



## **Руководство дизайнера форм**

Листов: 33

2022

## АННОТАЦИЯ

В настоящем документе описаны операции по настройке форм документов для работы в веб-клиенте Системы планирования и проведения закупок «Cognitive Lot», а также действия по разрешению ситуаций, которые могут возникнуть в процессе работы.

## СОДЕРЖАНИЕ

1. Глоссарий.....	3
2. Введение .....	4
2.1. Область применения .....	4
2.2. Общие сведения о формах документа.....	4
3. Описание бизнес-процесса работы с формами.....	5
4. Редактирование формы документа .....	6
5. Создание формы документа.....	7
6. Типы данных формы документа .....	9
7. Создание скриптов.....	11
8. Создание хуков.....	13
9. Прикрепление формы документа к потоку .....	29
10. Проверка работоспособности формы документа .....	31

## 1. ГЛОССАРИЙ

В рамках настоящего документа использован набор сокращений и терминов, приведенных в таблицах 1 и 2, соответственно.

Таблица 1 – Сокращения

Сокращение	Расшифровка
БД	База данных
ЕИС	Единая информационная система в сфере закупок
Модуль, модуль «Ошибка! Неизвестное имя свойства документа.»	Программный модуль « <b>Ошибка! Неизвестное имя свойства документа.</b> »
ОЗУ	Оперативное запоминающее устройство, оперативная память
ОС	Операционная система
ПО	Программное обеспечение
Система	Система планирования и управления закупками «Cognitive Lot»

Таблица 2 – Термины

Термин	Определение
<b>Модель содержания</b>	Компонент формы документа, описывающий модель данных этой формы
<b>Скрипт</b>	Программный файл сценария, который в рамках Системы используют для указания определенных условий, при которых нужно применить то или иное свойство для поля (обязательность, скрытие и т. д.) данной формы документа
<b>Статическая форма</b>	Компонент формы документа, описывающий визуальное представление и расположение элементов этой формы
<b>Хук</b>	Файл, который в рамках Системы используют для автоматического выполнения некоторых действий на формах документов (например, для подсчета вычисляемых полей или переноса значений реквизитов из одного документа в другой)
<b>Шаблоны представления данных</b>	Специальные файлы формата xml, определяющие статическую форму

## 2. ВВЕДЕНИЕ

### 2.1. Область применения

Система планирования и проведения закупок «Cognitive Lot» (далее – Система) предназначена для автоматизации процессов планирования и проведения закупок товаров, работ, услуг для обеспечения удовлетворения потребностей в товарах, работах, услугах отдельных видов юридических лиц в соответствии с определенными нормативными документами.

### 2.2. Общие сведения о формах документа

Работа в Системе предполагает регистрацию, просмотр и редактирование документов различных потоков. Отображение каждой формы документа строится на основе данных из ее составных компонентов:

- *модель содержания*. Описывает модель данных формы документа. Создание статической формы и ее публикация на сервере Системы выполняются средствами модуля «**Ошибка! Неизвестное имя свойства документа.**»;

- *статическая форма*. Описывает визуальное представление и расположение элементов формы документа. Статическая форма определяется с помощью ряда специальных *шаблонов представления данных* (далее – шаблоны), представляющих собой файлы формата xml.

При заполнении пользователем полей статической формы в веб-клиенте Системы, осуществляется автоматическая проверка введенных в эти поля значений на соответствие типам данных, выбранному диапазону значений и т. д., после чего введенные пользователем значения присваиваются соответствующим элементам данных модели содержания.

Комплект поставки Системы включает в себя формы документа, соответствующие ее базовой конфигурации. Существует возможность редактировать эти формы, а также создавать новые.

### **3. ОПИСАНИЕ БИЗНЕС-ПРОЦЕССА РАБОТЫ С ФОРМАМИ**

Общий порядок работы при создании новой или редактировании существующей формы документа:

- 1) перейти по ссылке: адрес продуктивного контура/FormDesigner
- 2) выберите действие «Редактировать форму» или «Новая форма»;
- 3) при необходимости создайте или отредактируйте скрипты и хуки;
- 4) при необходимости в модуле «Администратор» прикрепите форму документа к интересующему потоку документов;
- 5) проверьте работоспособность формы документа.

## 4. РЕДАКТИРОВАНИЕ ФОРМЫ ДОКУМЕНТА

Для того чтобы отредактировать форму документа:

- 1) перейти по ссылке: адрес продуктивного контура/FormDesigner;
- 2) нажмите на кнопку «Форма»;
- 3) из отобразившегося списка форм выберите необходимую (см. рис. Рисунок 1);

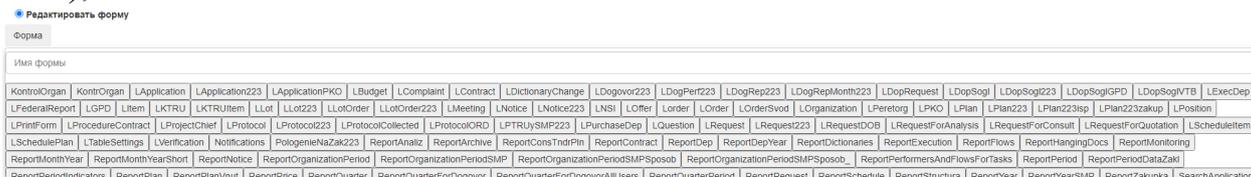


Рисунок 1

Откроется форма документа в формате xml (рис. Рисунок 2):



Рисунок 2

- 4) отредактируйте схему формы документа;
- 5) нажмите на кнопку «Сохранить схему формы»;

*Примечание.* Если необходимо отменить несохраненные изменения нажмите на кнопку «Сбросить текущие изменения».

- 6) при необходимости создайте или отредактируйте скрипты и хуки (см.п. 7 и п. 8);
- 7) проверьте работоспособность формы документа (см. п. 10).

## 5. СОЗДАНИЕ ФОРМЫ ДОКУМЕНТА

Для того чтобы создать форму документа:

- 1) перейти по ссылке: адрес продуктивного контура/FormDesigner;
- 2) выберите **Новая форма** (рис. Рисунок 3);

Рисунок 3

3) введите название формы и в поле **На базе существующей формы** выберите шаблон, на основе которого будет создаваться форма (рис. Рисунок 4):

Рисунок 4

Отобразится форма выбранного шаблона в формате xml (рис. Рисунок 5):

Рисунок 5

- 4) отредактируйте схему формы документа (типы данных формы документа приведены в Таблицае3);
- 5) нажмите на кнопку "Сохранить схему формы";

*Примечание.* Если необходимо отменить несохраненные изменения нажмите на кнопку «Сбросить текущие изменения».

- 6) при необходимости создайте или отредактируйте скрипты и хуки (см. п. 7 и п. 8);
- 7) прикрепите форму документа к потоку (см. п. 9);
- 8) проверьте работоспособность формы документа (см. 10).

## 6. ТИПЫ ДАННЫХ ФОРМЫ ДОКУМЕНТА

В форму шаблона можно добавлять данные нескольких типов (таблица Таблица 3)

Таблица 3 – Типы данных формы документа

Тип данных	Описание	Соответствие на статической форме	Имя по умолчанию на панели отображения модели содержания
Элемент <e>	Простое единичное данное (например, текстовое поле)	Простые поля ввода, раскрывающиеся и комбинированные списки	- text (по умолчанию), longtext; - real; - integer, hour, minute, second, month, year, day, weekday; - boolean; - time; - date; - datetime; - url; - file; - money
Массив <a>	Объединение заранее неизвестного количества однотипных данных (например, список прикрепленных к документу файлов). Массив содержит одну структуру, которая может содержать произвольное количество других структур, массивов и (или) элементов	Таблицы, динамические списковые формы	Отображается в виде типов: <a> по умолчанию – на текущем уровне иерархии; <s> по умолчанию – на втором уровне иерархии; <e> по умолчанию – на третьем уровне иерархии
Структура <s>	Объединение заранее известного количества однотипных или разнотипных элементов, массивов и (или) структур. Последнее позволяет создавать иерархию данных произвольной глубины вложенности	Группировка произвольного количества элементов интерфейса, соответствующих элементам и массивам (поля ввода, списки, таблицы и т. д.)	Корневая структура текущей модели содержания отображается вверху иерархического списка в виде значка 

*Пример.* Структура «Сотрудник» может включать следующие составные части:

- структура «ФИО», состоящая из элементов «Фамилия», «Имя» и «Отчество»;
- элемент «Дата рождения»;
- массив «Образование», состоящий из произвольного количества сгруппированных однотипных данных, соответствующих основному, второму и дополнительному образованиям.

Для того чтобы добавить данное в форму документа укажите соответствующий тег атрибуты. Значения атрибутов указаны в таблице Таблица 4.

В общем случае поле в шаблоне определяется следующим образом:

-<E displayname="Отображаемое название поля" name="Название\_поля" datatype="Тип значения поля" index="true" />.

Таблица 4 – Значения атрибутов поля

Название атрибута	Значение
displayname	Отображаемое название поля
name	Название поля
datatype	<p>Тип значения поля. Возможные значения:</p> <ul style="list-style-type: none"> <li>- Text – текстовое однострочное поле;</li> <li>- LongText – текстовое многострочное поле;</li> <li>- Flow – поток;</li> <li>- AddressBook – элемент адресной книги в Системе</li> <li>- Date – дата;</li> <li>- Time – время;</li> <li>- DateTime – дата и время;</li> <li>- Boolean – логическая переменная;</li> <li>- Money – переменная денежного типа</li> </ul> <p>Тип значения поля должен соответствовать типу, указанному в модели содержания регистрационной карточки</p>
index	Индекс - true

## 7. СОЗДАНИЕ СКРИПТОВ

*Скрипты* используют для указания определенных условий, при которых нужно применить то или иное свойство для поля (обязательность, скрытие и т. д.) данной формы документа. Если для какого-либо поля следует всегда применять определенное свойство, то достаточно указать его в шаблоне регистрационной карточки этой формы.

Файлы скриптов располагаются в папке «..\Scripts\Content\Document\RegCard» и могут быть использованы во всех шаблонах форм. Такой подход позволяет использовать одни и те же общие функции на многих страницах и ускоряет их загрузку. Для написания скриптов используется библиотека jQuery языка JavaScript, обеспечивающая взаимодействие этого языка с языком HTML.

Существует возможность использовать скрипты на языке JavaScript для выполнения нестандартных действий с данными, содержащимися в полях регистрационной карточки, таких как выполнение математических действий над значениями этих полей и т. п. Например, результатом выполнения скрипта может являться отображение в каком-либо поле регистрационной карточки суммы значений нескольких других полей.

Для того чтобы создать новый скрипт:

- 1) в папке «..\Scripts\Content\Document\RegCard» создайте файл скрипта формата js, имя которого должно совпадать с указанным при создании именем соответствующей модели содержания;
- 2) необходимым образом отредактируйте этот файл;
- 3) подключите созданный скрипт к соответствующей форме документа, необходимым образом отредактировав шаблон регистрационной карточки этой формы (см. приложение **Ошибка! Источник ссылки не найден.**).

### Описание содержимого файлов скриптов

Атрибуты Скриптов в шаблоне регистрационной карточки

Блок подключения скриптов:

В блоке подключения скриптов в шаблоне регистрационной карточки именем поля является **script**. Значением атрибута **file** для каждого поля является путь к файлу формата js, содержащему текст скрипта.

Пример части шаблона отображения регистрационной карточки с блоком подключения скриптов:

```
<scripts>  
<script file="Scripts/Content/Document/RegCard/NZapiska.js"/>  
</scripts>
```

Блок дополнений:

Для корректной работы скрипта могут понадобиться дополнительные данные. Эти данные перечисляются в блоке дополнений. Теги **<extensions>** и **</extensions>** указывают, где начинается и где заканчивается перечисление указанных данных.

В блоке дополнений именем поля является **extension**. Значением атрибута **key** для каждого поля является ключевое выражение, предназначенное для формирования дополнительных данных. Данные формируются в формате JSON (текстовом формате обмена данными, в основанном на языке JavaScript).

Примеры ключевых выражений:

- currentUser (формирование данных о текущем пользователе);
- addressbook (формирование данных из адресной книги).
- 

Пример части шаблона отображения регистрационной карточки с блоком дополнений:

```
<extensions>  
<extension key="currentUser" />  
<extension key="addressbook" />  
</extensions>
```

## 8. СОЗДАНИЕ ХУКОВ

Хуки используют для автоматического выполнения некоторых действий на формах документов, например:

- подсчет вычисляемых полей;
- перенос значений реквизитов из одного документа в другой;
- выполнение действий по кнопкам, по нажатию на которые открывается страница регистрации нового документа другого потока на основе текущего документа.

Если для данной формы документа следует использовать шаблон хуков по умолчанию

«..\Templates\Document\Hooks\DocumentLinks.xml»

то создавать новый шаблон хуков не нужно.

Для того чтобы создать новый шаблон хуков:

1) в папке «..\Server\Templates\Document\Hooks\CustomTemplates» создайте файл формата xml, имя которого должно совпадать с указанным при создании именем соответствующей модели содержания (см. п. **Ошибка! Источник ссылки не найден.**);

2) необходимым образом отредактируйте этот файл с учетом его общей структуры и атрибутов.

### Атрибуты шаблона хуков

Общая структура шаблона хуков представлена в таблице **Таблица 5.**

Таблица 5 – Структура шаблона хуков

Название хука	Описание
<Hooks>	Корневой элемент всех хуков потока
<OnDocumentCreated/>	Событие создания документа
<NamedConstants>	Формирование именованного счетчика
<CommonConditions>	Условия, необходимые для того, чтобы счетчики отработали
<Condition path="" operation="" value="" /> <Constant/>	

Название хука	Описание
<Conditions>	Условия, необходимые для отработки счетчика
<Condition path="" operation="" value="" />	
<Setting/>	
key="IncrementedElementPath"	Путь к инкрементированному полю
key="CounterNameFormat"	Формат имени счетчика
<Aggregation/>	Тип обработки события. Выборка всех связанных документов и набор действий
<Select/>	Выборка документов. В результате получаем набор документов исходного потока
<CurrentDoc />	Текущий документ
<Actions/>	Действия, которые необходимо выполнить
<CopyField/>	Перезаписывание полей после сохранения, если были внесены изменения в документ исходного потока
<SetFieldGuid/>	Заполнение целевого поля значением guid
targetFlow	Целевой поток, в котором нужно заполнить guid
targetField	Целевое поле, которое нужно заполнить значением guid
<OnRegcardEdited/>	Событие изменения регистрационной карточки
<Aggregation/>	Тип обработки события. Выборка всех связанных документов и набор действий
<Select/>	Выборка документов. В результате получаем набор документов исходного потока
<Link flow="" />	Поиск связанных документов целевого потока
<Link flow="" />	Поиск документов исходного потока в найденных документах целевого потока
<Actions/>	Действия, которые необходимо выполнить
<Sum/>	Заполнение полей целевого потока суммой значений полей исходного потока
sourceFlow	Исходный поток, в котором ищет поле
targetFlow	Целевой поток, в котором будет происходить запись значения
sourceField	Из какого поля брать значение в исходном потоке
targetField	Поле целевого потока, в которое записывается значение из исходного потока
<Min/>	Заполнение полей целевого потока минимальным значением полей из исходного потока
sourceFlow	Исходный поток, в котором ищет поле

Название хука	Описание
targetFlow	Целевой поток, в котором будет происходить запись значения
sourceField	Из какого поля брать значение в исходном потоке
targetField	Поле целевого потока, в которое записывается значение из исходного потока
<OnDocumentDeleted/>	Событие удаления документа
<Aggregation/>	Тип обработки события. Выборка всех связанных документов и набор действий
<Select/>	Выборка документов. В результате получаем набор документов исходного потока
<Link flow=""/>	Поиск связанных документов целевого потока
<Link flow=""/>	Поиск документов исходного потока в найденных документах целевого потока
<Actions/>	Действия, которые необходимо выполнить
<Sum/>	Заполнение полей целевого потока суммой значений полей исходного потока
sourceFlow	Исходный поток, в котором следует искать поле
targetFlow	Целевой поток, в котором будет происходить запись значения
sourceField	Из какого поля брать значение в исходном потоке
targetField	Поле целевого потока, в которое записывается значение из исходного потока
<Min/>	Заполнение полей целевого потока минимальным значением полей из исходного потока
sourceFlow	Исходный поток, в котором следует искать поле
targetFlow	Целевой поток, в котором будет происходить запись значения
sourceField	Из какого поля брать значение в исходном потоке
targetField	Поле целевого потока, в которое записывается значение из исходного потока
<OnRouteStarted/>	Событие запуска маршрута
<ConditionalActions/>	Тип обработки события. Выполнение действий по условиям
<ConditionalAction/> <Actions/> <SetField/>	Условия и действия
targetField	Поле целевого потока, в которое записывается значение

Название хука	Описание
value=""	Значение, которое записывается в targetField
<Conditions/> <Condition/>	Условия выполнения маршрута
name="RouteKey"	Маршрут
value=""	Значение id маршрута
<FieldsModify> <CopyField/>	Событие предзаполнения полей
src	Поле документа исходного потока
dst	Поле документа целевого потока. Если @dst не указан, то он совпадает с @src
<PurchasePlanFieldsModify/>	Проверка, что не существует созданных планов с указанными статусами. Если статус не указан, то используются статусы: Отправлено в ЕИС, Опубликован, Размещен, Загружен, Неактуальный.
<ThrowExceptionIfDocsExists>	Проверка наличия документов, удовлетворяющих запросу, и генерация исключения, если хотя бы один документ найден. Перечень атрибутов
query	Запрос (возможно указывать параметризованный запрос)
message	Текст сообщения/ошибки, который будет отображен пользователю в случае наличия в базе документов, удовлетворяющих запросу
param-*	Перечень пользовательских параметров для подстановки в запрос, которые ссылаются на установленные значения текущего документа вышестоящими тегами (SetField). Вместо символа * необходимо указать имя параметра
<LinksModify>	Событие добавления связей на новый документ
<LinkCurrentDoc />	Связь исходного документа с целевым документом
<OnLinkDeleted>	Событие удаления связки
<OnLinkAdded>	Событие добавления связки
<Aggregation/>	Тип обработки события. Выборка всех связанных документов и набор действий
<Select>	Выборка документов. В результате получаем набор документов исходного потока
<CurrentDoc />	Текущий документ

## Работа с транзакциями

Хуки могут выполняться в двух режимах:

1) в отдельных транзакциях с сохранением документа (**transaction="split"**). При этом сначала документ сохраняется в базу, а затем выполняются хуки. В случае, если хуки не выполняются (произошла ошибка или будет выполнено целенаправленное действие на отмену; некоторые хуки это позволяют), отменится только транзакция, в которой выполнялись хуки, а документ останется сохраненным. Это может привести к рассогласованию значений в разных документах, которые должны были бы быть синхронными;

2) в одной транзакции с сохранением документа (**transaction="single"**). При этом откат в результате ошибки при обработке хуков или целенаправленный откат приведут к тому, что документ не будет сохранен и останется в состоянии редактирования с выдачей ошибки пользователю (красная рамка с сообщением над документом).

Таким образом, рекомендуется использовать хуки в одной транзакции с сохранением документа, чтобы избежать рассогласования документов и информировать пользователя об ошибках.

По умолчанию хуки запускаются в отдельных транзакциях для совместимости с настройками, которые были созданы на момент добавления возможности работать в единой транзакции. Для явного оказания режима необходимо добавить атрибут **transaction** к элементу, описывающему событие.

Пример настройки:

```
<Hooks>
  <OnRegcardEdited>
    <Aggregation>
      ...
    </Aggregation>
  </OnRegcardEdited>
  <OnRegcardEdited transaction="single">
    <Aggregation>
      ...
    </Aggregation>
  </OnRegcardEdited>
```

Если **transaction="single"**, хуки выполняются в одной транзакции с сохранением документа. Если **transaction="split"** или атрибут не указан, хуки выполняются в отдельных транзакциях с сохранением документа. Таким образом, возможно использование двух секций настроек на конкретное действие.

При этом:

- для действий, в которых сохранение документа не происходит (**OnRouteStarted**), атрибут **transaction** значения не имеет;
- для **transaction="split"** событие **OnDocumentDeleted** не запускается (т. к. после закрытия первой транзакции текущий документ уже удален из базы).

При выполнении хуков в отдельных транзакциях, текущий документ уже сохранен в базе данных, поэтому для доступа к его полям может быть использован следующий способ:

```
<Select>
  <Select searchString="document[doc_Id = '{current_Id}']" tag="current" />
  <Select searchString="document[doc_RegCard/rc_FlowName = 'Заявки' and
doc_RegCard/rc_Index/text_ИД_в_ЕИС={current}/doc_RegCard/rc_Index/text_ИД_в_ЕИС']"
tag="Заявки" />
</Select>
```

В этом случае тег **current** может использоваться как для доступа к документу в действиях, так и для использования в качестве подзапроса в качестве подзапроса в других селектах (как в примере).

При выполнении хуков в одной транзакции с сохранением документа, в базе данных на этот момент либо еще нет документа (в случае его создания), либо хранится еще не измененная версия (и этот момент можно использовать для того, чтобы в хуках проанализировать произошедшие изменения). Поэтому доступ к текущему документу с актуальными изменениями осуществляется иначе. Причем этими способами можно пользоваться и в случае отдельных транзакций, и зачастую это будет работать быстрее.

## Доступ к текущему документу в селектах

Если нужно использовать значения полей документа или список связей в поиске, можно использовать следующие постановки:

- **{current\_Id}** – идентификатор текущего документа;
- **{currentDocField=|Document|Имя\_поля}** – значение поля из текущего документа;
- **{currentDocLinkList}** – список идентификаторов связанных документов из текущего документа (если их нет, то значение будет «-1»)

Пример использования постановок:

```
<Select>
  <Select searchString="document[doc_RegCard/rc_FlowName = 'Лоты' and
doc_RegCard/rc_Index/int_Позиция={current_Id}]" tag="Лоты" />
  <Select searchString="document[doc_RegCard/rc_FlowName = 'Лоты' and
doc_RegCard/rc_Index/text_ИД_в_ЕИС='{currentDocField=|Document|ИД_в_ЕИС}']"
tag="Лоты" />
</Select>
```

```

<Select searchString="document[xn-in(doc_Id, {currentDocLinkList})]"
tag="СвязанныеДокументы" />
<Select searchString="document[doc_RegCard/rc_FlowName = 'Позиции' and xn-
in(doc_Links/doc_Link/link_Document, {currentDocLinkList})]"
tag="СвязанныеСоСвязаннымиДокументамиПозиции" />
</Select>

```

Может возникнуть необходимость в одном запросе выбрать несколько документов, включая текущий. Обычным поисковым запросом это сделать не получится, т. к. текущего документа в базе либо нет, либо там пока еще старая неизменная версия. Поэтому нужно выбрать все необходимые документы, кроме текущего, а затем объединить их с текущим с помощью следующей конструкции:

```

<Select>
  <Select searchString="document[doc_RegCard/rc_FlowName = 'Лоты' and doc_Id !=
  {current_Id}]" tag="Лоты" />
  <Select tag="ВсеЛоты" joinTags="Лоты,{current}" />
</Select>

```

## Доступ к текущему документу в действиях

Текущий документ неявно добавляется к списку документов, получаемых с помощью селектов, с тегом "{current}". Таким образом его можно использовать в действиях, например:

```

<Actions>
  <Sum sourceFlow="Позиции" targetFlow="{current}"
  sourceField="|Document|Сумма_выплат_всего"
  targetField="|Document|Сумма_выплат_всего" />
  <CopyField sourceFlow="{current}" targetFlow="Лоты" sourceField="|Document|ИНН"
  targetField="|Document|ИНН" />
</Actions>

```

Также неявно существует тег "{oldCurrent}", который позволяет получать доступ к данным текущего документа, которые были до его изменения. Естественно, «старых» данных нет, когда создается новый документ (**OnDocumentCreated**). Также нельзя применять к нему хуки, которые его изменяют (возникнет ошибка **NotImplementedException**).

## Специальные функции в запросах при обращении к полям текущего документа

В некоторых случаях возникает необходимость заменить значения текущего документа при подстановке его в запрос.

Например, запрос, содержащий выражение

```
doc_RegCard/rc_Index/boolean_Иновационная_продукция =
{currentDocField=|Document|Иновационная_продукция}'
```

будет выдавать ошибку в случае незаполненного поля документа «Иновационная\_продукция»: в запрос будет подставлена пустая строка, и левая часть вышеуказанного выражения вернет ошибку, т. к. логический тип сравнивается со строковым.

Для этого введена специальная функция **NullToZero** для строк запроса, которая преобразует пустое значение в «0». Таким образом, после подстановки получится запрос **doc\_RegCard/rc\_Index/boolean\_Иновационная\_продукция = '0'**, который успешно выполнится. Пример использования:

```
doc_RegCard/rc_Index/boolean_Иновационная_продукция =
' {NullToZero ({currentDocField=|Document|Иновационная_продукция}) } '
```

## Агрегирование значений

Агрегатные функции предназначены для вычисления некоторых показателей для заданного множества строк (сумма, минимальное и максимальное значения и т. д.). Таким множеством строк может быть группа строк (если агрегатная функция применяется к сгруппированной таблице) или вся таблица.

Доступны следующие агрегатными функции:

- Max – вычисление максимума из заданных значений;
- Min – вычисление минимума из заданных значений;
- Sum – вычисление суммы для заданных значений;
- Average – вычисление среднего арифметического для заданных значений;
- Count – вычисление общего количества заданных значений.

В случае, если среди числовых данных попадаются пустые (незаполненные) значения, они игнорируются, т. е. не участвуют в общем подсчете (за исключением функции Count).

Пример:

```
<Columns>
  <Column name="Дата" />
  <Column name="Сумма_платежа" aggregate="sum" />
  <Column name="Сумма_платежа" aggregate="count" />
</Columns>
```

При такой настройке, табличные данные будут «сжаты» по дате (т. к. для столбца «Дата» не указана операция агрегирования). Для каждой даты будет подсчитана сумма платежей и их количество.

Для более быстрой выборки рекомендуется использовать атрибут `useSqlAggregate="yes"`, в таком случае агрегатная функция будет вычисляться средствами СУБД (через хранилище). Если этот атрибут не указан, то вычисление агрегатной функции происходит традиционным способом – путем загрузки всех документов кодом и «ручного» извлечения значений.

## Вычисление значения выражений

Данное действие выполняет вычисление значения выражения и его запись в реквизит.

Пример:

```
<CalculateExpression targetFlow="Позиции плана-графика1" targetField="|Document|Остаток_в_ППЗ">
  <Variables>
    <Variable name="СуммаППЗ" source="Позиции плана закупок2"
      path="|Document|Сумма_всего" />
    <Variable name="Сумма_всего" source="Позиции плана-графика1"
      path="|Document|Сумма_всего" aggregation="sum" />
  </Variables>
  <Expression value="$NullToZero({СуммаППЗ}) - $NullToZero({Сумма_всего})"/>
</CalculateExpression>

<CalculateExpression targetFlow="Все позиции" targetField="|Document|Остаток_в_ППГ">
  <Variables>
    <Variable name="СуммаППГ" source="Позиции плана-графика"
      path="|Document|Сумма_всего" />
    <Variable name="СуммаПозиции" source="Все позиции"
      path="|Document|Сумма_выплат_всего" aggregation="sum" />
  </Variables>
  <Expression value="$NullToZero({СуммаППГ}) - $NullToZero({СуммаПозиции})"/>
</CalculateExpression>
```

Атрибуты **targetFlow** и **targetField** определяют, в какие документы и по какому пути записывать вычисленное выражение:

1) **Variables** – список переменных, которые участвуют в вычислении выражения. Каждая переменная описывается следующим образом:

- **name** – название переменной, которая используется в выражении;
- **source** – отбор документов, которые участвуют в вычислении значения переменной. Возможны значения, которые используются в селектах (атрибут tag);
- **path** – путь до реквизита, на основе которого вычисляется значение переменной;
- **aggregation** – агрегатная функция, которую следует применить к значениям, полученным из реквизита, указанного в path. Допустимые значения:
  - none – первое непустое значение. Используется по умолчанию, если атрибут **aggregation** не указан;
  - min – минимальное значение;
  - max – максимальное значение;
  - Sum – сумма;
  - Average – среднее арифметическое;
  - Count – количество.

2) **Expression** – выражение, значение которого необходимо вычислить.

## Изменение и проверка полей таблиц (**TableFieldOperations**)

Данное действие позволяет вносить изменения в поля таблиц во все или в некоторые отобранные строки, а также выполнять проверки полей по условиям и отменять сохранение документа с выводом сообщения.

Таблица и строки, которые будут отредактированы с помощью **TableFieldOperations** (целевая таблица, целевые строки), описываются атрибутами **targetTag**, **targetTable** и **targetRowFilter**. Для каждой из всех или отобранных целевых строк будет выполнено действие, которое описано в элементах **Action**.

Источник данных описывается атрибутом **sourceTag**. Если в теге будет отобрано несколько документов-источников, то все действия (**Action**) будут выполняться соответствующее количество раз (таким образом, например, можно подсчитать сумму). При необходимости в документах-источниках можно указать таблицу-источник (с помощью необязательного элемента **SourceTable**). В этом случае все действия (**Action**) будут выполняться для каждой строки таблицы в каждом документе.

Настройки атрибутов **TableFieldOperations** представлены в таблице **Таблица 6.**

Таблица 6 – Настройка атрибутов TableFieldOperations

Наименование атрибута	Значение атрибута	Пример
<b>sourceTag</b>	Документ(ы)-источник(-и) данных (атрибут <b>tag</b> из раздела <b>Select</b> )	
<b>targetTag</b>	Целевой(-ые) документ(ы) (атрибут <b>tag</b> из раздела <b>Select</b> )	
<b>targetTable</b>	Имя целевой таблицы	
<b>targetRowFilter</b>	Формула фильтрации строк целевой таблицы. Если не указана – используются все строки, если указана – только строки, подходящие под условия. В формуле можно использовать подстановки	
{ИмяСтолбцаТаблицы}	Значение берется из указанного поля текущей строки	<b>targetRowFilter="{Этап_закрыт}=1 And {Количество}=0"</b>
{field=ИмяПоля}	Значение поля текущего документа-источника	<b>targetRowFilter="{Этап_закрыт}=1 And {field= Document Количество}=0"</b>
{fieldTarget=ИмяПоля}	Значение поля целевого документа	<b>formula="{Цена}*({fieldTarget= Document Количество}+100)"</b>
{field=ЗначениеТега:ИмяПоля}	Аналогично предыдущему, но значение поля берется из первого документа найденного набора документов (атрибут <b>tag</b> из раздела <b>Select</b> )	<b>targetRowFilter="{Этап_закрыт}=1 And {field= Лоты: Document Количество}=0"</b>
{sourceTableColumn=ИмяСтолбца}	Работает только при наличии элемента <b>SourceTable</b> , значение поля берется из указанного столбца текущей строки таблицы-источника ( <b>ИмяСтолбца</b> должно соответствовать атрибуту <b>name</b> в элементе <b>Column</b> )	
<b>checkFormula</b>	Формула проверки, которая будет применена к отображенным строкам целевой таблицы. Если любая строка	

Наименование атрибута	Значение атрибута	Пример
	удовлетворяет формуле, пользователю будет выдано заданное в <b>checkMessage</b> сообщение и сохранение документа будет отменено. В формуле можно использовать подстановки аналогичные <b>targetRowFilter</b>	
<b>checkMessage</b>	Сообщение, которое будет выдано пользователю при непрохождении проверки	
<b>Атрибуты таблицы-источника (SourceTable)</b>		
<b>tableName</b>	Имя таблицы	Возможно указать подчиненную таблицу в следующем виде: " <b> Document Таблица * Подтаблица </b> " – в этом случае будут перебираться все строки всех подтаблиц во всех строках основной таблицы
<b>Атрибуты столбца (Column)</b>		
<b>name</b>	Имя столбца таблицы	
<b>aggregate</b>	Операция агрегации, которая будет производиться над значением столбца	
<b>Атрибуты действия (Action)</b>		
<b>column</b>	Имя столбца целевой таблицы, в который будет вноситься значение	
<b>formula</b>	Формула, результат вычисления которой будет записан в ячейку столбца. В формуле можно использовать подстановки	
{ИмяСтолбцаТаблицы}	Значение берется из указанного поля текущей строки целевой таблицы	<b>formula="{Цена}*{Количество}"</b>
{field=ИмяПоля}	Значение поля документа-источника	<b>formula="{Цена}*({field= Document Количество}+100)"</b>
{fieldTarget=ИмяПоля}	Значение поля целевого документа	<b>formula="{Цена}*({fieldTarget= Document Количество}+100)"</b>
{field=ЗначениеТега:Имя}	Аналогично предыдущему, но	<b>formula="{Цена}*({field=</b>

Наименование атрибута	Значение атрибута	Пример
Поля}	значение поля берется из первого документа найденного набора документов (атрибут <b>tag</b> из раздела <b>Select</b> )	Лоты: Document Количество}+100)"
{sourceTableColumn=Имя Столбца}	Работает только при наличии элемента <b>SourceTable</b> , значение поля берется из указанного столбца текущей строки таблицы-источника ( <b>ИмяСтолбца</b> должно соответствовать атрибуту <b>name</b> в элементе <b>Column</b> )	

Элементами **TableFieldOperations** являются:

- **SourceTable** – таблица-источник. Если этого элемента нет, то список действий будет выполняться один раз для каждого документа-источника, если есть – то для каждой строки указанной таблицы каждого документа;

- **Action** (множественный) – перечень действий, которые будут произведены над полями отобранных строк.

Пример элементов **TableFieldOperations**:

```
<TableFieldOperations sourceTag="{current}" targetTag="контракты"
targetTable="|Document|Номенклатура" targetRowFilter="{Этап_закрыт}=1 And
{field=|Document|Количество}=0">
  <Action column="Стоимость" formula="{Цена} * {Количество}" />
  <Action column="Обновлено" formula="1" />
</TableFieldOperations>
```

Пример атрибутов действия (Action)

```
<TableFieldOperations sourceTag="{current}" targetTag="контракты"
targetTable="|Document|Этапы_исполнения_контракта"
targetRowFilter="{Дата_окончания_этапа_контракта}={field=|Document|Этап_контракта}">
  <SourceTable tableName="|Document|Исполнение_контракта">
    <Column name="фактически_оплачено" aggregate="Sum" />
  </SourceTable>
  <Action column="фактическая_оплата_по_этапу"
formula="{фактическая_оплата_по_этапу}+{sourceTableColumn=фактически_оплачено}" >
</TableFieldOperations>
```

*Примечание.* Чтобы обновлять разные строки по-разному, добавляйте несколько блоков **TableFieldOperations** с разными фильтрами (**targetRowFilter**).

## Консолидация табличных данных (TableConsolidation)

Данное действие позволяет объединять строки из разных таблиц и (или) полей документов (с возможным применением функций агрегации, например, суммирование) и записывать результат в другие таблицы.

В качестве источника данных могут служить как таблицы в документах, так и наборы документов (во втором случае поля документов становятся полями таблиц).

В качестве источника можно описать несколько различных таблиц в одних и тех же или разных документах (**SourceTable**). В качестве приемника указывается одна таблица в целевом наборе документов (**TargetTable**).

Столбцы таблицы (**Column**) консолидируются в порядке их следования в описаниях таблиц-источников (**SourceTable**) и таблицы приемника (**TargetTable**). Количество столбцов может не соответствовать друг другу, значения в недостающих столбцах будут считаться пустыми. Описывать в источниках (**SourceTable**) количество столбцов больше, чем в приемнике (**TargetTable**), не имеет смысла – лишние столбцы будут проигнорированы.

Консолидация табличных данных осуществляется следующим образом:

1) перебираются все документы, соответствующие тегам, указанным в атрибуте **flow** для каждого элемента **SourceTable**;

2) значения всех столбцов таблиц (**tableName**) или полей документов (**useDocsAsTable**) из всех **SourceTable** собираются в единый список;

3) если в каких-то столбцах таблицы-приемника указана агрегатная операция, то строки таблицы «сжимаются» по всем остальным полям;

4) результирующие строки записывается в таблицу-приемник. При этом, если в одном или нескольких столбцах указано **findExisting="true"**, то выполняется попытка найти строку с совпадающими в этих столбцах данными (совпадать должны все столбцы с **findExisting="true"**). Если строка не найдена, то добавляется новая.

Настройка элементов и атрибутов описана в таблице Таблица 7.

Таблица 7 – Настройка элементов и атрибутов

Наименование	Значение
<b>Элементы TableConsolidation</b>	
SourceTable (множественный)	Описывает таблицы, из которых будут консолидироваться данные
TargetTable	Описывает таблицу, в которую будет записан результат
<b>Элементы таблицы-источника (SourceTable)</b>	
Column (множественный)	Описание столбца таблицы
<b>Атрибуты таблицы-источника (SourceTable)</b>	
flow	Имя тега выборки документов (из <b>Select</b> или

Наименование	Значение
	текущий документ)
tableName	Имя таблицы
useDocsAsTable	<b>true</b> – признак того, что в качестве столбцов будет использоваться не таблица, а поля самого документа (используется вместо <b>tableName</b> )
<b>Атрибуты столбца таблицы-источника (Column)</b>	
name	Имя столбца таблицы или поля документа
<b>Элементы таблицы-приемника (TargetTable)</b>	
Column (множественный)	Описание столбца таблицы
<b>Атрибуты таблицы-приемника (TargetTable)</b>	
flow	Имя тега выборки документов (из <b>Select</b> или текущий документ)
tableName	Имя таблицы
<b>Атрибуты столбца таблицы-приемника (Column)</b>	
name	имя столбца таблицы
findExisting	<b>true</b> – в данном столбце будет производиться поиск уже существующего значения
aggregate	Операция агрегации, которая будет производиться над значением столбца

## Пример:

```

<Hooks>
  <OnRegcardEdited transaction="single">
    <Aggregation>
      <Select />
      <Actions>
        <TableConsolidation>
          <SourceTable flow="{current}"
            tableName="|Document|Этапы_исполнения_бюджетные_средства">
            <Column name="Дата_окончания_этапа_бюджетные_средства" />
            <Column name="Сумма_платежей_этапа_БС" />
          </SourceTable>
          <SourceTable flow="{current}"
            tableName="|Document|Этапы_исполнения_внебюджетные_средства">
            <Column name="Дата_окончания_этапа_внебюджетные_средства" />
            <Column name="Сумма_платежей_этапа_БС" />
          </SourceTable>
          <TargetTable flow="{current}"
            tableName="|Document|Этапы_исполнения_контракта">
            <Column name="Дата_окончания_этапа_контракта" findExisting="true" />
            <Column name="Сумма_платежей_этапа_контракта" aggregate="Sum" />
          </TargetTable>
        </TableConsolidation>
      </Actions>
    </Aggregation>
  </OnRegcardEdited>
</Hooks>

```

## 9. ПРИКРЕПЛЕНИЕ ФОРМЫ ДОКУМЕНТА К ПОТОКУ

Необходимость прикрепить форму документа к интересующему потоку документов может возникнуть в следующих случаях:

- создается новая форма;
- редактируется существующая форма, которую следует прикрепить к другому потоку.

Для того чтобы прикрепить форму к потоку:

- 1) запустите модуль «**Ошибка! Неизвестное имя свойства документа.**»;
- 2) в режиме работы **Потоки и маршруты** на вкладке **Потоки** выберите поток, к которому следует прикрепить форму (рис. Рисунок 6);

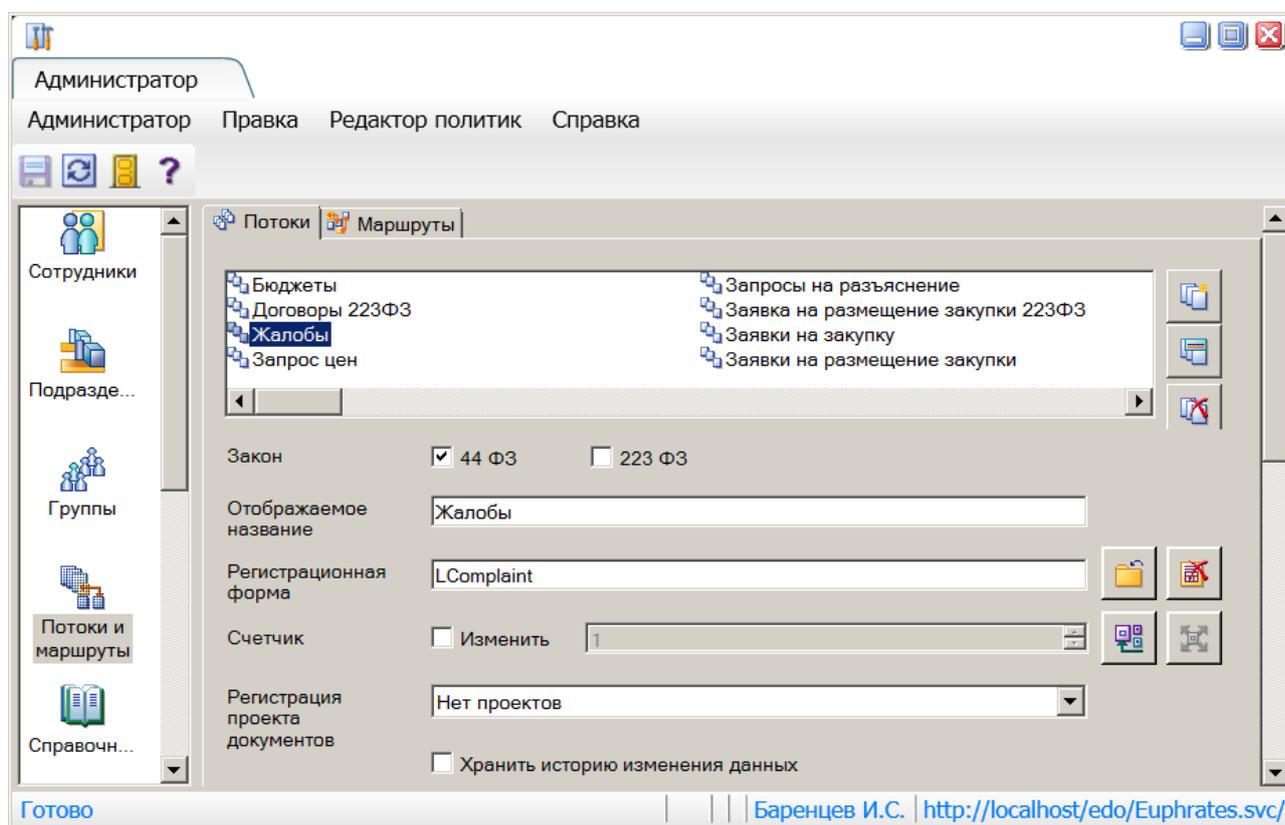


Рисунок 6

3) справа от поля **Регистрационная форма** нажмите на кнопку . Откроется окно **Список регистрационных форм** (рис. Рисунок 7);

4) выберите форму, которую следует прикрепить к потоку, и нажмите на кнопку **ОК**. Если интересующая форма не отображается в списке, нажмите на кнопку ;

5) на панели инструментов нажмите на кнопку .

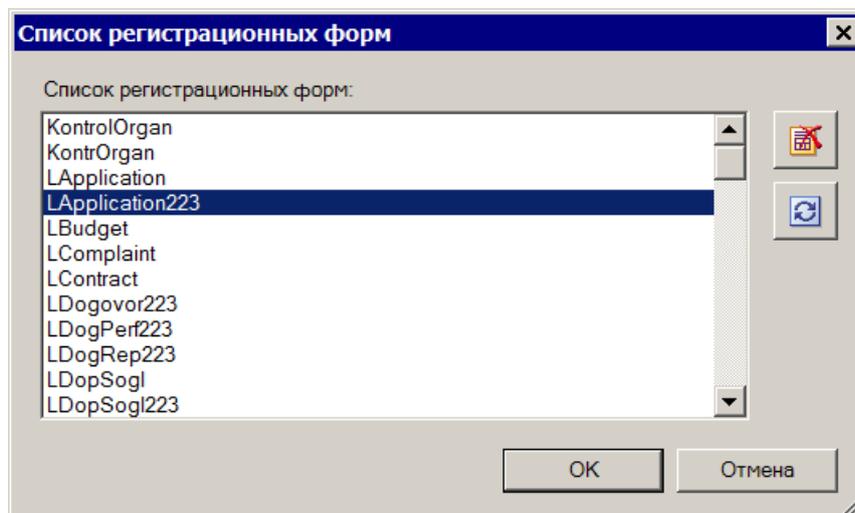


Рисунок 7

## 10. ПРОВЕРКА РАБОТОСПОСОБНОСТИ ФОРМЫ ДОКУМЕНТА

Для того чтобы проверить работоспособность формы документа:

- 1) войдите в Систему под учетной записью пользователя, обладающего правом создания документа потока, соответствующего интересующей форме;
- 2) для создания документа перейдите на соответствующую вкладку (см. документ «Руководство пользователя»);
- 3) нажмите на кнопку **Сформировать** **<Наименование потока документа>** (рис. Рисунок 8);

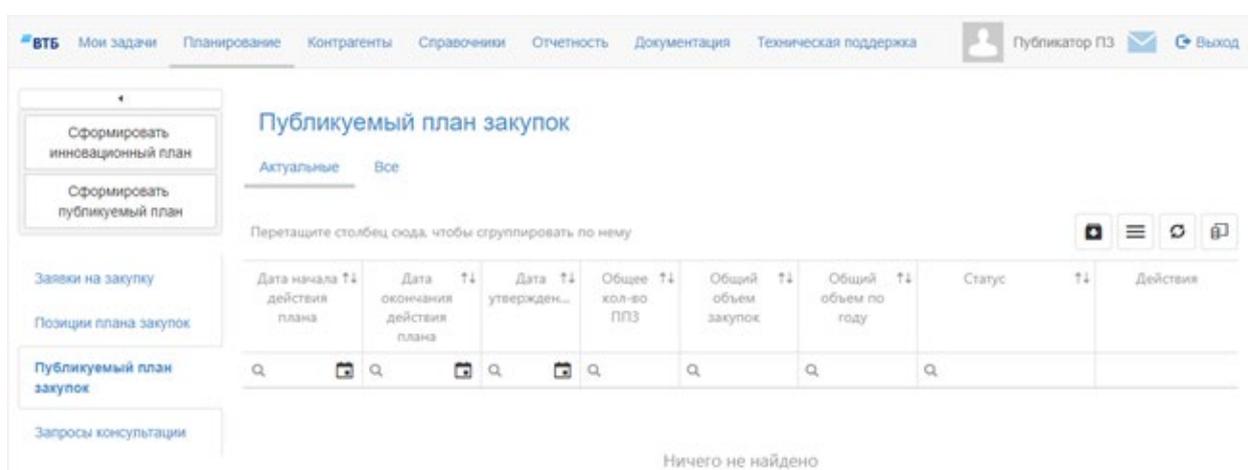


Рисунок 8

- 4) на открывшейся странице выбора потока щелкните левой кнопкой мыши по интересующему потоку. Откроется страница регистрации документа (рис. Рисунок 9)

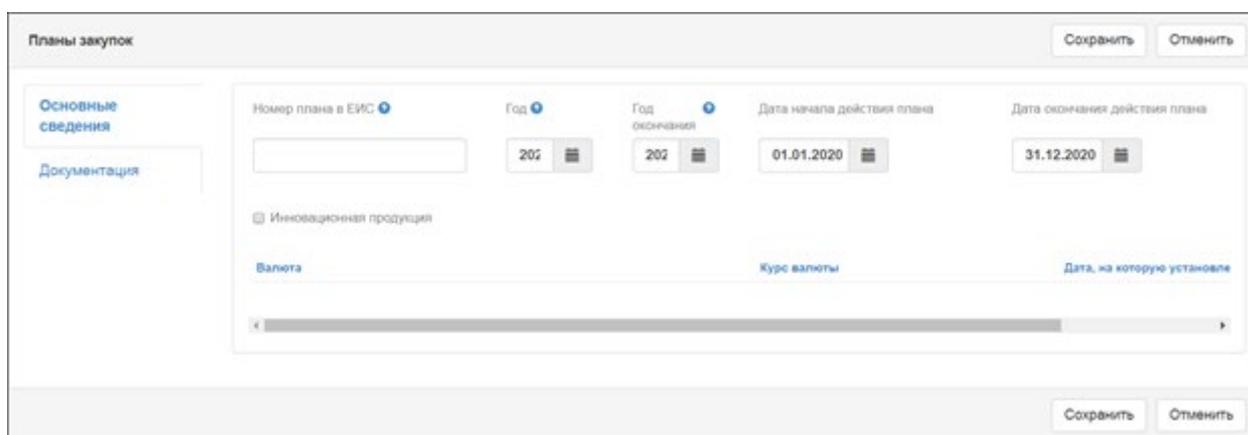


Рисунок 9

- 5) заполните все поля на всех вкладках регистрационной карточки
- 6) присоедините необходимые файлы;
- 7) на панели инструментов нажмите на кнопку **Сохранить**;

- 8) убедитесь, что на экране:
- отображается доступная только для просмотра регистрационная карточка зарегистрированного документа;
  - не отображаются никакие сообщения об ошибках;
- 9) убедитесь, что зарегистрированный документ появляется в результатах поиска документов по нескольким поисковым параметрам.