

---

# diadoc1cpro-docs Documentation

*Выпуск 1*

Diadoc

янв. 17, 2019



<b>1</b>	<b>Работа подключаемого модуля</b>	<b>3</b>
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
1.2.2	Дополнительная обработка исходящих пакетов . . . . .	10
1.2.3	Как обработать входящий пакет документов . . . . .	10
1.2.3.1	Сопоставление документов . . . . .	11
1.2.3.2	Создание документов . . . . .	11
1.2.3.3	Сопоставление номенклатуры . . . . .	12
1.2.3.4	Анализ документов . . . . .	12
1.2.4	Как доработать таблицу списка документов . . . . .	14
1.2.5	Как добавить кнопки в действия . . . . .	15
1.2.6	Регламентное задание . . . . .	16
1.2.7	Дополнительные возможности . . . . .	17
1.2.7.1	При изменении статуса документа . . . . .	18
1.2.7.2	Переопределение типов Организации и Контрагентов . . . . .	18
1.2.7.3	Фильтр контрагентов для приглашения . . . . .	18
<b>2</b>	<b>Функции подключаемого модуля</b>	<b>21</b>
2.1	Исходящие документы . . . . .	21
2.1.1	ПолучитьТаблицуИспользуемыхПакетов . . . . .	21
2.1.2	ПолучитьТаблицуИспользуемыхВидовДокументов . . . . .	21
2.1.3	Получить текст запроса для списка пакетов на отправку по массиву видов пакетов . . . . .	22
2.1.4	ПодготовитьПакет . . . . .	23
2.1.5	ПодготовитьЭлектронныйДокумент . . . . .	23
2.1.6	ПослеПодготовкиПакета . . . . .	24

2.1.7	ПослеОтправкиПакета . . . . .	24
2.2	Входящие документы . . . . .	24
2.2.1	НайтиСопоставлениеДокумента . . . . .	24
2.2.2	ПолучитьЗаполненныйОбъектДокумента1С . . . . .	25
2.2.3	СопоставитьНоменклатуруПоставщика . . . . .	25
2.2.4	ПровестиАнализДокумента . . . . .	26
2.2.5	ПослеАнализаПакета . . . . .	27
2.2.6	ВыполнитьТребуемоеДействие . . . . .	27
2.3	Дополнение формы модуля . . . . .	28
2.3.1	Получить настройки дополнительных колонок списка документов . . . . .	28
2.3.2	ПослеОбновленияСпискаДокументов . . . . .	28
2.3.3	НажатиеДополнительнойКнопки . . . . .	29
2.3.4	ЗаполнитьСписокДополнительныхКнопок . . . . .	29
2.4	Прочие функции . . . . .	30
2.4.1	ОбработатьСобытие . . . . .	30
2.4.2	ВыполнитьРегламентныеДействия . . . . .	30
2.4.3	ПослеУстановкиСтатусаДокумента . . . . .	30
2.4.4	ОпределитьТипЗначенияОбъекта . . . . .	31
2.4.5	ПолучитьКоличествоДокументовЗаПериод . . . . .	31
<b>3</b>	<b>Основные объекты</b>	<b>33</b>
<b>4</b>	<b>Вспомогательные структуры</b>	<b>35</b>
4.1	Пакет . . . . .	35
4.2	Типы документов . . . . .	36
4.3	Типы контента . . . . .	37

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",
↪?(ФункцияУПД = "ДОП", Неопределено,
↪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».

## 1.2.2 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 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);
```

(continues on next page)

(продолжение с предыдущей страницы)

```

    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;

```

### 1.2.3.3 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;

```

### 1.2.3.4 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 = "xm1torg12"
  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

```

(continues on next page)

(продолжение с предыдущей страницы)

```

Документ1CОбъект = ДанныеДокумента.ДокументВ1С.ПолучитьОбъект();
Документ1CОбъект.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.

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

    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;
    КонечЦикла;

    Возврат ПолучитьНастройкиДополнительныхКолонокСпискаДокументов(Новый Структура("Режим",
    |Режим));

КонечЕсли;

```

### 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
- использование электронной подписи 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

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
    ОстатокСобытийВЛенте = 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".
ТаблицаДокументов = Новый ТаблицаЗначений;

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

## 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 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();

КонецЕсли;

Возврат Истина;
```

### 1.2.7.2 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;
```

### 1.2.7.3 Фильтр контрагентов для приглашения

В форме списка контрагентов модуля есть возможность оправить 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
 |ИЗ
 | ВТ_Документы КАК ВТ_Документы
 |
 |СГРУППИРОВАТЬ ПО
 | ВТ_Документы.Связь1";
  
```



## 2.1 Исходящие документы

### 2.1.1 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).

Пример использования: *Как подготовить пакет документов для отправки*

### 2.1.2 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*

### 2.1.3 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*

#### 2.1.4 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*

#### 2.1.5 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*

### 2.1.6 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*

### 2.1.7 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*

## 2.2 Входящие документы

### 2.2.1 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*

## 2.2.2 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 Диалока.

**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 Диалока

Пример использования: *Как обработать входящий пакет документов*

## 2.2.3 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*

## 2.2.4 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 Диадока (Строка). Допустимые варианты - «Подписать», «ОтказатьВПодписи», «ЗапроситьУточнение»
- **ТипОшибкиВалидации** - произвольный текст, отображаемый в списке, для классификации ошибок
- **КонтентОтвета** - ОбъектХДГО, соответствующий титулу покупателя (для формализованных документов) или отказу в подписи. Именно такой ответ будет отправлен на сервер Диадока, если заполнено поле «ДействиеВДД».

Пример использования: *Как обработать входящий пакет документов*

## 2.2.5 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*

## 2.2.6 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*

## 2.3 Дополнение формы модуля

### 2.3.1 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*

### 2.3.2 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*

### 2.3.3 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*

### 2.3.4 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*

## 2.4 Прочие функции

### 2.4.1 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.

### 2.4.2 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*

### 2.4.3 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*

#### 2.4.4 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*

#### 2.4.5 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*







---

## Вспомогательные структуры

---

### 4.1 Пакет

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.

## 4.2 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».

### 4.3 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
- **ucd** - УКД в формате приказа №189
- **tovtorg** - накладная в формате приказа №551
- **rezru** - акт в формате приказа №552
- **<пустая строка>** - неформализованный документ