращает:

Неопределено / Массив ссылок на документ 1С

Описание:

Функция ищет ссылку на объект в 1С. Параметры поиска задаются согласно реквизитам электронного документа.

Параметры:

Элементы структуры **Параметры**:

- **ДокументДД** - электронный документ, для которого выполняется сопоставление (СправочникСсылка.Диадок_Документы)
- **ПакетДД** - пакет документов, в состав которого входит ДокументДД (СправочникСсылка.Диадок_ПакетыДокументов)

Пример использования: *Как обработать входящий пакет документов*

2.2.3 ПолучитьЗаполненныйОбъектДокумента1С

Синтаксис:

ПолучитьЗаполненныйОбъектДокумента1С(<Параметры>)

Возвращает:

Ссылку на документ 1С / Структуру, содержащую ссылки на документы 1С и описание ошибок / Неопределено

Описание:

Создание документа по данным из Диадока.

Параметры:

Элементы структуры **Параметры**:

- **ДокументДД** - ссылка на справочник «Диадок_Документы», содержащий описание электронного документа, который надо создать в 1С
- **ПакетДД** - ссылка на справочник «Диадок_ПакетыДокументов», содержащий описание пакета, к которому относится электронный документ
- **Контент** - двоичные данные контента электронного документа, полученное с сервера Диадока

Пример использования: *Как обработать входящий пакет документов*

2.2.4 ПослеАнализаПакета

Синтаксис:

ПослеАнализаПакета(<Параметры>)

Возвращает:

Ссылка на документ 1С / Неопределено

Описание:

Вызывается действием «Выполнить анализ пакета»: сначала для каждого документа выполняется событие ПровестиАнализДокумента, потом ПослеАнализаПакета.

Параметры:

Элементы структуры **Параметры**:

- **ПакетДД** - ссылка на справочник «Диадок_ПакетыДокументов», содержащий описание пакета, к которому относится электронный документ
- **ДокументыПакета** - таблица ссылок на элементы справочника «Диадок_Документы», которые относятся к данному пакету

Поля таблицы значений **ДокументыПакета**:

- **Ссылка** - элементы справочника «Диадок_Документы»
- **ТипДокумента** - наименование *типа документа*
- **ДатаДокумента** - дата документа 1С или из контента электронного документа
- **НомерДокумента** - номер документа 1С или из контента электронного документа

Пример использования: *Как обработать входящий пакет документов*

2.2.5 ПравилаСопоставленияДокумента

Синтаксис:

ПравилаСопоставленияДокумента(<Параметры>)

Описание:

Задаёт ограничения на типы документов 1С, которые выпадают при ручном сопоставлении документа ЭДО.

Параметры:

Элементы структуры **Параметры**:

- **РеквизитыДокумента** - структура данных документа ЭДО

Пример использования: *Как обработать входящий пакет документов*

2.2.6 ПровестиАнализДокумента

Синтаксис:

ПровестиАнализДокумента(<Параметры>)

Возвращает:

Структура

Описание:

Выполняет анализ отдельно взятого входящего документа. Результата анализа заполняется в структуру `Параметры.РезультатАнализа`.

Параметры:

Элементы структуры `Параметры`:

- **ДокументДД** - ссылка на справочник «Диадок_Документы», содержащий описание электронного документа, для которого надо провести анализ
- **ПакетДД** - ссылка на справочник «Диадок_ПакетыДокументов», содержащий описание пакета, к которому относится электронный документ
- **РезультатАнализа** - структура, содержащая поля для сохранения результата анализа

Элементы структуры `РезультатАнализа`:

- **Ошибки** - массив строк с описаниями ошибок
- **ДокументДляСопоставления** - ссылка на документ/справочник 1С, соответствующий данному документу
- **ДействиеВ1С** - действие, которое необходимо выполнить с данным документом в 1С (Строка). Например, «СоздатьСчетФактуру», «ПометитьНаУдаление» и т.д.
- **ДействиеВДД** - действие, которое необходимо выполнить с данным документом на сервере Диодока (Строка). Допустимые варианты - «Подписать», «ОтказатьВПодписи», «ЗапроситьУточнение»
- **ТипОшибкиВалидации** - произвольный текст, отображаемый в списке, для классификации ошибок
- **КонтентОтвета** - ОбъектХДТО, соответствующий титулу покупателя (для формализованных документов) или отказу в подписи. Именно такой ответ будет отправлен на сервер Диодока, если заполнено поле «ДействиеВДД».

Пример использования: *Как обработать входящий пакет документов*

2.2.7 СопоставитьНоменклатуруПоставщика

Синтаксис:

`СопоставитьНоменклатуруПоставщика(<ТаблицаНоменклатурыДляСопоставления>)`

Возвращает:

Структуру с ключом «ОписаниеОшибки». Значение заполняется текстом возникшей ошибки при попытке записать сопоставление.

Описание:

Выполняет запись сопоставленной пользователем номенклатуры поставщика.

Параметры:

Состав колонок и возможные типы значений определяются в функции подключаемого модуля «**ДобавитьНоменклатуруДляСопоставления**»

Типовые колонки таблицы значений **ТаблицаНоменклатурыДляСопоставления**:

- **Контрагент** - ссылка на контрагента в базе 1С, от которого пришел электронный документ

- **Наименование** - наименование номенклатуры поставщика, как оно указано в электронном документе
- **Код** - код номенклатуры поставщика, как он указан в электронном документе
- **Артикул** - артикул номенклатуры поставщика, как он указан в электронном документе
- **Номенклатура** - ссылка на номенклатуру для сопоставления в базе 1С

Пример использования: *Как обработать входящий пакет документов*

2.3 Дополнение формы модуля

2.3.1 ЗаполнитьСписокДополнительныхКнопок

Синтаксис:

ЗаполнитьСписокДополнительныхКнопок(<Параметры>)

Описание:

Определение дополнительных кнопок в меню «Действия» основной формы через дополнение таблицы значений «ТаблицаКнопок».

Параметры:

Элементы структуры **Параметры:**

- **РежимОтображенияДокументов** - текущий режим отображения документов
- **ТаблицаКнопок** - таблица значений, для хранения описания новых кнопок

Режим отображения документов принимает значения:

- ОтправкаПакетов
- ОтправленныеДокументы
- ПолученныеДокументы
- ПолученныеПакеты

Таблица кнопок содержит поля:

- **Имя** - наименование кнопки для внутреннего обращения
- **Заголовок** - представление кнопки в меню

Пример использования: *Как добавить кнопки в действия*

2.3.2 НажатиеДополнительнойКнопки

Синтаксис:

НажатиеДополнительнойКнопки(<Параметры>)

Описание:

Обработчики нажатия дополнительных кнопок, которые задали в *ЗаполнитьСписокДополнительныхКнопок*.

Параметры:

Элементы структуры **Параметры:**

- **РежимОтображенияДокументов** - текущий режим отображения документов
- **ВыделенныеСтроки** - массив выделенных строк списка документов
- **ИмяКнопки** - внутреннее наименование кнопки, нажатие которой выполнено

Режим отображения документов принимает значения:

- ОтправкаПакетов
- ОтправленныеДокументы
- ПолученныеДокументы
- ПолученныеПакеты

Пример использования: *Как добавить кнопки в действия*

2.3.3 Получить настройки дополнительных колонок списка документов

Синтаксис:

ПолучитьНастройкиДополнительныхКолонокСпискаДокументов(<Параметры>)

Возвращает:

Массив дополнительных колонок списка документов в форме модуля

Описание:

Определение дополнительных колонок списка документов/пакетов (не более 5 штук).

Параметры:

Элементы структуры **Параметры**:

- **Режим** - текущий режим отображения списка документов

Режим принимает значения:

- ОтправкаПакетов
- ОтправленныеДокументы
- ПолученныеДокументы
- ПолученныеПакеты

Пример использования: *Как доработать таблицу списка документов*

2.3.4 ПослеОбновленияСпискаДокументов

Синтаксис:

ПослеОбновленияСпискаДокументов(<Параметры>)

Возвращает:

Массив заполненных дополнительных колонок

Описание:

Обработчики заполнения доп. колонок, которые задали в *ПолучитьНастройкиДополнительныхКолонокСпискаДокументов*

Параметры:

Элементы структуры **Параметры:**

- **СписокДокументов** - табличная часть «СписокДокументов» в текущем состоянии
- **Режим** - текущий режим отображения документов

Режим принимает значения:

- ОтправкаПакетов
- ОтправленныеДокументы
- ПолученныеДокументы
- ПолученныеПакеты

Пример использования: *Как доработать таблицу списка документов*

2.3.5 ПриОткрытииФормы

Примечание: Событие для обычных форм

Синтаксис:

ПриОткрытииФормы(<Параметры>)

Описание:

Используется для доступа из подключаемого модуля к свойствам обычной формы в целом, а также к коллекции ее элементов управления.

Параметры:

Элементы структуры **Параметры:**

- **Форма** - обычная форма основного модуля.

Пример использования: *Как доработать таблицу списка документов*

2.3.6 ПриПолученииДанныхТабличнойЧасти

Примечание: Событие для обычных форм

Синтаксис:

ПриПолученииДанныхТабличнойЧасти(<Параметры>)

Описание:

Используется для доступа из подключаемого модуля к свойствам табличной части обычной формы.

Параметры:

Элементы структуры **Параметры:**

- **Форма** - обычная форма основного модуля
- **ИмяФормы** - название формы
- **Элемент** - табличное поле формы

- **ОформленияСтрок** - коллекция элементов оформления строки табличного документа

Пример использования: *Как доработать таблицу списка документов*

2.3.7 УстановитьУсловноеОформление

Примечание: Событие для управляемых форм

Синтаксис:

УстановитьУсловноеОформление(<Параметры>)

Описание:

Используется для доступа из подключаемого модуля к свойствам управляемой формы в целом, а также к коллекции ее элементов управления.

Параметры:

Элементы структуры **Параметры**:

- **Форма** - управляемая форма основного модуля.

Пример использования: *Как доработать таблицу списка документов*

2.4 Прочие функции

2.4.1 ВыполнитьРегламентныеДействия

Синтаксис:

ВыполнитьРегламентныеДействия(<Параметры>)

Возвращает:

Любое значение

Описание:

Выполняет регламентные действия, требующие инициализацию основного модуля.

Параметры:

- **Параметры** (тип Структура) - структура, элементы которой будут использоваться в функции. По умолчанию в значении Неопределено.

Пример использования: *Регламентное задание*

2.4.2 ОбработатьСобытие

Синтаксис:

ОбработатьСобытие(<ИмяСобытия>, <Параметры>)

Возвращает:

Значение вызываемой функции

Описание:

Определяется, какую функцию вызвать в зависимости от имени события.

Параметры:

- **ИмяСобытия** (тип Строка) - название события, которое требуется обработать.
- **Параметры** (тип Структура) - структура, элементы которой будут использоваться в функции-обработчике конкретного события.

2.4.3 ОпределитьТипЗначенияОбъекта

Синтаксис:

ОпределитьТипЗначенияОбъекта(<Параметры>)

Возвращает:

Строку с названием ссылочного типа

Описание:

Позволяет определить тип используемых справочников Организации и Контрагенты.

Параметры:

Элементы структуры **Параметры:**

- **Синоним** - текстовое значение: «Организации» или «Контрагенты»

Пример использования: *Дополнительные возможности*

2.4.4 ПолучитьКоличествоДокументовЗаПериод

Синтаксис:

ПолучитьКоличествоДокументовЗаПериод(<Параметры>)

Возвращает:

Результат запроса: таблица с колонками «Связь1» и «ДокументовЗаПериод»

Описание:

Выполняет запрос по расчету количества накладных за период. Используется в форме списка контрагентов в режиме «Поиск и приглашение».

Параметры:

Элементы структуры **Параметры:**

- **МассивКонтрагентов** - массив ссылок на элементы справочника контрагентов 1С, которые отобраны для списка на приглашение
- **Организация** - ссылка на организацию 1С, для которой отбираются контрагенты на приглашение

Пример использования: *Дополнительные возможности*

2.4.5 ПослеОбработкиЛентыСобытий

Синтаксис:

ПослеОбработкиЛентыСобытий(<Параметры>)

Описание:

Используется для выполнения методов API для новых/измененных документов при чтении ленты событий.

Параметры:

Элементы структуры **Параметры**:

- **ПараметрыСервиса** - структура, содержащая параметры, необходимые для авторизации на сервере Диадока (токен, настройки прокси)
- **ИдентификаторыДокументов** - таблица документов, по которым были изменения (статуса, перемещение в подразделение и т.д.). Состоит из идентификаторов документа (BoxId, LetterId, DocumentId) и ссылки на элемент справочника Диадок_Документы.

Пример использования: *Дополнительные возможности*

2.4.6 ПослеУстановкиСтатусаДокумента

Синтаксис:

ПослеУстановкиСтатусаДокумента(<Параметры>)

Описание:

Позволяет внести изменения в документ 1С при изменении статуса.

Параметры:

Элементы структуры **Параметры**:

- **ДокументДД** - ссылка на справочник «Диадок_Документы», содержащий описание электронного документа, у которого изменился статус
- **Статус** - наименование нового статуса документа

Пример использования: *Дополнительные возможности*

3.1 Структура «Пакет»

2 версии структуры:

- *Пакет (начиная с версии 4.0)*
- *Пакет (до версии 4.0)*

3.1.1 Пакет (начиная с версии 4.0)

Пакет - структура, содержащая сведения о пакете документов. На текущий момент такая структура используется внутри модуля. В подключаемом модуле приходит дополненным до предыдущей версии структуры.

Элементы:

- **ДатаЭДО** (тип Дата) - дата создания пакета
- **ДокументУчета** (тип ДокументСсылка, СправочникСсылка) - ссылка на документ в 1С, который является основанием для данного пакета
- **Документы** (тип Массив) - массив *структур документов*, входящих в состав пакета
- **ДопРеквизиты** (тип Структура) - структура дополнительных данных любого типа, которые могут быть использованы в событие *ПодготовкиПакета* и после. Допустимо использование не более 5 реквизитов
- **Заблокирован** (тип Булево) - признак того, что пакет заблокированный. В таком пакете любая операция применяется ко всем документам сразу
- **ИдентификаторВида** (тип Строка 100) - ID пакета из события *ПолучитьТаблицуИспользуемыхПакетов*
- **Идентификаторы** (тип Структура) - структура идентификаторов на сервере Диодока, характеризующих пакет

- **Направление** (тип Строка) - принимает значение «Исходящее» / «Входящее»
- **Отправитель** (тип Структура) - структура, описывающая участника ЭДО, от которого был отправлен пакет
- **ПараметрыШаблона** (тип Структура) - структура идентификаторы на сервере Диадока, характеризующих пакет-шаблон
- **Получатель** (тип Структура) - структура, описывающая участника ЭДО, который получил пакет
- **Реквизиты** (тип Структура) - структура основных реквизитов. Для исходящих: реквизиты документа-основания пакета. Для входящих: реквизиты первого полученного документа пакета
- **СлужебнаяИнформация** (тип Структура) - вспомогательная структура, содержащая временные переменные
- **Состояние** (тип Строка) - принимает значение «Подготовлен» / «Отправлен» / «ЗагруженИзДиадок» / «Получен» / «Переотправлен»
- **Шаблон** (тип Булево) - признак того, что это пакет-шаблон

Структура **Идентификаторы**:

- **BoxId** (тип Строка 36) - идентификатор ящика организации
- **LetterId** (тип Строка 36) - идентификатор пакета Диадока в ящике организации
- **ПакетЭДО** (тип СправочникСсылка.Диадок_ПакетыДокументов) - ссылка на описание электронного пакета в базе 1С

Структура **Отправитель / Получатель**:

- **BoxId** (тип Строка) - идентификатор ящика отправителя/получателя пакета на сервере Диадока
- **ОрганизацияЭДО** (тип СправочникСсылка.Диадок_ЮрФизЛица) - ссылка на организацию / контрагента в 1С
- **Организация** (тип СправочникСсылка) - ссылка на сопоставленную организацию / контрагента в базе 1С
- **FNSParticipantId** (тип Строка 47) - ФНС идентификатор участника электронного документооборота
- **DepartmentId** (тип Строка 50) - идентификатор подразделения участника электронного документооборота

Структура **ПараметрыШаблона**:

- **MessageFromBoxId** (тип Строка) - идентификатор ящика отправителя документов, созданных на основе шаблонов
- **MessageToBoxId** (тип Строка) - идентификатор ящика получателя документов, созданных на основе шаблонов
- **MessageToDepartmentId** (тип Строка) - идентификатор подразделения получателя сообщения, которое будет создано на основе отправляемого шаблона
- **LockMode** (тип Строка) - режим блокировки сообщения с шаблонами, доступны значения:
 - *None* - блокировка не производится
 - *Send* - для документов пакета возможна только совместная отправка. Свойства залоченности работают по аналогии с закрытыми пакетами (режимом Full) и сохраняются до отправки пакета. После отправки пакета документов все ограничения на поддокументную обработку снимаются.

- *Full* - документы сообщения будут отправлены закрытым пакетом. В таком пакете любая операция применяется ко всем документам сразу.

Структура **Реквизиты**:

- **Дата** (тип Дата) - дата документа-основания исходящего пакета / дата первого документа входящего пакета
- **Номер** (тип Строка 30) - номер документа-основания исходящего пакета / номер первого документа входящего пакета
- **Сумма** (тип Число 15,2) - сумма документа-основания исходящего пакета / сумма первого документа входящего пакета

Структура **СлужебнаяИнформация**:

- **ОтложеннаяОтправка** (тип Булево) - признак того, что пакет сохранен на сервере Диадока без отправки контрагенту
- **ЗашифрованныйПакет** (тип Булево) - признак того, что пакет содержит зашифрованные документы

3.1.2 Пакет (до версии 4.0)

Пакет - структура, содержащая сведения о пакете документов. Создание пустой структуры происходит в функции основного модуля ЭДО_ПакетДокументовМенеджер_СоздатьНовый().

Элементы:

- **Данные1С** (тип Структура) - структура Свойства1С, содержащая общие поля из 1С для всех документов пакета.
- **Документы** (тип Массив) - массив структур для хранения сведений о каждом документе пакета: описание контента, тип документа, ссылка на документ в 1С и т.п.
- **ДополнительныеРеквизиты** (тип Структура) - структура дополнительных данных любого типа, которые могут быть использованы в событие ПодготовкиПакета и после. Допустимо использование не более 5 реквизитов.

Структура **Свойства1С**:

- **Организация** - ссылка на организацию в 1С, от лица которой сформирован пакет.
- **Контрагент** - ссылка на контрагента в 1С, которому предназначен пакет.
- **ПодразделениеОрганизации** - ссылка на организацию в 1С, являющуюся обособленным подразделением головной организации, от лица которой сформирован пакет.
- **ПодразделениеКонтрагента** - ссылка на подразделение контрагента в 1С, которому предназначен пакет.
- **ЭтоВнутреннийДокумент** - признак пакета с внутренними документами.
- **Ссылка** - ссылка на пакет в 1С.
- **ВидПакета** - ссылка на вид пакета в 1С.
- **ВидПакетаID** - внутренний идентификатор вида пакета.
- **Документ** - ссылка на документ в 1С, который является основным для данного пакета.
- **НомерДокумента** - номер документа в 1С.
- **ДатаДокумента** - дата документа в 1С.

- **СуммаДокумента** - сумма документа в 1С.
- **ПереотправляемыйПакетСсылка** - ссылка на переотправляемый пакет в 1С.
- **ПараметрыШаблона** - содержит поля:
 - **MessageFromBoxId** - идентификатор ящика отправителя документов, созданных на основе шаблонов.
 - **MessageToBoxId** - идентификатор ящика получателя документов, созданных на основе шаблонов.
 - **MessageToDepartmentId** - идентификатор подразделения получателя сообщения, которое будет создано на основе отправляемого шаблона.
 - **LockMode** - режим блокировки сообщения с шаблонами, доступны значения:
 - * *None* - блокировка не производится
 - * *Send* - для документов пакета возможна только совместная отправка. Свойства залоченности работают по аналогии с закрытыми пакетами (режимом Full) и сохраняются до отправки пакета. После отправки пакета документов все ограничения на поддокументную обработку снимаются.
 - * *Full* - документы сообщения будут отправлены закрытым пакетом. В таком пакете любая операция применяется ко всем документам сразу.

3.2 Структура «Документ»

Используется в модуле с версии 4.0. На текущий момент доступна в ряде событие подключаемого модуля:

- *ВыполнитьТребуемоеДействие*
- *ПровестиАнализДокумента*
- *НайтиСопоставлениеДокумента*
- *ПолучитьЗаполненныйОбъектДокумента1С*

Элементы:

- **ДатаЭДО** (тип Дата) - дата создания документа на сервере Диадока
- **ДокументУчета** (тип ДокументСсылка, СправочникСсылка) - ссылка на документ в 1С, который сопоставлен с данным электронным документом
- **Документооборот** (тип Структура) - содержит структуры, содержащие описание контента, титул покупателя и т.п. Используется внутри модуля
- **ДопРеквизиты** (тип Структура) - дополнительных данных любого типа. Допустимо использование не более 5 реквизитов
- **Заблокирован** (тип Булево) - признак того, что документ находится в составе заблокированного пакета. В таком пакете любая операция применяется ко всем документам сразу
- **ИдентификаторВида** (тип Строка) - ID документа из события *ПолучитьТаблицуИспользуемыхВидовДокументов*
- **ИдентификаторВидаПакета** (тип Строка) - ID пакета из события *ПолучитьТаблицуИспользуемыхПакетов*

- **Идентификаторы** (тип Структура) - идентификаторы на сервере Диадока, характеризующие документ
- **ИмяФайла** (тип Строка) - имя файла из метаданных документа
- **Ключ** (тип Строка) - уникальный идентификатор структуры документа
- **Метаданные** (тип Структура) - метаданных документа. Состав зависит от *типа контента* документа
- **Направление** (тип Строка) - принимает значение «Исходящее» / «Входящее»
- **Отправитель** (тип Структура) - структура, описывающая участника ЭДО, от которого был отправлен пакет
- **ПараметрыШаблона** (тип Структура) - идентификаторы на сервере Диадока, характеризующие документ из пакета-шаблона
- **Получатель** (тип Структура) - структура, описывающая участника ЭДО, который получил пакет
- **РезультатАнализа** (тип Структура) - содержит итоги анализа, выполненного в событии *ПровестиАнализДокумента* или *ПослеАнализаПакета*
- **Реквизиты** (тип Структура) - основные реквизитов документа
- **Связи** (тип Структура) - идентификаторы документов из других пакетов, которые связаны с текущим документом
- **СлужебнаяИнформация** (тип Структура) - вспомогательная структура, содержащая временные переменные
- **Состояние** (тип Строка) - метка документа. Принимает значение «Подготовлен» / «Отправлен» / «ЗагруженИзДиадок» / «Получен» / «Переотправлен»
- **Статусы** (тип Структура) - состоит из строкового представления основного и дополнительного статуса документа
- **Тип** (тип Структура) - описывает тип документа
- **Шаблон** (тип Булево) - признак того, что документ находится в составе пакета-шаблона

Структура **Идентификаторы**:

- **BoxId** (тип Строка 36) - идентификатор ящика организации на сервере Диадока
- **LetterId** (тип Строка 36) - идентификатор пакета в ящике организации на сервере Диадока
- **DocumentId** (тип Строка 36) - идентификатор документа в ящике организации на сервере Диадока
- **ДокументЭДО** (тип СправочникСсылка.Диадок_Документы) - ссылка на описание электронного документа в базе 1С
- **ПакетЭДО** (тип СправочникСсылка.Диадок_ПакетыДокументов) - ссылка на описание электронного пакета в базе 1С

Структура **Отправитель / Получатель**:

- **BoxId** (тип Строка) - идентификатор ящика отправителя/получателя пакета на сервере Диадока
- **ОрганизацияЭДО** (тип СправочникСсылка.Диадок_ЮрФизЛица) - ссылка на организацию / контрагента в 1С
- **Организация** (тип СправочникСсылка) - ссылка на сопоставленную организацию / контрагента в базе 1С

- **FNSParticipantId** (тип Строка 47) - ФНС идентификатор участника электронного документооборота
- **DepartmentId** (тип Строка 50) - идентификатор подразделения участника электронного документооборота

Структура **ПараметрыШаблона**:

- **MessageFromBoxId** (тип Строка) - идентификатор ящика отправителя документов, созданных на основе шаблонов
- **MessageToBoxId** (тип Строка) - идентификатор ящика получателя документов, созданных на основе шаблонов
- **MessageToDepartmentId** (тип Строка) - идентификатор подразделения получателя сообщения, которое будет создано на основе отправляемого шаблона
- **EditingSettingId** (тип Строка) - идентификатор настройки редактирования содержимого документа
- **LockMode** (тип Строка) - режим блокировки сообщения с шаблонами, доступны значения:
 - *None* - блокировка не производится
 - *Send* - для документов пакета возможна только совместная отправка. Свойства залоченности работают по аналогии с закрытыми пакетами (режимом Full) и сохраняются до отправки пакета. После отправки пакета документов все ограничения на поддокументную обработку снимаются.
 - *Full* - документы сообщения будут отправлены закрытым пакетом. В таком пакете любая операция применяется ко всем документам сразу.

Структура **РезультатАнализа**:

- **АнализПроведен** (тип Булево) - признак завершения анализа документа
- **ТекстОшибки** (тип Строка) - любое строковое значение ошибки при выполнении анализа. Определяется в подключаемом модуле. Отображается в колонке «Ошибки проверки» в списке документов. Видимость ограничена 150 символами.
- **ТребуемоеДействиеВ1С** (тип Строка 50) - строковое представление действия в 1С после анализа. Обрабатывается в событии *ВыполнитьТребуемоеДействие*
- **ТребуемоеДействиеВДД** (тип Строка 50) - строковое представление действия для отправки на сервер Диадока после анализа. Обрабатывается при нажатии на кнопку на форме «Отправить готовые ответы». Возможные варианты:
 - *Подписать*
 - *ОтказатьВПодписи*
 - *ЗапроситьУточнение*
 - *Аннулировать*
 - *ОтказатьВАннулировании*

Структура **Реквизиты**:

- **Дата** (тип Дата) - дата электронного документа из метаданных (DocumentDate)
- **Номер** (тип Строка 30) - номер электронного документа из метаданных (DocumentNumber)
- **Сумма** (тип Число 15,2) - сумма электронного документа из метаданных (TotalSum)
- **СуммаНДС** (тип Число 15,2) - сумма НДС электронного документа из метаданных (TotalVat)

- **КодВалюты** (тип Строка 4) - код валюты электронного документа из метаданных (CurrencyCode)

Структура **Связи**:

- **InitialDocumentIds** (тип Массив) - идентификаторы документов, на которые ссылается данный документ
- **SubordinateDocumentIds** (тип Массив) - идентификаторы документов, которые ссылаются на данный документ
- **TransformedToLetterIds** (тип Массив) - идентификаторы пакетов, созданных на основе данного шаблона и содержащих данный документ
- **PacketId** (тип Строка 36) - идентификатор пакета Диадока, в котором в данный момент находится документ
- **CustomDocumentId** (тип Строка) - необязательный идентификатор документа во внешней системе; используется для выстраивания связей между документами внутри отправляемого сообщения
- **OriginMessageId** (тип Строка) - идентификатор исходного шаблона

Структура **Тип**:

- **ТипДокумента** (тип Строка) - название *типа документа*
- **ТипКонтента** (тип Строка) - название *типа контента*
- **ФункцияДокумента** (тип Строка) - для документов в формате УПД. Принимает значение InvoiceAndBasic / Invoice / Basic
- **Представление** (тип Строка) - представление типа документа на русском языке из списка «*Типы документа*»
- **Формализованный** (тип Булево) - признак того, что документ в формате, утвержденном ФНС

4.1 Типы документов

Поддерживаются следующие типы документов:

- **XmlTorg12** - товарная накладная ТОРГ-12 в XML-формате
- **XmlAcceptanceCertificate** - акт о выполнении работ / оказании услуг в XML-формате
- **Invoice** - счет-фактура
- **InvoiceRevision** - исправление счета-фактуры
- **InvoiceCorrection** - корректировочный счет-фактура
- **InvoiceCorrectionRevision** - исправление корректировочного счета-фактуры
- **UniversalTransferDocument** - универсальный передаточный документ
- **UniversalTransferDocumentRevision** - универсальный передаточный документ (исправление)
- **UniversalCorrectionDocument** - универсальный корректировочный документ
- **UniversalCorrectionDocumentRevision** - универсальный корректировочный документ (исправление)
- **ProformaInvoice** - счет на оплату
- **Nonformalized** - неформализованный документ
- **TrustConnectionRequest** - запрос на инициацию канала обмена документами через Диадок
- **Torg12** - товарная накладная ТОРГ-12 (неформализованная)
- **AcceptanceCertificate** - акт о выполнении работ / оказании услуг (неформализованный)
- **PriceList** - ценовой лист
- **PriceListAgreement** - протокол согласования цены
- **CertificateRegistry** - реестр сертификатов

- **ReconciliationAct** - акт сверки
- **Contract** - договор
- **Torg13** - накладная ТОРГ-13
- **ServiceDetails** - детализация
- **SupplementaryAgreement** - дополнительное соглашение к договору
- **MesNotification** - уведомление о переходе на ЭДО
- **StorageInventoryAcceptanceCertificate** - акт МХ-1
- **ReturnInventoryAcceptanceCertificate** - акт МХ-3
- **Torg1** - акт о приемке товаров ТОРГ-1
- **Torg2** - акт об установленном расхождении ТОРГ-2
- **ForwarderAssignment** - поручение экспедитору
- **PerformedWorkAcceptanceCertificate** - акт приемки выполненных работ КС-2
- **PerformedWorkCostCertificate** - справка о стоимости выполненных работ и затрат КС-3

Список настраивается для конкретного ящика. Доступные типы вернет метод API [GetDocumentTypes](#). Автоматическая загрузка таких типов документов происходит при синхронизации организаций или контрагентов.

Сохраняются в справочник «Диадок_ДополнительныеСправочники» с именем справочника «Типы-ДокументовAPI».

4.2 Типы контента

Используемые обозначения типов контента, определяющих формат ФНС:

- **torg12** - накладная в формате приказа №172
- **act** - акт в формате приказа №172
- **invoice** - счет-фактура в формате приказа №93
- **invoicecor** - корректировочная счет-фактура в формате приказа №93
- **utd** - УПД в формате приказа №155
- **utd820** - УПД в формате приказа №820
- **ucd** - УКД в формате приказа №189
- **tovtorg** - накладная в формате приказа №551
- **rezru** - акт в формате приказа №552
- **<пустая строка>** - неформализованный документ

Структуры для заполнения контента УПД в формате приказа ММВ-7-15/820@

Примечание: Поддержка осуществляется в модуле Pro с версии 3.2.0.

5.1 Титул продавца Utd820SellerContent

Содержание титула продавца *Универсального передаточного документа* в формате приказа ММВ-7-15/820@

Свойства

Function Строка (1-6) - функция (СЧФ/СЧФДОП/ДОП) [Функция]

ApprovedStructureAdditionalInfoFields Строка (=14) - информация о наличии согласованной структуры дополнительных информационных полей [СоглСтрДопИнф]

DocumentName Строка (1-255) - наименование первичного документа, определенное организацией [НаимДокОпр]

DocumentDate Дата (ДД.ММ.ГГГГ) - дата составления (выписки) счета-фактуры [ДатаСчФ]

DocumentNumber Строка (1-1000) - порядковый номер счета-фактуры [НомерСчФ]

Currency Строка (=3) - код валюты [КодОКВ]

CurrencyRate Строка - курс валюты [КурсВал]

RevisionDate Дата (ДД.ММ.ГГГГ) - дата исправления [ДатаИспрСчФ]

- HyphenRevisionDate** Булево - признак того, что в графе «Исправление» не указана дата исправленного счета-фактуры. Если Истина, то будет указано значение «-» в поле [ДефДатаИспрСчФ]
- RevisionNumber** Число (3) - номер исправления [НомИспрСчФ]
- HyphenRevisionNumber** Булево - признак того, что в графе «Исправление» не указан номер исправленного счета-фактуры. Если Истина, то будет указано значение «-» в поле [ДефНомИспрСчФ]
- DocumentCreator** Строка (1-1000) - составитель файла обмена счета-фактуры (информации продавца) [НаимЭконСубСост]
- DocumentCreatorBase** Строка (1-120) - основание, по которому экономический субъект является составителем файла обмена счета-фактуры (информации продавца) [ОснДоверОргСост]
- GovernmentContractInfo** Строка (20-25) - Идентификатор государственного контракта, договора [ИдГосКон]
- CircumFormatInvoice** Строка (=1) - обстоятельства формирования счета-фактуры, применяемого при расчетах по налогу на добавленную стоимость [ОбстФормСчФ]
- Sellers** Коллекция *ExtendedOrganizationInfo* - сведения о продавце [СвПрод]
- Buyers** Коллекция *ExtendedOrganizationInfo* - сведения о покупателе [СвПокуп]
- Shippers** Коллекция *Shipper* - сведения о грузоотправителе [ГрузОт]
- Consignees** Коллекция *ExtendedOrganizationInfo* - грузополучатель и его адрес [ГрузПолуч]
- Signers** Коллекция *ExtendedSigner* - подписанты документа [Подписант]
- PaymentDocuments** Коллекция *PaymentDocumentInfo* - сведения о платежно-расчетном документе [СвПРД]
- AdditionalInfoId** *AdditionalInfoId* - информационное поле документа [ИнфПолФХЖ1]
- Table** *Table* - сведения таблицы счета-фактуры [ТаблСчФакт]
- TransferInfo** *TransferInfo* - сведения о передаче (сдаче) [СвПер]
- FactorInfo** *ExtendedOrganizationInfo* - сведения о факторе [СвФактор]
- MainAssignMonetaryClaim** *TransferBase* - основание уступки денежного требования [ОснУстДенТреб]
- SellerInfoCircumPublicProc** *SellerInfoCircumPublicProc* - информация продавца об обстоятельствах закупок для государственных и муниципальных нужд [ИнфПродГосЗакКазн]
- DocumentShipments** Коллекция *DocumentShipment* - реквизиты документа, подтверждающего отгрузку товаров (работ, услуг, имущественных прав) [ДокПодтвОтгр]

5.2 Титул покупателя Utd820BuyerContent

Содержание титула покупателя *Универсального передаточного документа* в формате приказа ММВ-7-15/820@

Свойства

- DocumentCreator** Строка (1-1000) - составитель файла обмена (информации покупателя) [НаимЭконСубСост]
- DocumentCreatorBase** Строка (1-120) - основание, по которому экономический субъект является составителем файла обмена (информации покупателя) [ОснДоверОргСост]
- OperationCode** Строка (1-255) - вид операции [ВидОперации]
- OperationContent** Строка (1-255) - содержание операции (текст) [СодОпер]
- AcceptanceDate** Дата (ДД.ММ.ГГГГ) - дата принятия товаров (результатов выполненных работ), имущественных прав (подтверждения факта оказания услуг) [ДатаПрин]
- Employee** *Employee* - работник организации покупателя [РабОргПок]
- OtherIssuer** *OtherIssuer* - иное лицо [ИнЛицо]
- AdditionalInfoId** *AdditionalInfoId* - информационное поле документа [ИнфПолФХЖ4]
- ContentOperCode** *ContentOperCode* - код содержания операции [КодСодОпер]
- BuyerInfoCircumPublicProc** *BuyerInfoCircumPublicProc* - информация покупателя об обстоятельствах закупок для государственных и муниципальных нужд [ИнфПокГосЗакКазн]
- Signers** Коллекция *ExtendedSigner* - подписанты документа [Подписант]

5.3 AdditionalInfoId

Информационное поле факта хозяйственной жизни (ИнфПолФХЖ)

Свойства

- InfoFileId** Строка (36) - идентификатор файла информационного поля [ИдФайлИнфПол]
- StructedAdditionalInfos** Коллекция *StructedAdditionalInfo* - текстовая информация [ТекстИнф]

5.4 AddressInfo

Сведения об адресе (АдресТип)

Свойства

- IsForeign** Булево - признак того, что адрес является иностранным (за пределами РФ). Если установлен признак Истина, тогда будет заполнен [АдрРФ]
- ZipCode** Строка (=6) - индекс [Индекс]
- RegionCode** Строка (=2) - код региона РФ [КодРегион]
- Territory** Строка (1-50) - район [Район]
- City** Строка (1-50) - город [Город]

Locality Строка (1-50) - населенный пункт [НаселПункт]
Street Строка (1-50) - улица [Улица]
Building Строка (1-20) - дом [Дом]
Block Строка (1-20) - корпус [Корпус]
Apartment Строка (1-20) - квартира [Кварт]
CountryCode Строка (=3) - код страны (для адреса за пределами РФ) [КодСтр]
AddressText Строка (1-255) - строка адреса [АдрТекст]
AddressCode Строка (1-36) - уникальный номер адреса объекта адресации в государственном адресном реестре [КодГАР]

5.5 BuyerInfoCircumPublicProc

Информация покупателя об обстоятельствах закупок для государственных и муниципальных нужд (для учета Федеральным казначейством денежных обязательств) (ИнфПокГосЗакКазн)

Свойства

ProcCode Строка (1-36) - идентификационный код закупки [ИдКодЗак]
PersonalAccountBuyer Строка (=11) - номер лицевого счета покупателя [ЛицСчетПок]
NameFinAuthority Строка (1-2000) - наименование финансового органа покупателя [НаимФинОргПок]
BuyerRegistryEntryNumber Строка (=8) - номер реестровой записи покупателя [НомРеестрЗапПок]
BuyerLiabNumber Строка (16-19) - учетный номер бюджетного обязательства покупателя [УчНомБюдОбязПок]
BuyerTreasuryCode Строка (=4) - код территориального органа Федерального казначейства покупателя [КодКазначПок]
SellerTreasuryName Строка (1-2000) - наименование территориального органа Федерального казначейства покупателя [НаимКазначПок]
ОКТМОBuy Строка (8-11) - код покупателя в Общероссийском классификаторе территорий муниципальных образований [ОКТМОПок]
ОКТМОPlaceDelivery Строка (8-11) - код места поставки в Общероссийском классификаторе территорий муниципальных образований [ОКТМОМесПост]
PayDeadLine Дата (ДД.ММ.ГГГГ) - предельная дата оплаты [ДатаОплПред]
NumberFundsLiab Строка (=22) - учетный номер денежного обязательства [УчНомДенОбяз]
PriorityPayment Строка (=1) - очередность платежа [ОчерПлат]
TypePayment Строка (=1) - вид платежа [ВидПлат]
InfoFundsLiab Коллекция *InfoFundsLiab* - информация для сведений о денежном обязательстве [ИнфСведДенОбяз]

5.6 ContentOperCode

Код содержания операции (КодСодОпер)

Свойства

TotalCode Строка (=1) - код, обозначающий итог приемки товара (работ, услуг, прав) [КодИтога]

NameDiscrepDocument Строка (1-255) - наименование документа, оформляющего расхождения [НаимДокРасх]

TypeDiscrepDocument Строка (=1) - код вида документа о расхождениях [ВидДокРасх]

NumberDiscrepDocument Строка (1-255) - номер документа покупателя о расхождениях [НомДокРасх]

DateDiscrepDocument Дата (ДД.ММ.ГГГГ) - дата документа о расхождениях [ДатаДокРасх]

IdDiscrepDocument Строка (1-255) - идентификатор файла обмена документа о расхождениях, сформированного покупателем [ИдФайлДокРасх]

5.7 CustomDeclaration

Сведения о таможенной декларации (СвТД)

Свойства

CountryCode Строка (=3) - цифровой код страны происхождения товара [КодПроисх]. Если значение не заполнено, то будет установлено значение «-» в поле [ДефКодПроисх]

HyphenCountry Булево - признак того, что цифровой код страны отсутствует. Если Истина, то будет установлено значение «-» в поле [ДефКодПроисх]

DeclarationNumber Строка (1-29) - регистрационный номер таможенной декларации [НомерТД]

5.8 DocumentShipment

Реквизиты документа, подтверждающего отгрузку товаров (работ, услуг, имущественных прав) (ДокПодтвОтгр)

Свойства

Name Строка (1-255) - наименование документа об отгрузке [НаимДокОтгр]

Number Строка (1-255) - номер документа об отгрузке [НомДокОтгр]

Date Дата (ДД.ММ.ГГГГ) - дата документа об отгрузке [ДатаДокОтгр]

5.9 Employee

Работник организации (РабОргПрод)

Свойства

EmployeePosition Строка (1-128) - должность [Должность]

EmployeeInfo Строка (1-255) - иные сведения, идентифицирующие физическое лицо [ИныеСвед]

EmployeeBase Строка (1-120) - основание полномочий (доверия) [ОснПолн]

Surname Строка (1-60) - фамилия [Фамилия]

FirstName Строка (1-60) - имя [Имя]

Patronymic Строка (1-60) - отчество [Отчество]

5.10 ExtendedOrganizationInfo

Сведения об участнике факта хозяйственной жизни (УчастникТип)

Свойства

Name Строка - наименование организации. Для индивидуального предпринимателя наименование задается в формате «Фамилия Имя Отчество» [НаимОрг/ФИО]

Inn Строка (10-12) - ИНН организации [ИННЮЛ]/ [ИННФЛ]. Если не заполнено, то будет указано значение «-» в поле [ДефИННЮЛ]/ [ДефИННФЛ].

NurphenInn Булево - признак того, что ИНН организации не указан. Если Истина, то будет указано значение «-» в поле [ДефИННЮЛ]/ [ДефИННФЛ].

Kpp Строка (=9) - КПП организации [КПП]

Address *AddressInfo* - юридический адрес организации [Адрес]

FnsParticipantId Строка - идентификатор участника ЭДО [ИдОтпр/ИдПол]

Type Строка, чтение/запись - тип организации «LegalEntity» ([СвЮЛУч]), «IndividualEntity» ([СвИП]), «ForeignEntity» ([СвИнНеУч]), «PhysicalEntity» ([СвФЛУчастФХЖ])

Okpro Строка (1-10) - код в общероссийском классификаторе предприятий и организаций [ОКПО]

Phone Строка (1-255) - номер контактного телефона [Тлф]

Email Строка (1-255) - адрес электронной почты [ЭлПочта]

CorrespondentAccount Строка (1-20) - корреспондентский счет банка [КорСчет]

BankAccountNumber Строка (1-20) - номер банковского счета организации [НомерСчета]

BankName Строка (1-1000) - наименование банка [НаимБанк]

BankId Строка (1-9) - БИК банка [БИК]

Department Строка (1-1000) - структурное подразделение [СтруктПодр]

AdditionalInfo Строка (1-255) - информация для участника документооборота [ИнфД-
ляУчаст]

OrganizationOrPersonInfo Строка (1-255) - иные сведения, идентифицирующие физическое или юридическое лицо [ИныеСвед]

IndividualEntityRegistrationCertificate Строка (1-100) - реквизиты свидетельства о государственной регистрации индивидуального предпринимателя [СвГосРегИП]

5.11 ExtendedSigner

Свойства

SignerDetails *ExtendedSignerDetails* - информация о подписанте

5.12 ExtendedSignerDetails

Сведения о лице, подписывающем файл обмена счета-фактуры в электронной форме (Подписант)

Свойства

Surname Строка (1-60) - фамилия [Фамилия]

FirstName Строка (1-60) - имя [Имя]

Patronymic Строка (1-60) - отчество [Отчество]

JobTitle Строка (1-128) - должность [Должн]

Inn Строка (10-12) - ИНН юридического лица или индивидуального предпринимателя [ИННЮЛ]/[ИННФЛ]

RegistrationCertificate Строка - реквизиты свидетельства о государственной регистрации индивидуального предпринимателя, выдавшего доверенность организации на подписание счета-фактуры [ГосРегИПВыдДов]

SignerType Строка - тип подписанта *Возможные значения*

OrganizationName Строка (1-1000) - наименование организации [НаимОрг]

SignerInfo Строка (1-255) - иные сведения, идентифицирующие физическое лицо [ИныеСвед]

Powers Строка - область полномочий *Возможные значения* [ОблПолн].

Status Строка - статус *Возможные значения* [Статус].

PowersBase Строка (1-255) - основания полномочий (доверия) [ОснПолн]

OrganizationPowersBase Строка (1-255) - основания полномочий (доверия) организации [ОснПолнОрг]

Дополнительная информация

Значение <i>SignerType</i>	Описание
LegalEntity	представитель юридического лица ([ЮЛ])
IndividualEntity	индивидуальный предприниматель ([ИП])
PhysicalPerson	физическое лицо ([ФЛ])

Значение <i>Powers</i>	Описание
InvoiceSigner	лицо, ответственное за подписание счетов-фактур
PersonMadeOperation	лицо, совершившее сделку, операцию
MadeAndSignOperation	лицо, совершившее сделку, операцию и ответственное за её оформление
PersonDocumentedOperation	лицо, ответственное за оформление свершившегося события
MadeOperationAndSignedInvoice	лицо, совершившее сделку, операцию и ответственное за подписание счетов-фактур
MadeAndResponsibleForOperationAndSignedInvoice	лицо, совершившее сделку, операцию и ответственное за её оформление и за подписание счетов-фактур
ResponsibleForOperationAndSignedInvoice	лицо, ответственное за оформление свершившегося события и за подписание счетов-фактур

Значение <i>Status</i>	Описание
SellerEmployee	работник организации продавца товаров (работ, услуг, имущественных прав)
InformationCreatorEmployee	работник организации - составителя информации продавца
OtherOrganizationEmployee	работник иной уполномоченной организации
AuthorizedPerson	уполномоченное физическое лицо (в том числе индивидуальный предприниматель)
BuyerEmployee	работник организации покупателя товаров (работ, услуг, имущественных прав)
InformationCreatorBuyerEmployee	работник организации - составителя файла обмена информации покупателя, если составитель файла обмена информации покупателя не является покупателем

5.13 InfoFundsLiab

Информация для сведений о денежном обязательстве (ИнфСведДенОбяз)

Свойства

- NumberString** Число (6) - номер строки таблицы информации продавца [НомСтр]
- ObjectCodeFAIP** Строка (1-24) - код объекта капитального строительства федеральной адресной инвестиционной программы [КодОбъектФАИП]
- TypeFundsSource** Строка (=1) - вид средств [ВидСредств]
- BuyerBudjetClassCode** Строка (=20) - код по бюджетной классификации (покупатель) [КодПокБюджКласс]
- BuyerTargetCode** Строка (1-20) - код цели (покупатель) [КодЦелиПокуп]
- AmountAdvance** Число (19,2) - сумма перечисленного аванса [СумАванс]

5.14 Item

Сведения об отгруженных товарах (о выполненных работах, оказанных услугах), переданных имущественных правах (СведТов)

Свойства

Product Строка (1-1000) - наименование товара (описание выполненных работ, оказанных услуг), имущественных прав [НаимТов]

Unit Строка (3-4) - код единицы измерения [ОКЕИ_Тов]

HyphenUnit Булево - признак того, что код единицы измерения не указан. Если Истина, то будет установлено значение «-» в поле [ДефОКЕИ_Тов]

UnitName Строка (1-255) - наименование единицы измерения [НаимЕдИзм]

Quantity Число (26,11) - количество (объем) [КолТов]

Price Число (26,11) - цена (тариф) за единицу измерения [ЦенаТов]

Excise Число (19,2) - сумма акциза [СумАкциз]. Если не указано значение, тогда будет заполнено поле [БезАкциз].

TaxRate Строка (1-35) - налоговая ставка, принимает одно из значений [НалСт]

SubtotalWithVatExcluded Число (19,2) - стоимость товаров (работ, услуг), имущественных прав без налога - всего [СтТовБезНДС]

Vat Число (19,2) - сумма налога, предъявляемая покупателю [СумНал].

HyphenVat Булево - признак того, что сумма налога не указана. Если Истина, то будет установлено значение «-» в поле [ДефНДС].

WithoutVat Строка - принимает значение либо «false», либо «true». Используется для заполнения [СумНал]. Если указано значение «true», тогда будет заполнено [БезНДС]

Subtotal Число (19,2) - стоимость товаров (работ, услуг), имущественных прав с налогом - всего [СтТовУчНал]. Если не указано значение, тогда будет заполнено поле [ДефСтТовУчНал].

HyphenSubtotal Булево - признак того, что стоимость товаров (работ, услуг) не указана. Если Истина, то будет установлено значение «-» в поле [ДефСтТовУчНал].

ItemMark Строка - принимает значения «Property» (имущество), «Job» (работа), «Service» (услуга), «PropertyRights» (имущественные права), «Other» (иное), «NotSpecified» (Неопределено) [ПрТовРаб]

AdditionalProperty Строка (1-4) - дополнительная информация о признаке [ДопПризн]

ItemVendorCode Строка (1-100) - код товара [КодТов]

ItemToRelease Число (26,11) - заказанное количество (количество надлежит отпустить) [НадлОтп]

ItemCharact Строка (1-1000) - характеристика/описание товара [ХарактерТов]

ItemArticle Строка (1-50) - артикул товара [АртикулТов]

ItemKind Строка (1-10) - сорт товара [СортТов]

CatalogCode Строка (=27) - код каталога [КодКат]

ItemTypeCode Строка (=10) - код вида товара [КодВидТов]

CustomDeclarations Коллекция *CustomDeclaration* - сведения о таможенной декларации [СвТД]

StructedAdditionalInfos Коллекция *StructedAdditionalInfo* - информационное поле факта хозяйственной жизни 2 [ИнфПолФХЖ2]

ItemTracingInfos Коллекция *ItemTracingInfo* - сведения о товаре, подлежащем прослеживаемости [СведПрослеж]

ItemIdentificationNumbers Коллекция *ItemIdentificationNumber* - номер средств идентификации товаров [НомСредИдентТов]

5.15 ItemTracingInfo

Сведения о товаре, подлежащем прослеживаемости (СведПрослеж)

Свойства

RegNumberUnit Строка (1-29) - регистрационный номер партии товаров [НомТовПрослеж]

Unit Строка (3-4) - единица количественного учета товара, используемая в целях осуществления прослеживаемости [ЕдИзмПрослеж]

UnitName Строка (1-255) - наименование единицы количественного учета товара, используемой в целях осуществления прослеживаемости [НаимЕдИзмПрослеж]

Quantity Число (26,11) - количество товара в единицах измерения прослеживаемого товара [КолВЕдПрослеж]

ItemAddInfo Строка (1-255) - дополнительный показатель для идентификации товаров, подлежащих прослеживаемости [ДопПрослеж]

5.16 ItemIdentificationNumber

Номер средств идентификации товаров (НомСредИдентТов)

Свойства

TransPackageId Строка (1-255) - уникальный идентификатор транспортной упаковки [ИдентТрансУпак]

Units Коллекция строк (1-255) - контрольный идентификационный знак [КИЗ]

PackageIds Коллекция строк (1-255) - уникальный идентификатор вторичной (потребительской)/третичной (заводской, транспортной) упаковки [НомУпак]

5.17 OtherIssuer

Иное лицо (ИнЛицо)

Свойства

- EmployeePosition** Строка (1-128) - должность [Должность]
- EmployeeInfo** Строка (1-255) - иные сведения, идентифицирующие физическое лицо [ИныеСвед]
- OrganizationName** Строка (1-255) - наименование организации [НаимОргПрин]
- OrganizationBase** Строка (1-120) - основание, по которому организации доверено принятие товаров (груза) [ОснДоверОргПрин]
- EmployeeBase** Строка (1-120) - основание полномочий представителя организации на принятие товаров (груза) [ОснПолиПредПрин]
- Surname** Строка (1-60) - фамилия [Фамилия]
- FirstName** Строка (1-60) - имя [Имя]
- Patronymic** Строка (1-60) - отчество [Отчество]

5.18 PaymentDocumentInfo

Сведения о платежно-расчетном документе (строка 5 счета-фактуры) (СвПРД)

Свойства

- Number** Строка (1-30) - номер платежно-расчетного документа [НомерПРД]
- Date** Дата (ДД.ММ.ГГГГ) - дата составления платежно-расчетного документа [ДатаПРД]
- Total** Число (19,2) - сумма платежно-расчетного документа [СуммаПРД]

5.19 SellerInfoCircumPublicProc

Информация продавца об обстоятельствах закупок для государственных и муниципальных нужд (для учета Федеральным казначейством денежных обязательств) (ИнфПродГосЗакКазн)

Свойства

- DateStateContract** Дата (ДД.ММ.ГГГГ) - дата государственного контракта [ДатаГосКонт]
- NumberStateContract** Строка (1-100) - номер государственного контракта [НомерГосКонт]
- PersonalAccountSeller** Строка (=11) - номер лицевого счета продавца [ЛицСчетПрод]
- SellerBudgetClassCode** Строка (=20) - код по бюджетной классификации (продавец) [КодПродБюджКласс]
- SellerTargetCode** Строка (=20) - код цели (продавец) [КодЦелиПрод]
- SellerTreasuryCode** Строка (1-4) - код территориального органа Федерального казначейства продавца [КодКазначПрод]

SellerTreasuryName Строка (1-2000) - наименование территориального органа Федерального казначейства продавца [НаимКазначПрод]

5.20 Shipper

Сведения о грузоотправителе (строка 3 счета-фактуры) (ГрузОт)

Свойства

SameAsSeller Булево - указано «он же» [ОнЖе]

OrganizationInfo *ExtendedOrganizationInfo* - грузоотправитель и его адрес [ГрузОтпр]

5.21 StructedAdditionalInfo

Текстовая информация (ТекстИнфТип)

Свойства

Key Строка (1-50) - идентификатор [Идентиф]

Value Строка (1-2000) - значение [Значен]

5.22 Table

Сведения таблицы счета-фактуры (содержание факта хозяйственной жизни 2 - наименование и другая информация об отгруженных товарах (выполненных работах, оказанных услугах), о переданных имущественных правах (ТаблСчФакт)

Свойства

TotalWithVatExcluded Число (19,2) - всего к оплате, стоимость товаров (работ, услуг), имущественных прав без налога - всего [СтТовБезНДСВсего]

Total Число (19,2) - всего к оплате, стоимость товаров (работ, услуг), имущественных прав с налогом - всего [СтТовУчНалВсего]. Если значение не заполнено, то будет установлено значение «-» в поле [ДефСтТовУчНалВсего]

NurphenTotal Булево - признак того, что всего к оплате, стоимость товаров (работ, услуг), имущественных прав с налогом не указана. Если Истина, то будет указано значение «-» в поле [ДефСтТовУчНалВсего]

TotalNet Число (26,11) - количество (масса нетто/количество) - всего по документу [КолНеттоВс]

Vat Число (19,2) - всего к оплате, сумма налога, предъявляемая покупателю [СумНалВсего].

NurphenVat Булево - признак того, что всего к оплате, сумма налога, предъявляемая покупателю не указана. Если Истина, то будет указано значение «-» в поле [ДефНДС]

WithoutVat Строка - принимает значение либо «false», либо «true». Используется для заполнения [СумНалВсего]. Если указано значение «true», тогда будет заполнено [БезНДС]

Items Коллекция *Item* - сведения об отгруженных товарах (о выполненных работах, оказанных услугах), переданных имущественных правах [СведТов]

5.23 TransferBase

Основание отгрузки (передачи, уступки) (ОснованиеТип)

Свойства

DocumentName Строка (1-255) - наименование документа - основания [НаимОсн]

DocumentNumber Строка (1-255) - номер документа - основания [НомОсн]

DocumentDate Дата (ДД.ММ.ГГГГ) - дата документа - основания [ДатаОсн]

DocumentInfo Строка (1-1000) - дополнительные сведения [ДопСвОсн]

BaseDocumentId Строка (1-255) - идентификатор документа - основания [ИдентОсн]

5.24 TransferInfo

Сведения о передаче (сдаче) товаров (результатов работ), имущественных прав (о предъявлении оказанных услуг) (СвПер)

Свойства

OperationInfo Строка (1-255) - содержание операции [СодОпер]

OperationType Строка (1-255) - вид операции [ВидОпер]

TransferDate Дата (ДД.ММ.ГГГГ) - дата отгрузки товаров (передачи результатов работ), передачи имущественных прав (предъявления оказанных услуг) [ДатаПер]

TransferTextInfo Строка (1-1000) - сведения о транспортировке и грузе [СвТранГруз]

Carrier *ExtendedOrganizationInfo* - перевозчик [Перевозчик]

Employee *Employee* - работник организации продавца [РабОргПрод]

OtherIssuer *OtherIssuer* - иное лицо [ИнЛицо]

CreatedThingTransferDate Дата (ДД.ММ.ГГГГ) - дата передачи вещи, изготовленной по договору подряда [ДатаПерВещ]

CreatedThingInfo Строка - сведения о передаче [СвПерВещ]

AdditionalInfoId *AdditionalInfoId* - информационное поле факта хозяйственной жизни 3 [ИнфПолФХЖЗ]

TransferBases Коллекция *TransferBase* - основание отгрузки товаров (передачи результатов работ), передачи имущественных прав (предъявления оказанных услуг) [ОснПер]

Waybills Коллекция *Waybill* - транспортная накладная [ТранНакл]

TransferStartDate Дата (ДД.ММ.ГГГГ) - дата начала периода оказания услуг (выполнения работ, поставки товаров) [ДатаНач]

TransferEndDate Дата (ДД.ММ.ГГГГ) - дата окончания периода оказания услуг (выполнения работ, поставки товаров) [ДатаОкон]

5.25 Waybill

Транспортная накладная (ТранНакл)

Свойства

DocumentNumber Строка (1-255) - номер транспортной накладной [НомТранНакл]

DocumentDate Дата (ДД.ММ.ГГГГ) - дата транспортной накладной [ДатаТранНакл]