



Руководство по настройке экранных форм

АННОТАЦИЯ

Настоящий документ предназначен для пользователей, наделенных правами администрирования и осуществляющих обслуживание и поддержку работоспособности информационной системы управления закупками, интегрированной с единой информационной системой в сфере закупок.

В настоящем документе приведено описание операций, выполняемых администратором для настройки и поддержания работоспособности системы, а также инструкции для настройки экранных форм.

Содержание

1. Введение.....	6
1.1. Область применения.....	6
1.2. Краткое описание возможностей.....	6
1.3. Условные обозначения.....	7
1.3.1. Оформление специального текста.....	7
2. Назначение и условные применения.....	9
2.1. Автоматизируемые функции.....	9
2.2. Задачи администрирования легкого клиента.....	10
2.2.1. Требования к программному обеспечению для настройки легкого клиента.....	10
2.2.2. Уровень подготовки пользователей.....	11
3. Основные понятия модуля «Дизайнер форм».....	12
3.1. Главное меню и панель инструментов.....	12
3.2. Модель содержания.....	13
3.2.1. Модель данных.....	16
3.2.2. Элемент данных.....	16
3.2.3. Структура.....	17
3.2.4. Массив.....	17
3.2.5. Простое данное.....	17
3.3. Форма.....	18
3.3.1. Экранная форма.....	18
4. Настройка шаблонов легкого клиента.....	19
4.1. Расположение файлов настройки.....	19
4.2. Основные понятия.....	20
4.3. Отображение регистрационной и контрольной карточек папки «Templates». Краткое описание папки «Administration».....	21
4.4. Краткое описание папки «Dictionary».....	22
4.5. Краткое описание папки «Document».....	22
4.6. Краткое описание папки «Plugins».....	25
4.6.1. Описание шаблона Назначение ответственного руководителя, исполнителя.....	25
4.6.2. Описание шаблона Регистрация заявки.....	27
4.6.3. Описание шаблона Назначение исполнителя правового отдела.....	28
4.6.4. Структура файла настройки «Flows.xml».....	30
4.7. Краткое описание папки «Reports».....	31
4.8. Краткое описание папки «SearchForm».....	31
4.9. Краткое описание папки «SearchResult».....	32
4.10. Краткое описание папки «Upload».....	33
4.11. Краткое описание папки «Scripts».....	33
4.11.1. Настройка использования скриптов.....	33
4.12. Выбор шаблона для отображения пользовательской папки.....	34
4.13. Выбор шаблона для отображения связанных документов.....	37

4.14. Настройка шаблонов отображения регистрационной карточки	41
4.14.1. Общие сведения	41
4.14.2. Отображение списка полей карточки документа	42
4.14.3. Таблицы связанных документов	42
4.14.4. Набор колонок для отображения в таблице связанных документов	45
4.14.5. Сортировка в таблице связанных документов	46
4.14.1. Настройки таблицы связанных документов	46
4.14.2. Настройка кнопки удаления связанных документов	47
4.14.3. Настройка блока просмотра	48
4.14.4. Настройка отображения таблиц	50
4.14.5. Выбор значений из полей /таблиц и других форм (FormdataDict)	53
4.14.6. Группировка дат для контроля хронологии	55
4.14.7. Настройка видимости кнопок верхней панели	56
4.14.8. Настройка блока редактирования	57
4.14.9. Переменные в блоке редактирования	62
4.15. Порядок создания шаблона отображения новой модели содержания или потока документов	62
4.16. Настройка соответствия шаблонов отображения регистрационных карточек документов конкретным потокам	63
4.16.1. Механизм определения шаблона отображения регистрационной карточки	63
4.17. Настройка шаблонов форм поиска	63
4.18. Формат элемента иерархического словаря	68
4.19. Формат шаблона присоединенных файлов	73
5. Типы полей	75
5.1.1. Адресная книга	75
5.1.2. Текстовое поле	80
5.1.3. Числовое поле инфы	80
6. Хуки	82
6.1.1. Работа с транзакциями	82
6.2. Конфигурирование файлов	84
6.2.1. Агрегатные функции	87
6.2.2. Вычисление значения выражений	88
6.2.3. Изменение (проверка) полей таблиц (TableFieldOperations)	89
6.2.4. Консолидация табличных данных (TableConsolidation)	92
6.3. Сервисные плагины и утилиты	103
6.4. Настройка ЭП	104
6.4.1. Общие сведения	104
6.4.2. Общий порядок настройки ЭП	106
6.4.3. Формирование новых сертификатов ключей подписи	112

6.4.4. Особенности, связанные со сменой криптопровайдера или изменением параметров криптопровайдера	113
---	-----

1. ВВЕДЕНИЕ

1.1. Область применения

Информационная система управления закупками, интегрированная с единой информационной системой в сфере закупок (далее — ИС «Управление закупками» или Система) предназначена для автоматизации процессов планирования закупок товаров, работ, услуг для обеспечения государственных и муниципальных нужд в соответствии с Федеральным законом от 05 апреля 2013 года «О контрактной системе в сфере закупок товаров, работ, услуг для обеспечения государственных и муниципальных нужд» (далее – 44-ФЗ), для обеспечения удовлетворения потребностей в товарах, работах, услугах отдельных видов юридических лиц в соответствии с Федеральным законом от 18 июля 2011 года № 223-ФЗ «О закупках товаров, работ, услуг отдельными видами юридических лиц» (далее – 223-ФЗ), осуществления закупок по 44-ФЗ (далее – закупка по 44-ФЗ) и № 223-ФЗ (закупка по 223-ФЗ), исполнения контрактов (договоров), мониторинга и аудита в сфере закупок, контроля за соблюдением законодательства Российской Федерации и иных нормативных правовых актов о контрактной системе в сфере закупок и Федерального закона от 18 июля 2011 года № 223-ФЗ «О закупках товаров, работ, услуг отдельными видами юридических лиц» (далее – 223-ФЗ). Система должна формировать единый достоверный и актуальный реестр закупок товаров, работ, услуг.

Объектами автоматизации является функциональная деятельность заказчиков (получатели бюджетных средств, главные распорядители бюджетных средств), осуществляющих закупки товаров, работ, услуг в соответствии с 44-ФЗ (далее – заказчик по 44-ФЗ), заказчиков, осуществляющих закупки товаров, работ, услуг в соответствии с 223-ФЗ (далее – заказчик по 223-ФЗ).

1.2. Краткое описание возможностей

Легкий клиент обеспечивает выполнение большинства прикладных функций комплекса. Подробное описание возможностей приводиться в п. 2.1.

1.3. Условные обозначения

1.3.1. Оформление специального текста

В настоящем документе специальный текст оформляется в соответствии с таблицей 1.

Таблица 1— Оформление специального текста

Тип специального текста	Пример использования в тексте
Текст в полужирном начертании	
Названия окон	В окне Создать задачу настройте параметры для создания новой задачи
Названия кнопок	Нажмите на кнопку Сохранить
Названия полей ввода и областей ввода	В поле Логин укажите регистрационное имя
Названия списков (раскрывающихся списков, иерархических списков), названия элементов в списках (кроме иерархических списков)	В списке Исполнитель выберите исполнителя задачи
Названия переключателей и их положений	По умолчанию переключатель Неявные права на документ после завершения установлен в положение Отменить
Названия флажков	Установите флажок Выбрать все
Названия секций в окнах	В секции Регистрационная карточка указываются регистрационные данные документа
Названия столбцов в таблицах	В столбце Название отображается название документа
Названия вкладок	Выберите вкладку На контроле
Слово «Внимание!» (этим словом выделяется текст с информацией о важных последствиях какой-либо операции)	Внимание! Удаление маршрута со статусом «активный» делает этот маршрут недоступным для использования в модуле «Е1»
Текст в полужирном начертании шрифта «Courier New»	
Регистрационные имена и пароли	При запуске сеанса работы с модулем укажите регистрационное имя пользователя sysadmin
Текст, вводимый с клавиатуры, и значения параметров	По умолчанию значение параметра равно 1

Текст в угловых скобках	
Названия клавиш на клавиатуре	Нажмите комбинацию клавиш <Ctrl+Shift>
Текст в курсивном начертании	
Термины при первом употреблении	<i>Маршрутом</i> называется последовательность этапов обработки документа
Слово «Примечание» (этим словом выделяется текст с дополнительной информацией)	<i>Примечание 1.</i> При публикации маршрута автоматически выполнится его сохранение

Тип специального текста	Пример использования в тексте
Текст в угловых кавычках	
Названия модулей	Модуль «Дизайнер маршрутов» применяется на рабочем месте администратора
Названия элементов в иерархических списках	Посмотреть список документов в папке «Результаты последнего поиска»
Любые названия, не предусмотренные настоящим соглашением и требующие специального выделения	

2. НАЗНАЧЕНИЕ И УСЛОВНЫЕ ПРИМЕНЕНИЯ

2.1. Автоматизируемые функции

Средствами легкого клиента обеспечивается выполнение следующих функций:

- работа с папками «Мои документы» и «Общие документы»;
- регистрация документов и создание проектов документов;
- использование многозначных реквизитов (например, каких-либо таблиц);
- присоединение файлов к документам;
- создание версий и подверсий присоединенных файлов;
- создание ссылок на связанные документы;
- перемещение электронных документов между сотрудниками;
- создание и исполнение поручений по документам;
- согласование документов;
- ознакомление с документами;
- создание и использование маршрутов и подмаршрутов обработки документов;
- контроль исполнения документов, а также маршрутов и поручений, согласований и ознакомлений по ним;
- создание отчетов различных видов по документообороту;
- поиск документов, поручений, согласований и ознакомлений в базе данных (БД) комплекса.

Далее по тексту документа, в качестве термина, объединяющего термины «поручение», «согласование» и «ознакомление», используется термин «задача».

Легкий клиент не предназначен для выполнения следующих операций:

- работа с системными папками комплекса (кроме папки «Общие документы»);
- регистрация на основе текущего документа;
- регистрация документа на основе присоединяемого файла;
- регистрация документа со сканированием;
- регистрация документа с распознаванием;
- регистрация документа из внешних приложений;
- присоединение документов к электронным сообщениям;
- выгрузка документов в формате xml, соответствующем ГОСТ Р 53898-2010 «Системы электронного документооборота. Взаимодействие систем управления документами. Требования к электронному сообщению»;
- использование электронной подписи в документообороте;

- назначение явных прав доступа к документу;
- удаление документов;
- прямое редактирование присоединенных файлов с сохранением

результата на сервере комплекса;

- редактирование маршрутов;
- обмен электронными сообщениями между сотрудниками.

Для выполнения этих операций следует использовать модуль «Е1 закупки».

2.2. Задачи администрирования легкого клиента

Отображение данных на страницах легкого клиента настраивается в специальных *шаблонах*. К отображению данных относятся, например, наборы и расположение полей регистрационной карточки на страницах документов, просматриваемых с помощью легкого клиента, наборы столбцов в списках документов и задач. Шаблоны отображения данных представляют собой файлы формата xml (далее также — файлы настройки). Легкий клиент поставляется с набором шаблонов отображения, настроенным по умолчанию. В частности, в него входят шаблоны отображения для регистрационной карточки базовой конфигурации комплекса. Администратор комплекса имеет возможность изменить настройку шаблонов отображения, выполненную по умолчанию, а также создать шаблоны отображения для регистрационной карточки, не входящих в базовую конфигурацию.

Некоторые параметры легкого клиента, например, автоматическая доменная авторизация, продолжительность тайм-аута сеанса работы с комплексом через легкий клиент, настраиваются в специальном файле — «Web.config». Администратор комплекса имеет возможность изменить настройку этого файла, выполненную по умолчанию.

2.2.1. Требования к программному обеспечению для настройки легкого клиента

Настройка шаблонов отображения данных и файла «Web.config» выполняется на компьютере, на котором установлен сервер приложений комплекса «Е1». Для работы с файлами настройки и файлом «Web.config» необходимо, чтобы на компьютере было установлено программное обеспечение для редактирования файлов формата xml (например, MS Notepad, Блокнот и т.д.).

2.2.2. Уровень подготовки пользователей

Администратору комплекса, выполняющему настройку шаблонов отображения данных, следует обладать следующими знаниями и навыками:

- знание основных понятий электронного документооборота (см. документ «Руководство пользователя»);
- понимание принципа работы приложений на основе трехзвенной архитектуры клиент-сервер;
- представление о внешнем виде страниц легкого клиента (см. документ «Легкий клиент. Руководство пользователя»);
- знание языка разметки XML;
- знание принципов построения экранных форм, используемых в комплексе «Е1»;
- знание принципов построения отчетных форм, используемых в комплексе «Е1».

3. ОСНОВНЫЕ ПОНЯТИЯ МОДУЛЯ «ДИЗАЙНЕР ФОРМ»

3.1. Главное меню и панель инструментов

Главное окно модуля «Дизайнер форм» состоит из следующих рабочих областей (см. рис1. и рис. 2):

- **Главное меню модуля.** Содержит набор основных функций модуля;
- **Главная панель инструментов.** Содержит набор общих команд модуля;
- **Панель инструментов.** Модели содержания содержит набор команд, выполняемых над моделью содержания;
- **Панель инструментов статической формы.** Содержит набор команд, выполняемых над статической формой;
- **Панель инструментов выравнивания.** Содержит набор команд для выравнивания элементов формы;
- **Окно отображения модели содержания.** Отображается модель содержания;
- **Окно отображения словарей.** Отображаются словари, подключенные к данной модели содержания;
- **Окно отображения статической формы.** Отображается статическая форма;
- **Панель отображения свойств элементов формы.** Содержится набор свойств выбранного на форме элемента;
- **Панель отображения свойств элементов модели.** Содержится набор свойств выбранного в модели элемента;
- **Строка состояния.** Отображаются подсказки для команд меню.

простые данные. В промежуточных вершинах дерева могут находиться только структуры и массивы.

Структуры объединяют в группу различные данные по какому-либо признаку, назначению данных. Например, структура **ФИО** может состоять из трех простых данных: **Фамилия, Имя** и **Отчество**.

Иерархия в модели содержания необходима для отражения подчинения данных. Например, про человека можно знать: фамилию, имя, отчество (это одна структура), место и дату рождения (это может быть вторая структура), образование (это третья структура).

Причем, данные по образованию могут также состоять из нескольких элементов: основное образование, второе образование, дополнительное образование.

Пример такого дерева представлен на рисунке 3.

Пример иерархической структуры

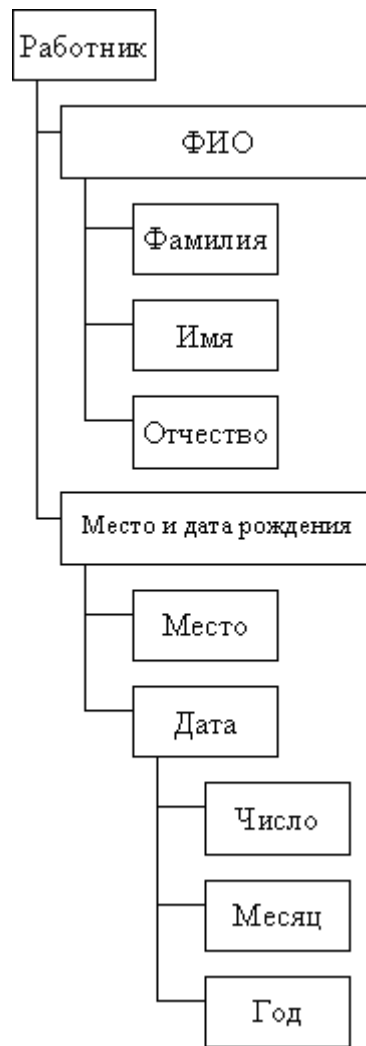


Рисунок 3

Смысл иерархичности заключается в следующем: реальные объекты имеют набор свойств, позволяющих их идентифицировать, и могут состоять из подобъектов, которые, в свою очередь, тоже обладают свойствами.

Сложный информационный объект можно представить в виде дерева с корнем. Корень дерева — это сам объект, вершины — свойства объекта и его подобъекты со своими свойствами.

Надо отметить, что структура объекта не менее важна, чем данные, его составляющие. Изменив внутреннюю структуру объекта, даже оставив неизменными данные, мы получим в результате совсем другой объект. Наличие иерархичной структуры облегчает классификацию, поиск и обработку информации.

3.2.1. Модель данных

Модель данных отображает структуру данных формы и представляется в виде дерева. Понятие *модель данных* полностью соответствует понятию *модель содержания*. В настоящем документе употребляется понятие *модель содержания*.

Вершинам дерева соответствуют простые данные, структуры или массивы (см. п. 3.2.2).

3.2.2. Элемент данных

Элементы данных отображают модель данных, ее структуру, свойства и тип отображаемой на экранной форме информации.

Модель содержания состоит из набора разнотипных элементов, сформированных и сгруппированных по типу отображаемой информации.

Как было описано выше (см. п.3.2.1), существует четыре типа элемента данных:

- модель содержания (корень дерева);
- структура;
- массив;
- простое данное.

Корень дерева выполняет одну функцию — объединение (завершение) структурной схемы.

Остальные элементы данных делятся на две группы (рис. 4): *простые* (простое данное) и *составные* (структура и массив). Составные элементы делятся на подгруппы: неопределенного заранее размера (*массив*) и определенного размера (*структура*).

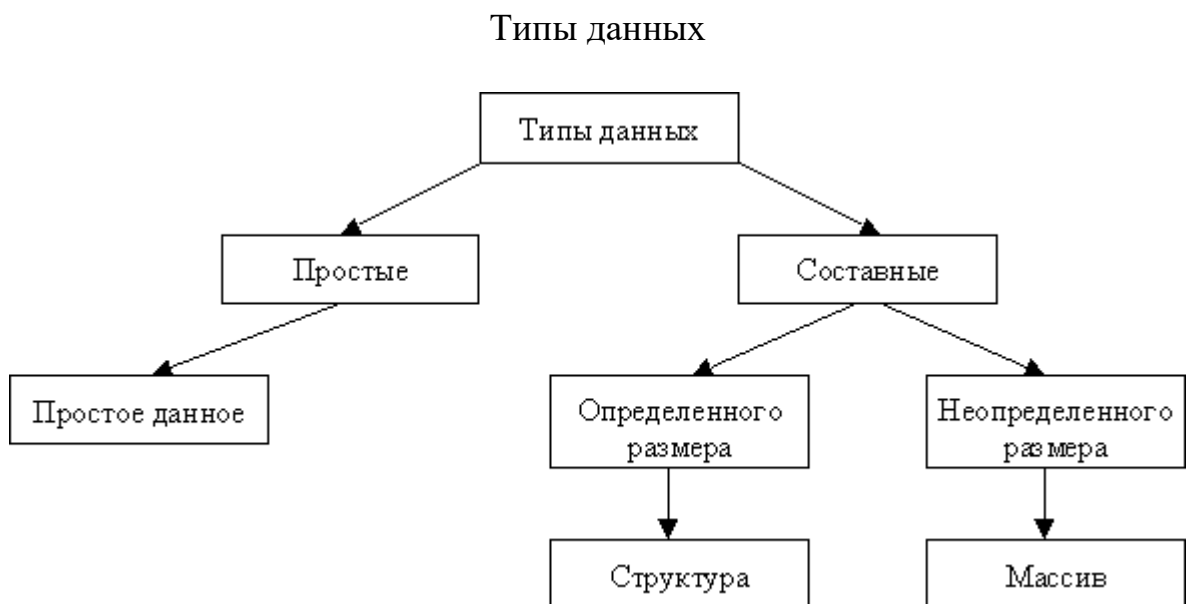


Рисунок 4

Для каждого элемента данных задаются его свойства — имя элемента данных, текст всплывающей подсказки к этому элементу данных, а также тип данных, которые может хранить элемент (текст, целое или вещественное число, массив).

Элементы данных связываются с соответствующими полями статической формы. При заполнении статической формы осуществляется автоматическая проверка введенных в эти поля значений на соответствие типам данных, выбранному диапазону значений и т. д., после чего значения, введенные пользователем в поля статической формы, присваиваются соответствующим элементам данных модели содержания.

3.2.3. Структура

Элемент данных типа «структура» представляет собой объединение других элементов данных и используется главным образом для группировки элементов данных. Структура может включать в себя произвольный набор элементов данных типа «простое данное», «массив» и «структура». Последнее позволяет создавать иерархию данных произвольной глубины вложенности.

Каждому элементу данных типа «структура» в модели содержания может соответствовать своя статическая форма (указывается в свойстве **Форма** окна элементы модели содержания). Если модель содержания состоит из нескольких структур верхнего уровня, то соответствующие им статические формы располагаются непосредственно друг под другом.

Структура отображается на статической форме как панель или область, которая включает в себя все элементы, входящие в нее в модели содержания (см. рис.4)

3.2.4. Массив

Элемент данных типа массив используется для хранения большого количества однотипных структурированных данных, например, список прикрепленных к документу файлов. Для массива задается описание структуры хранимых данных. Массив отображается на форме как таблица или динамическая списковая форма.

3.2.5. Простое данное

Элемент данных типа «Простое данное» используется для ввода простых единичных данных типа «строка», «число», «дата», «логическое (имеющее значения **Правда** или **Ложь**) значение». На статической форме такому типу данных сопоставляются простые поля ввода, выпадающие и комбинированные списки. Простое данное отображается на форме как *текстовое поле, поле со списком.*

3.3. Форма

3.3.1. Экранная форма

С помощью модуля «Дизайнер форм» создаются или редактируются формы, которые именуются экранными.

Экранная форма строится на основе двух компонент — *модели содержания*, описывающей модель данных экранной формы, и *статической формы*, описывающей визуальное представление и расположение элементов на экранной форме.

Шаблоны отображения данных представляют собой файлы формата xml (далее также — файлы настройки). С помощью файлов xml возможно описание, настройка, а также конфигурация электронной формы.

4. НАСТРОЙКА ШАБЛОНОВ ЛЕГКОГО КЛИЕНТА

4.1. Расположение файлов настройки

Файлы настройки легкого клиента по умолчанию располагаются в папке «C:\Program Files\Cognitive\Euphrat\Server\Templates».

Папка «Templates» содержит список шаблонов, то есть список элементов, которые содержит в себе такие данные как, определение где содержаться данные для замещения вставок в шаблоне, где содержаться шаблоны и куда загружать получившиеся документы.

Эта папка содержит следующие папки:

- «Administration» — папка, которая содержит такие пункты меню, как «Группы доступа» где отображается список групп доступа пользователя, «Пользователи» и «Поисковые разделы»;
- «Dictionary» — папка, которая содержит набор заданных значений (справочник);
- «Document» — папка, которая содержит шаблоны отображения регистрационных и контрольных карточек документов;
- «Plugins» — папка, которая может содержать плагины для легкого клиента (см.п. 4.6);
- «Reports» — папка, которая содержит шаблоны отображения секции **Параметры отчета** вкладки **Создать отчет** для отчетов различных типов;
- «SearchForm» — папка, предназначена для хранения шаблонов поисковых форм, содержит шаблоны отображения форм поиска по документам или задачам на вкладке Поиск;
- «SearchResult» — папка, которая содержит шаблоны отображения списков документов или задач на вкладках второго уровня, на вкладке **Поиск** и на вкладке **Папки**, списка связанных документов на странице документа (см. п. 5.7); для хранения шаблонов результата поиска;
- «Upload» — папка, для хранения настроек для загрузки данных.

Примечание 2. Для понимания отсылок к элементам внешнего вида страниц легкого клиента (например, «вкладки второго уровня», «секция **Маршруты**») ознакомьтесь с документом «Легкий клиент. Руководство пользователя».

4.2. Основные понятия

Основным понятием при работе с шаблонами отображения является поле. Полем называется минимальный элемент описания объекта, с помощью которого можно указать на какое-либо его свойство. Под объектом следует понимать документ, задачу и т. п., в зависимости от назначения шаблона. Внешний вид поля может отличаться в зависимости от назначения шаблона. В общем случае поле имеет следующий вид:

```
<имя_тега_поля fieldType="Тип поля" key="Ключевое наименование поля"
name="Название поля" valueType="Тип значения поля" />
```

Имя тега поля зависит от шаблона отображения. Значения атрибутов поля описаны в таблице 2.

В шаблонах отображения, помимо атрибутов, перечисленных в таблицах 2 и 3, могут быть использованы дополнительные атрибуты.

Таблица 2— Значения атрибутов поля

Название атрибута	Значение
fieldType	Тип поля. Может принимать одно из следующих значений: <ul style="list-style-type: none"> - «attribute» — реквизиты для поиска документов или задач на вкладке Поиск, а также реквизиты списка найденных документов или задач; - «field» — любое поле модели содержания регистрационной (контрольной) карточки документа; - «property» — свойства объекта, то есть данные, которые невозможно получить через поля модели содержания регистрационной (контрольной) карточки документа, например, ход исполнения задачи
key	Ключевое наименование поля. В зависимости от типа поля может принимать одно из следующих значений: <ul style="list-style-type: none"> - системное наименование поля для типа «attribute»; - полный путь к полю модели содержания регистрационной (контрольной) карточки документа для типа «field»; - одно из значений, перечисленных в таблице 4, для типа «property»
name	Название (отображаемое имя) поля. Может принимать любое значение по усмотрению пользователя

valueType	<p>Тип значения поля. Может принимать одно из следующих значений:</p> <ul style="list-style-type: none"> - «Text» — текстовое однострочное поле; - «LongText» — текстовое многострочное поле; - «Flow» — поток; - «AddressBook» — учетная запись пользователя; - «Date» — дата (без времени); - «Time» — время; - «DateTime» — дата и время; - «Boolean» — логическая переменная типа «флажок»; - «Money» — переменная денежного типа
-----------	---

Таблица 3— Значения атрибута **key** для поля типа **property**

Значение	Описание
DocumentPerformer	Имя ответственного по документу с аватаром
DocumentObserver	Имя контролирующего по документу с аватаром
DocumentDueTime	Срок исполнения документа
ActivityState	Состояние задачи, которое отображается в списках задач на вкладках второго уровня, в том числе в списке найденных задач на вкладке Поиск
DocumentActivityState	Состояние задачи, которое отображается на вкладке Маршруты на странице документа
ActivityImage	Значок задачи
ActivityName	Название задачи со значком
WFLog	Ход исполнения задачи
ActivityPerformers	Список имен исполнителей задачи с аватарами

4.3. Отображение регистрационной и контрольной карточек папки «Templates». Краткое описание папки «Administration»

Папка «Administration» содержит следующие пункты меню:

1. Группы доступа. Где отображается список групп доступа пользователя
2. Пользователи. Данный раздел появляется если включена самостоятельная регистрация пользователей.
3. Поисковые разделы. Данные разделы настраиваются аналогичным образом, как и меню страницы поиска, только в отдельном xml файле.

Для каждого пункта меню должны существовать конфигурация результата поиска и поисковой формы в папках «Templates\SearchResult» и «Templates\SearchForm» соответственно.

4.4. Краткое описание папки «Dictionary»

Папка «Dictionary» папка, которая содержит набор заданных значений (справочник), в том числе храниться описание шаблонов пополняемых справочников.

4.5. Краткое описание папки «Document»

Папка «Document» содержит шаблоны отображения регистрационных и контрольных карточек документов. Назначение папок и файлов приведено в таблице 4.

Таблица 4— Назначение папок и файлов в папке «Document»

Название папки или файла		Назначение
Attachments		Папка для хранения шаблонов
	CustomTemplate	Папка для хранения шаблонов отображения контрольных карточек документов для отдельных потоков или моделей содержания, созданных администратором комплекса дополнительно
Links		Шаблон отображения списка документов, связанных с данным
	CustomTemplates	Папка для хранения шаблонов отображения контрольных карточек документов для отдельных потоков или моделей содержания, созданных администратором комплекса дополнительно
	SearchForm	Папка содержит шаблоны отображения форм поиска

		документов и задач на вкладке Поиск
RegCard		Папка для хранения шаблонов отображения регистрационных карточек документов
	CustomTemplates	Папка для хранения шаблонов отображения регистрационных карточек документов для отдельных потоков или моделей содержания
	_Sample.xml	Файл «образец» для создания шаблонов отображения регистрационных карточек документов для отдельных потоков или моделей содержания
	Flows.xml	Файл, в котором указывается соответствие файлов настройки (т. е. шаблонов отображения) регистрационных карточек документов конкретным потокам
	RegCard.xml	Шаблон отображения регистрационных карточек документов по умолчанию
	RegCardFields	Папка для хранения карточки, настройки полей модального окна при согласовании
RegCardFields	CustomTemplates	Папка для хранения шаблонов для настройки отображения полей документа при каких-либо действиях
	RegCardFields	Папка в которой находится описание списка полей карточки документа представляет собой xml-разметку, на основе которой формируется видимый пользователю список.
Activities.xml		Набор колонок по умолчанию для отображения деятельности в папках или результатах поиска
DigitalSignatures.xml		Описание таблицы подписей документа представляет собой xml-разметку, на основе которой формируется видимая пользователю таблица.

DocumentActions.xml		Описание настройки фильтров представляет собой xml-разметку, на основе которой формируется настройка видимости кнопок верхней панели
DocumentsGroupActions.xml		В папке описываются кнопки для массовых действий (например, есть возможность несколько мелких позиций объединить и создать общую позицию), т.е. описывает, кнопки для массовых действий
ExcludedFlows.xml		В папке описываются потоки, которые не отображаются по которым нельзя создать новый документ
FieldRestrictions.xml		В папке описывается возможность не предоставление выбора определенное словарное значение для какого-либо поля(полей) в зависимости от значений реквизитов документа или связанных документов
Hooks.xml		Файл настройки переноса данных
MassChecking.xml		Файл представляет ограничение на создание документов в зависимости от определенных значений полей
ProcessActions		В данной папке прописываются действия, выполняемые в рамках процесса (например, если доступен процесс рассмотрения заявки мы можем снять с контроля, осуществить действие снятие с контроля (в модуле развития)
RoutesVisibilitySettings		Файл представляет настройку серверной валидации на основе правил отображения полей на форме
Название папки или файла		Назначение
<i>Корень папки «Document»</i>		

	Activities.xml	Шаблон отображения задач в секции Маршруты на странице документа (данный шаблон относится к шаблонам отображения списков)
	ExcludedFlows.xml	Файл, содержащий список шаблонов, регистрацию документов в которых с помощью легкого клиента следует сделать невозможной

4.6. Краткое описание папки «Plugins»

«Plugins» — папка, которая может содержать плагины для легкого клиента.

Существует возможность расширения функциональности легкого клиента путем добавления к нему плагинов. *Плагин* — это дополнительный программный модуль, который размещают в специальной папке для уже развернутого на компьютере, где установлен сервер приложений, программного обеспечения легкого клиента. Таким образом, добавление плагина не требует переустановки легкого клиента. По вопросам создания плагинов к легкому клиенту обращайтесь в службу технической поддержки Cognitive Technologies.

Подключение плагина выполняется согласно инструкциям, полученным в службе технической поддержки.

4.6.1. Описание шаблона Назначение ответственного руководителя, исполнителя

В комплексе «Е1-Закупки» предусмотрена настройка определенных групп пользователей. Для удобства настройки прав доступа к документам и потокам пользователи объединяются в группы.

Роль применяется в качестве аналога учетной записи пользователя и может использоваться везде, где указывается имя сотрудника: исполнитель или контролер поручений, получатель сообщений, при назначении прав на отдельные документы и потоки документов и т. п.

Администратор документооборота имеет возможность переназначить исполнение роли другому сотруднику.

Пример части документа назначения ответственного руководителя

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <access>
    <item name="Назначить исполнителя" type="AccessGroup"/>
  </access>
</root>
```

```

</access>
<flows>
  <flow name="Заявки на размещение закупки"/>
</flows>
<task template="Назначить ответственного исполнителя"/>
<view>
  <template>
    <rows>
      <row>
        <column>
          <field name="Выберите исполнителя отдела планирования" isLabel="true" />
        </column>
      </row>
      <row>
        <column>
          <field fieldType="field"
key="|Document|Ответственный_руководитель|Ответственный_руководитель_ФИО"
name="Ответственный_руководитель" isValue="true" valueType="addressbook" edit-
name="IspName" edit-addressbook-groups = "" address-book-tool = "IspName" edit-
required="true" address-book-tool-selectable = "AccessGroup, Person, 'Начальник отдела
УО'|currentOrganization"/>
        </column>
      </row>
    </rows>
  </template>
</view>
</root>

```

Поле с атрибутом `addressbook` предоставляет собой выбор ответственного, в свойстве `edit-addressbook-groups` задается перечисление типов адресных книг, которые будут присутствовать в контроле. В свойстве `address-book-tool-selectable` указывается объекты для выбора, т.е. группы доступа и сотрудники, в данном случае ответственный руководитель.

Шаблон отображения секции **Назначения ответственного исполнителя** имеет структуру, аналогичную структуре шаблонов отображения секции **Назначения ответственного руководителя**. В блоке ответственного исполнителя перечисляются поля, которые содержат соответствующую шаблону форму ввода параметров.

Пример части документа назначения ответственного исполнителя:

```

<?xml version="1.0" encoding="utf-8" ?>
<root>
  <access>
    <item name="Назначить исполнителя" type="AccessGroup"/>
  </access>
  <flows>
    <flow name="Заявки на размещение закупки"/>
  </flows>
  <task template="Назначить исполнителя"/>
  <view>

```

```

<template>
  <rows>
    <row>
      <column>
        <field name="Выберите исполнителя отдела закупок" isLabel="true" />
      </column>
    </row>
    <row>
      <column>
        <field fieldType="field"
key="|Document|Ответственный_исполнитель|Ответственный_исполнитель_ФИО"
name="Ответственный исполнитель" isValue="true" valueType="addressbook" edit-
name="IspName1" edit-addressbook-groups = "" address-book-tool = "IspName1" edit-
required="true" address-book-tool-selectable = "AccessGroup, Person, 'Исполнитель
УО'|currentOrganization"/>
      </column>
    </row>
  </rows>
</template>
</view>
</root>

```

4.6.2. Описание шаблона Регистрация заявки

После процесса создания заявки ее рассмотрения и отправки в уполномоченный орган происходит дальнейшая регистрация и рассмотрение заявки на размещение закупки в ГРБС или УО.

Пример части документа регистрации заявки:

```

<?xml version="1.0" encoding="utf-8" ?>
<root>
  <access>
    <item name="Секретарь УО" type="AccessGroup"/>
  </access>
  <flows>
    <flow name="Заявки на размещение закупки"/>
  </flows>
  <task template="Регистрация заявки"/>
  <view>
    <template>
      <rows>
        <row>
          <column>
            <field name="Регистрационный номер" isLabel="true" />
            <field fieldType="field" key="|Document|Регистрационный_номер"
name="Регистрационный номер" valueType="text" isValue="true" edit-name="regnumorder"
/>
          </column>
        </row>
      </rows>
    </template>
  </view>
</root>

```

```
</view>
</root>
```

В шаблонах регистрации заявки именем тега поля является **column**. Каждое поле описывает один столбец. Теги **<columns>** и **</columns>** указывают, где начинается и где заканчивается перечисление полей.

В шаблонах регистрации заявки допускается исполнение следующих атрибутов:

- **isValue** — При значении, равном «true», будет отображаться только значение поля;

- **edit-name** — Уникальное название поля, с помощью которого автоматически определяется отредактированное значение, которое следует присвоить полю.

4.6.3. Описание шаблона Назначение исполнителя правового отдела

Шаблоны отображения форм **Назначения исполнителя правового отдела**, **Назначение ответственного исполнителя 1**, **Назначение исполнителя 1** имеют однотипную структуру. Примеры шаблонов назначения исполнителей представлены ниже

Пример «Назначения исполнителя правового отдела»:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <access>
    <item name="Начальник правового отдела" type="AccessGroup"/>
  </access>
  <flows>
    <flow name="Извещения о закупочной процедуре"/>
  </flows>
  <task template="Назначение иполнителя правового отдела"/>
  <view>
    <template>
      <rows>
        <row>
          <column>
            <field name="Ответственный исполнитель правового отдела" isLabel="true" />

            <field fieldType="field"
key="|Document|Ответственный_юрист|Ответственный_юрист_ФИО" name="Ответственный
юрист_Name" isValue="true" valueType="addressbook" edit-name="IspName1Name" edit-
address-book-groups = "" address-book-tool = "IspName1Name" edit-required="true"
address-book-tool-selectable = "AccessGroup, Person, 'Исполнитель
УО'|currentOrganization"/>
          </column>
        </row>
      </rows>
    </template>
```

```
</view>
</root>
```

Пример «Назначение ответственного исполнителя 1»:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <flows>
    <flow name="Заявка на размещение закупки 223ФЗ"/>
  </flows>
  <task template="Назначить ответственного исполнителя"/>
  <view>
    <template>
      <rows>
        <row>
          <column>
            <field name="Выберите исполнителя отдела планирования" isLabel="true" />
          </column>
        </row>
        <row>
          <column>
            <field fieldType="field"
key="|Document|Ответственный_руководитель|Ответственный_руководитель_ФИО"
name="Ответственный_руководитель" isValue="true" valueType="addressbook" edit-
name="IspName" edit-addressbook-groups = "" address-book-tool = "IspName" edit-
required="true" address-book-tool-selectable = "AccessGroup, Person, 'Начальник отдела
УО по 223-ФЗ'|Organization"/>
          </column>
        </row>
      </rows>
    </template>
  </view>
</root>
```

Пример «Назначение исполнителя 1»:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <flows>
    <flow name="Заявка на размещение закупки 223ФЗ"/>
  </flows>
  <task template="Назначить исполнителя"/>
  <view>
    <template>
      <rows>
        <row>
          <column>
            <field name="Выберите исполнителя отдела закупок" isLabel="true" />
          </column>
        </row>
        <row>
          <column>
```

```

        <field fieldType="field"
key="|Document|Ответственный_исполнитель|Ответственный_исполнитель_ФИО"
name="Ответственный_исполнитель" isValue="true" valueType="addressbook" edit-
name="IspName1" edit-addressbook-groups = "" address-book-tool = "IspName1" edit-
required="true" address-book-tool-selectable = "AccessGroup, Person, 'Исполнитель УО
по 223-ФЗ'|Organization"/>

    </column>
</row>
</rows>
</template>
</view>
</root>

```

В шаблонах отображения форм поиска именем тега поля является **row**. Каждое поле описывает один столбец. Теги **<rows>** и **</rows>** указывают, где начинается и где заканчивается перечисление полей.

Поле с атрибутом **addressbook** предоставляет собой выбор ответственного, используемый атрибут **edit-addressbook-groups** задает перечисление типов элементов адресных книг, которые будут присутствовать в контроле и которые будут доступны для выбора. В свойстве **address-book-tool-selectable** также указывается объекты для выбора, т.е. в данном случае группы доступа и сотрудники юридического отдела для назначения исполнителей правого отдела, а также группы доступа ответственных исполнителей

4.6.4. Структура файла настройки «Flows.xml»

Шаблоны секции **Отправление уведомления** имеют структуру, аналогичную структуре шаблонов **Отзыв заявки**. В файле настройки «ExcludedFlows.xml» именем тега поля является **flow**. Каждое поле описывает один поток. Теги **<flows>** и **</flows>** указывают, где начинается и где заканчивается перечисление полей.

Отправить уведомление на корректировку заявки заказчику имеет возможность выполнить уполномоченный орган. Исполнитель УО возвращает заявку на размещение исполнителю заказчика с указанием необходимых корректировок и срока исправления.

Пример шаблона Отправления уведомления

```

<?xml version="1.0" encoding="utf-8" ?>
<root>
  <flows>
    <flow name="Позиции плана-графика"/>
  </flows>
  <process name="Рассмотрение заявки"/>
  <requisites>
    <requisite path="|Document|Отправка_уведомления" value="1"/>
    <requisite path="|Document|Статус" value="Ожидание ответа"/>
  </requisites>

```

```
</root>
```

Исполнитель заказчика, либо ответственный исполнитель отдела закупок УО (если данная закупка проходит через УО) имеет возможность выполнить отзыв заявки.

Пример шаблона **Отзыв заявки**

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <process name="Рассмотрение заявки"/>
  <requisites>
    <requisite path="|Document|Отзыв_заявки" value="1"/>
    <requisite path="|Document|Статус" value="Отзыв заявки"/>
  </requisites>
</root>
```

4.7. Краткое описание папки «Reports»

«**Reports**» — папка, которая содержит шаблоны отображения секции **Параметры отчета** вкладки **Создать отчет** для отчетов различных типов.

Папка содержит шаблоны отображения секции **Параметры отчета** вкладки **Создать отчет** для форм ввода параметров, которые используются при формировании отчетов различных типов.

4.8. Краткое описание папки «SearchForm»

Папка для хранения шаблонов поисковых форм, это форма в виде поисковых значений, при заполнении, которых выходит результат искомого.

Папка содержит шаблоны отображения форм поиска документов и задач на вкладке **Поиск**. Назначение файлов в папке представлено в таблице 5.

Таблица 5— Назначение файлов в папке «SearchForm»

Название папки или файла	Назначение
Activities.xml	Шаблон отображения формы поиска задач на вкладке Поиск (вкладка второго уровня По задачам)
Documents.xml	Шаблон отображения формы поиска документов на вкладке Поиск (вкладка второго уровня По документам)

4.9. Краткое описание папки «SearchResult»

Папка «SearchResult» содержит шаблоны отображения списков документов или задач на вкладках второго уровня, на вкладке **Поиск** и на вкладке **Папки**, списка связанных документов на странице документа. Назначение папок и файлов в папке «SearchResult» представлено в таблице 6.

Таблица 6— Назначение папок и файлов в папке «SearchResult»

Название папки или файла	Назначение
Control	Папка для хранения шаблонов отображения списков документов или задач на вкладке На контроле
Название папки или файла	Назначение
Folder	Папка для хранения шаблонов отображения списков документов или задач на вкладке Папки
FolderTemplates.xml	Файл, в котором указывается соответствие файлов настройки (т. е. шаблонов отображения) списков документов и задач на вкладке Папки отдельным папкам
Activities.xml	Шаблон отображения списка задач на вкладке Папки
Documents.xml	Шаблон отображения списка документов на вкладке Папки
Performer	Папка для хранения шаблонов отображения списков документов или задач на вкладке На исполнении
Activities.xml	Шаблон отображения списка задач на вкладке На исполнении
Documents.xml	Шаблон отображения списка документов на вкладке На исполнении
Positions	Папка для хранения шаблонов отображения формы позиций на вкладке Мои позиции
PurshareOrders	Папка для хранения шаблонов отображения формы заявки на закупку на вкладке Новая заявка
Search	Папка для хранения шаблонов отображения списков найденных документов или задач на вкладке Поиск
Activities.xml	Шаблон отображения списка найденных задач на вкладке Поиск
Documents.xml	Шаблон отображения списка найденных документов на вкладке Поиск
<i>Корень папки «SearchResult»</i>	

Activities.xml	Общий шаблон отображения списков задач на вкладках второго уровня
Название папки или файла	Назначение
Documents.xml	Общий шаблон отображения списков документов на вкладках второго уровня
DocumentLinks.xml	Шаблон отображения списка связанных документов на странице документа

Существует возможность создавать отдельные шаблоны отображения списков документов или задач (в отдельных файлах настройки) для конкретных папок.

4.10. Краткое описание папки «Upload»

«Upload» - папка для хранения настроек для загрузки данных с помощью сервиса.

Settings - Описание таблиц связанных документов представляет собой xml-разметку, на основе которой формируется видимая пользователю таблица.

4.11. Краткое описание папки «Scripts»

«Scripts» описывает скрипты, подгружаемые на форму. «Scripts» предназначен для описания скриптов. «Scripts» располагается во внешнем файле и связывается с любым xml-документом. Такой подход позволяет использовать одни и те же общие функции на многих страницах и ускоряет их загрузку, для написания скрипта используется язык jQuery.

4.11.1. Настройка использования скриптов

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.

Чтобы указать один или несколько скриптов, которые будут выполняться при просмотре или редактировании регистрационной (контрольной) карточки, в структуру шаблона отображения карточки следует

включить *блок подключения скриптов*. Теги `<scripts>` и `</scripts>` указывают, где начинается и где заканчивается перечисление скриптов.

Скрипты структурируются по определенным папкам в том числе скрипты для форм поиска для отчетов для регистрационных карточек.

В блоке подключения скриптов 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>
```

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 данных.

В блоке дополнений именем поля (см. п. 4.2) 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>
```

4.12. Выбор шаблона для отображения пользовательской папки

Данный раздел относится только для выбора шаблона общей папки. Личные папки не имеют возможность выбора шаблона, и для них используются умолчательные шаблоны.

Первоначально необходимо открыть файл

"/Templates/SearchResult/Folder/FolderTemplates.xml". Файл имеет структуру, аналогичную структуре папок.

Пример части шаблона отображения пользовательской папки:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <folders>
    < folder template="Common.xml" name="Общие папки">
      <folder template="CustomTemplates/FolderLot.xml" name="Мои лоты"/>
      <folder template="CustomTemplates/FolderItem.xml" name="Мои позиции"/>
      <folder template="CustomTemplates/FolderPurchase.xml" name="Мои заявки
на закупку"/>
      <folder template="CustomTemplates/FolderNotice.xml" name="Мои
извещения"/>
      <folder template="CustomTemplates/FolderDogovor.xml" name="Мои
договоры"/>
    </folder>
  </folders>
</root>
```

Настройки шаблонов для папок находятся внутри элемента **folders**. Каждый элемент **folder** содержит в себе настройки отображения конкретной папки: *name* - название папки на своем уровне иерархии, *template* - путь к файлу с шаблоном для отображения папки (можно использовать подпапки, например/customTemplates/folderTemplate.xml).

В случае, если атрибут *template* для элемента **folder** не задан, либо файл «FolderTemplates.xml» отсутствует, будет использоваться умолчательный шаблон, находящийся в «/Templates/SearchResult/Folder» - «Activities.xml» для папок с деятельностью, и «Documents.xml» для папок с документами соответственно. В случае, если данные умолчательного шаблона отсутствуют, то будут использоваться шаблоны из «/Templates/SearchResult».

Если не указано другое, то в качестве шаблонов для отображения регистрационной карточки документа будут использоваться шаблоны «Templates/Document/RegCard/RegCard.xml». Это поведение можно изменить, указав конкретный шаблон для конкретной модели содержания или даже для конкретного потока. Рассмотрим на примере регистрационной карточки.

Выбор шаблона по имени модели содержания, происходит следующим образом, например, если у документа модель содержания называется NInput, то перед тем как начать использовать шаблон «Templates/Document/RegCard/RegCard.xml» система проверит наличие файла «Templates/Document/RegCard/CustomTemplates/NInput.xml», который и будет использоваться вместо умолчательного, если будет найден.

В случае если не удалось найти шаблон по имени модели содержания, то она будет искать его по имени потока. Для этого система рассмотрит содержимое файла «Templates/Document/RegCard/Flows.xml» и будет искать

соответствия в нем. Регистрационная карточка может быть разработана таким образом, что заполнение одного или нескольких обязательных полей осуществляется с помощью формул или скриптов.

Пример Файла соответствия отображения пользовательской папки:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <flows>
    <flow name="Внутренние документы" template="Internal.xml" />
    <flow name="Входящие документы" template="Input.xml" />
  </flows>
</root>
```

В файле настройки «ExcludedFlows.xml» именем тега поля является **flow**. Каждое поле описывает один поток, который следует исключить из списка **Выберите поток** вкладки **Новый документ**. Теги **<flows>** и **</flows>** указывают, где начинается и где заканчивается перечисление полей.

В папках возможен расширенный поиск. Для поисковых форм есть возможность задавать дополнительный набор полей для поиска (фильтрации), который по умолчанию будет скрыт. При задании набора дополнительных полей рядом с кнопкой **Найти** появится кнопка **Расширенный поиск**, который отвечает за скрытие и показ дополнительного набора полей.

Пример добавления дополнительных полей в шаблон:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <edit>
  </edit>
  <extraSearchForm>
  </extraSearchForm>
  <search>
    <field fieldValue="поисковый запрос" />
  </search>
</root>
```

На одном уровне с элементом **edit**, в котором описываются поля для поиска, добавляется элемент **extraSearchForm**, структура содержимого, которого совпадает со структурой, используемой в **edit**. При указании дополнительного набора полей их также необходимо учитывать в поисковой строке. Иначе значения данных полей не будут учитываться.

4.13. Выбор шаблона для отображения связанных документов

По умолчанию используется файл.

\Templates\Document\Links\CustomTemplates\DocumentLinks.xml. Также есть возможность изменить данное поведение, для этого необходимо указать конкретный шаблон для конкретной модели содержания, либо для конкретного потока.

Выбор шаблона по имени модели содержания, происходит аналогично выбору шаблона для отображения пользовательской папки, т.е. если в случае, модель у потока документа называется *NInput*, то система осуществит проверку наличия файла по пути \Templates\Document\Links\CustomTemplates\NInput.xml, который и будет использоваться, если будет найден.

Если система не находит шаблон по имени модели содержания, то она попытается найти ее по имени потока. Для этого она рассмотрит содержимое файла \Templates\Document\Links\Flows.xml и будет искать соответствия в нем.

Пример Файла соответствия для отображения связанных документов:

```
<<?xml version="1.0" encoding="utf-8" ?>
<root>
  <flows>
    <flow name="Лоты" baseFolder="Lots">
      <filter flow="Извещения о закупочной процедуре" template="NoticesView.xml"/>
      <filter flow="Позиции" template="ItemsView.xml"/>
    </flow>
    <flow name="Входящие документы" baseFolder="InputDocuments">
      <filter flow="Лоты" template="LotsView.xml"/>
      <filter flow="Позиции" template="ItemsView.xml"/>
    </flow>
  </flows>
</root>
```

Каждый элемент **flow** содержит описание того, какой файл использовать для отображения связанных документов того или иного потока. Поле с атрибутом **name** предоставляет собой название потока, для отображения связок которого будет выбираться файл, атрибут **baseFolder** это название папки относительно пути \Templates\Document\Links\.

Элемент **flow** может содержать некоторое количество элементов **filter**, каждый из которых определяет, какой файл необходимо использовать для отображения связок того или иного потока. Поле с атрибутом **flow** имеет название потока связанных документов, для которого выбирается файл, атрибут **template** это название файла, который надо использовать.

Пример формата шаблона отображения таблицы связанных документов:

```

<?xml version="1.0" encoding="utf-8" ?>
<root>
  <table flow="Заявки на закупку" childLinksFlow="Позиции">
    <settings>
      <setting key="canModify" value="true" />
      <setting key="canCreateLinks" value="true" />
      <setting key="terminalRoute" value="Удаление заявки из извещения"/>
    </settings>
    <columns>
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/longtext_Наименование_объекта_закупки" name="Наименование
закупки" valueType="longtext" weight="55" />
      <column fieldType="attribute" key="doc_RegCard/rc_Index/money_НМЦК"
name="НМЦК" valueType="money" weight="25" />
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/integer_Планируемый_год_размещения" name="Планируемый год
размещения" valueType="text" weight="20" />
    </columns>
    <sorts>
      <sort key="doc_RegCard/rc_Index/text_Статус" direction="ascending"
priorityValues="В работе; Черновик"/>
    </sorts>
  </table>
  <table flow="Запросы на разъяснение">
    <settings>
      <setting key="canModify" value="true" />
      <setting key="canCreateLinks" value="true" />
    </settings>
    <columns>
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/text_Регистрационный_номер" name="Рег. номер"
valueType="text" weight="20" isOptional="true" />
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/date_Дата_регистрации" name="Дата" valueType="date"
weight="25" />
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/longtext_Описание_запроса" name="Описание запроса"
valueType="longtext" weight="55" />
    </columns>
  </table>
  <table>
    <settings>
      <setting key="canModify" value="true" />
      <setting key="canCreateLinks" value="true" />
      <setting key="terminalRoute" value="Удаление заявки из извещения"/>
    </settings>
    <columns>
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/longtext_Наименование_объекта_закупки" name="Наименование
закупки" valueType="longtext" weight="55" />
      <column fieldType="attribute" key="doc_RegCard/rc_Index/money_НМЦК"
name="НМЦК" valueType="money" weight="25" />
    </columns>
  </table>

```

```

        <column fieldType="attribute"
key="doc_RegCard/rc_Index/integer_Планируемый_год_размещения" name="Планируемый год
размещения" valueType="text" weight="20" />
    </columns>
</table>

</root>

```

Каждый элемент **table** определяет настройки и набор колонок, которые необходимо отображать в таблице.

Элемент **table** использует такие атрибуты как:

- **flow** - определяет, связанные документы какого потока будут отображаться в таблице. Если значение не указывать, то в таблице будут все связанные документы;

- **isInvisible** - при установке значения *true*, такая таблица не будет отображаться. Используется совместно с атрибутом *childLinksFlow*;

- **childLinksFlow** - определяет, связанные документы какого потока необходимо отображать у связанного документа как подчиненную таблицу. Если атрибут не указан - подчиненная таблица не будет отображаться. Если атрибут указан, но он пустой - в подчиненной таблице будут связанные документы текущего связанного документа.

Если указано какое-либо значение - в подчиненной таблице будут связанные документы указанного потока.

Сортировка в таблице связанных документов.

Внимание! Не рекомендуется настраивать сортировку, когда у документа может быть большое количество связанных документов, в результате чего произойдет длительное открытие страницы с документом.

Список очередностей сортировок определяется в разделе **sorts** у элемента **table**. Каждый элемент определяет сортировку по определенному атрибуту. Атрибут должен быть поисковым и не обязан быть в списке колонок таблицы. Список значений атрибутов:

- **key** - поисковый реквизит, по которому необходимо сортировать. Обязательный атрибут;

- **direction** - направление сортировки. Возможные значения: *ascending* (по возрастанию) и *descending* (по убыванию). Значение по умолчанию – *ascending*;

- **priorityValues** - список значений, разделенных точкой с запятой. Определяет список значений реквизита, элементы с которыми всегда будут выше остальных значений, независимо от направления сортировки.

Пример:

```

<sorts>
    <sort key="doc_RegCard/rc_Index/text_Статус" direction="ascending"
priorityValues="В работе; Черновик"/>
</sorts>

```

В данном примере отображаются все связанные документы, у которых статус «В работе», затем отображаются документы со статусом «Черновик», затем все остальные связки с сортировкой статусов по возрастанию.

Осуществление настроек таблицы связанных документов. В разделе **settings** содержатся настройки таблицы. Каждый элемент **setting** имеет следующий формат:

Пример:

```
<s<setting key="key" value="value" />
```

Поддерживается следующий набор значений для настроек:

- **canModify** - определяет, возможны ли какие-либо действия над связанными документами в таблице и определяет, возможно или нет добавить связанные документы. Допустимые значения - true и false;

- **canCreateLinks** – определяет возможно ли добавление связанных документов. Допустимые значения - true и false. Актуально только при **canModify = true**;

- **terminalRoute** - название маршрута, которое будет запущено при удалении связанного документа **в связанном документе**;

- **terminalMainDocumentRoute** - название маршрута, которое будет запущено при удалении связанного документа **в основном документе**;

- **addMainDocumentRoute** - название маршрута, которое будет запущено при добавлении связанного документа **в основном документе**;

- **addButtonText** - название для кнопки, по которой происходит добавление связанного документа.

Настройка кнопки удаления связанных документов. Для настройки кнопки необходимо использовать раздел **<removeSettings/>** в элементе **<table/>**. В этом разделе присутствуют такие атрибуты как:

- **value** - при значении true, кнопки удаления отображаются, при false - нет. Данный атрибут является обязательным;

- **docCondition** - условие для главного документа, при котором должен применяться атрибут **value**;

- **linkCondition** - условие для связанных документов при котором должен применяться атрибут **value**.

Пример настройки кнопки удаления связанных документов:

```
<removeSettings docCondition="(doc_RegCard/rc_Index/text_Статус = 'Проект')"  
linkCondition="(doc_RegCard/rc_Index/text_Статус = 'Включена в проект плана закупок')"  
value="false" />
```

Таким образом, если документ имеет статус «Проект» и связанный документ имеет статус «Включена в проект плана закупок», то кнопка удаления не отображается, если документ имеет другой статус кнопка отображается.

4.14. Настройка шаблонов отображения регистрационной карточки

4.14.1. Общие сведения

По умолчанию для отображения регистрационной карточки используются файлы настройки «RegCard.xml». Существует возможность создавать отдельные шаблоны (в отдельном файле настройки) для конкретных потоков или моделей содержания. Создание отдельного шаблона для потока может потребоваться в том случае, когда одна и та же модель содержания регистрационной карточки используется для нескольких потоков, а в организации есть потребность настроить специальное отображение данных для регистрационной карточки только для одного из этих потоков. В случае создания отдельного шаблона для потока документов, специальным образом устанавливается соответствие шаблона этому потоку.

Шаблоны отображения для отдельных потоков или моделей содержания сохраняются в папке:

«\Templates\Document\RegCard\CustomTemplates» — шаблоны для регистрационных карточек.

Название файла настройки, созданного для модели содержания, должно совпадать с названием этой модели содержания: «<название модели содержания>.xml». Название файла настройки, созданного для потока, задается произвольно: «<название_файла>.xml», но с учетом того, что оно должно отличаться от названий существующих моделей содержания регистрационных карточек.

Шаблоны отображения регистрационных карточек имеют однотипную структуру. Каждый шаблон включает *блок просмотра* (см. п. 5.10.2) и *блок редактирования* (см. п. 5.11.4).

В шаблоне отображения регистрационной карточки именем тега поля (см. п. 4.2) является **field**.

Расположение полей описывается по строкам. Для обозначения отдельной строки используется пара тегов **<row>** и **</row>**. Для обозначения отдельных столбцов в каждой строке используется пара тегов **<column>** и **</column>**. Теги **<rows>** и **</rows>** указывают, где начинается и где заканчивается перечисление строк и столбцов.

Внимание! После завершения редактирования файлов настройки или файла «Web.config» настоятельно рекомендуется проверить корректность работы легкого клиента (см. документ «Легкий клиент. Руководство пользователя»).

4.14.2. Отображение списка полей карточки документа

Выбор шаблона для отображения полей карточки документа. По умолчанию используется файл \Templates\Document\RegCardFields\RegCardFields.xml.

При указании конкретного шаблона для конкретной модели содержания, либо для конкретного потока, поведение может быть изменено.

Пример файла отображения списка полей карточки документа:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <location type="ApproveActivity">
    <field name="|Key"/>
    <field name="|Document|Плановая_дата_размещения"/>
    <field name="|Document|Регистрационный_номер"/>
  </location>
</root>
```

Каждый элемент location определяет настройки и набор полей, которые необходимо отображать в списке.

Атрибуты элемента location:

- type - определяет, для какого именно действия будет отображаться указанный список полей. В данном случае действие – «Одобрение согласования». Набор полей для отображения в списке полей карточки документа определяется в разделе location. Каждый элемент fields описывает одно поле. Список значений атрибутов:

- key - реквизит карточки документа

Настройки списка полей для отображения

В разделе location содержатся настройки списка. Каждый элемент location имеет следующий формат:

```
<location type="ApproveActivity">
```

- type - определяет действие, для которого будет использован данный шаблон. Допустимые значения:

- ApproveActivity (одобрение согласования).

4.14.3. Таблицы связанных документов

Описание таблиц связанных документов представляет собой xml-разметку, на основе которой формируется видимая пользователю таблица.

Выбор шаблона для отображения связанных документов. По умолчанию используется файл \Templates\Document\Links\CustomTemplates\DocumentLinks.xml. При указании конкретного шаблона для конкретной модели содержания, либо для конкретного потока поведение возможно изменить.

Система так же будет осуществлять проверку наличия файла по пути: `\Templates\Document\Links\CustomTemplates\NInput.xml`, файл будет использован при нахождении.

Пример Файла соответствия таблиц связанных документов:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <flows>
    <flow name="Лоты" baseFolder="Lots">
      <filter flow="Извещения о закупочной процедуре" template="NoticesView.xml"/>
      <filter flow="Позиции" template="ItemsView.xml"/>
    </flow>
    <flow name="Входящие документы" baseFolder="InputDocuments">
      <filter flow="Лоты" template="LotsView.xml"/>
      <filter flow="Позиции" template="ItemsView.xml"/>
    </flow>
  </flows>
</root>
```

Каждый элемент `flow` содержит описание того, какой файл использовать для отображения связанных документов того или иного потока.

Значение атрибутов:

- `baseFolder` - название папки относительно пути `\Templates\Document\Links\`.

Элемент **flow** может содержать некоторое количество элементов **filter**, каждый из которых определяет, какой файл необходимо использовать для отображения связей того или иного потока.

Значение атрибутов:

- `flow` - название потока связанных документов, для которого выбирается файл;

- `template` - название файла, который надо использовать.

В данном примере указано, что в потоке **Лоты** для отображения связанных документов потока **Извещения о закупочной процедуре** надо использовать файл по пути `\Templates\Document\Links\Lots\NoticesView.xml`, а для связанных документов потока **Позиции**

- `\Templates\Document\Links\Lots\ItemsView.xml`. Аналогично для потока **Входящие документы**.

Пример Формата шаблона отображения таблицы связанных документов:

```
< ?xml version="1.0" encoding="utf-8" ?>
<root>
  <table flow="Заявки на закупку" childLinksFlow="Позиции">
    <settings>
      <setting key="canModify" value="true" />
      <setting key="canCreateLinks" value="true" />
      <setting key="terminalRoute" value="Удаление заявки из извещения"/>
    </settings>
```

```

    <columns>
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/longtext_Наименование_объекта_закупки" name="Наименование
закупки" valueType="longtext" weight="55" />
      <column fieldType="attribute" key="doc_RegCard/rc_Index/money_НМЦК"
name="НМЦК" valueType="money" weight="25" />
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/integer_Планируемый_год_размещения" name="Планируемый год
размещения" valueType="text" weight="20" />
    </columns>
    <sorts>
      <sort key="doc_RegCard/rc_Index/text_Статус" direction="ascending"
priorityValues="В работе; Черновик"/>
    </sorts>
  </table>
  <table flow="Запросы на разъяснение">
    <settings>
      <setting key="canModify" value="true" />
      <setting key="canCreateLinks" value="true" />
    </settings>
    <columns>
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/text_Регистрационный_номер" name="Рег. номер"
valueType="text" weight="20" isOptional="true" />
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/date_Дата_регистрации" name="Дата" valueType="date"
weight="25" />
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/longtext_Описание_запроса" name="Описание запроса"
valueType="longtext" weight="55" />
    </columns>
  </table>
  <table>
    <settings>
      <setting key="canModify" value="true" />
      <setting key="canCreateLinks" value="true" />
      <setting key="terminalRoute" value="Удаление заявки из извещения"/>
    </settings>
    <columns>
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/longtext_Наименование_объекта_закупки" name="Наименование
закупки" valueType="longtext" weight="55" />
      <column fieldType="attribute" key="doc_RegCard/rc_Index/money_НМЦК"
name="НМЦК" valueType="money" weight="25" />
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/integer_Планируемый_год_размещения" name="Планируемый год
размещения" valueType="text" weight="20" />
    </columns>
  </table>
</root>

```

Каждый элемент **table** определяет настройки и набор колонок, которые необходимо отображать в таблице.

Атрибуты элемента **table**:

- **flow** - определяет, связанные документы какого потока будут отображаться в таблице. Если значение не будет указано, то в таблице будут все связанные документы;

- **isInvisible** - при установке значения *true*, такая таблица не будет отображаться. Используется совместно с атрибутом *childLinksFlow*;

- **childLinksFlow** - определяет, связанные документы какого потока необходимо отображать у связанного документа как подчиненную таблицу. Если атрибут не указан - подчиненная таблица не будет отображаться. Если атрибут указан, но он пустой - в подчиненной таблице будут связанные документы текущего связанного документа. Если указано какое-либо значение - в подчиненной таблице будут связанные документы указанного потока.

4.14.4. Набор колонок для отображения в таблице связанных документов

Набор колонок для отображения определяется в разделе **columns** у элемента **table**

Каждый элемент **column** описывает одну колонку.

Список значений атрибутов:

- **key** - реквизит документ, поисковый реквизит, либо одно из определенных свойств (не используется при настройке таблиц связанных документов);

- **fieldType** - тип поля. Возможные значения:

- **attribute** - поисковый атрибут. При этом в атрибут **key** необходимо писать поисковый реквизит;

- **field** - реквизит документа. При этом в **key** необходимо писать путь к реквизиту документа;

- **property** - одно из предопределенных свойств. Не используется при настройке таблиц связанных документов;

- **name** - название колонки;

- **valueType** - тип значения. от значения этого атрибута зависит то, как будет представлено значение. Возможные типы значений:

- **text**;

- **longtext**;

- **flow**;

- **addressbook**;

- **date**;

- **datetime**;

- **time**;

- **boolean**;

- **weight** - относительная ширина колонки.

4.14.5. Сортировка в таблице связанных документов

Список очередностей сортировок определяется в разделе **sorts** у элемента **table**. Каждый элемент **sort** определяет сортировку по определенному атрибуту. Атрибут должен быть поисковым и не обязан быть в списке колонок таблицы

Список значений атрибутов:

- **key** - поисковый реквизит, по которому необходимо сортировать. Данный атрибут является обязательным;

- **direction** - направление сортировки. Возможные значения: *ascending* (по возрастанию) и *descending* (по убыванию). Атрибут необязателен. Значение по умолчанию – *ascending*;

- **priorityValues** - список значений, разделенных точкой с запятой. Определяет список значений реквизита, элементы с которыми всегда будут выше остальных значений, независимо от направления сортировки.

Пример сортировки в таблице связанных документов:

```
<sorts>
  <sort key="doc_RegCard/rc_Index/text_Статус" direction="ascending"
priorityValues="В работе; Черновик"/>
</sorts>
```

В данном примере сначала отобразятся все связанные документы, у которых статус «В работе», потом связи со статусом «Черновик», и только потом все остальные связи с сортировкой статусов по возрастанию.

4.14.1. Настройки таблицы связанных документов

В разделе **settings** содержатся настройки таблицы. Каждый элемент **setting** имеет следующий формат:

Пример Формата настройки таблицы связанных документов:

```
<setting key="key" value="value" />
```

На данный момент поддерживается следующий набор настроечных значений:

- **canModify** - определяет, возможны ли какие-либо действия над связанными документами в таблице и определяет, возможно ли добавить связанные документы. Допустимые значения - true и false;

- **canCreateLinks** - можно ли добавлять связанные документы. Допустимые значения - true и false. Актуально только при **canModify = true**;

- **terminalRoute** - название маршрута, которое будет запущено при удалении связанного документа **в связанном документе**;

- **terminalMainDocumentRoute** - название маршрута, которое будет запущено при удалении связанного документа **в основном документе**;

- **addMainDocumentRoute** - название маршрута, которое будет запущено при добавлении связанного документа **в основном документе**;

- addButtonText - название для кнопки, по которой происходит добавление связанного документа.

4.14.2. Настройка кнопки удаления связанных документов

Настройка кнопки удаления связанных документов. Для настройки кнопки удаления необходимо использовать раздел `<removeSettings/>` в элементе `<table/>`. В этом разделе использованы следующие атрибуты

- **value** - при значении true, кнопки удаления отображаются, при false нет. Обязательный атрибут;

- **docCondition** - условие для главного документа, при котором должен применяться атрибут **value**;

- **linkCondition** - условие для связанных документов при котором должен применяться атрибут **value**.

Пример Настройки кнопок удаления связанных документов:

```
<removeSettings docCondition="(doc_RegCard/rc_Index/text_Статус = 'Проект')"  
linkCondition="(doc_RegCard/rc_Index/text_Статус = 'Включена в проект плана закупок')"  
value="false" />
```

т.е если документ имеет статус «Проект» и связанный документ имеет статус «Включена в проект плана закупок», то кнопка удаления не отображается, иначе кнопка отображается.

Пример Шаблона меню поиска:

```
<root>  
<block name="Планирование">  
  <field name="PlanPrev" value="Лоты (архив)" >  
    <access>  
      <group name="УОЗД (планирование)"/>  
      <group name="УОЗД (планирование) Кубаньэнерго"/>  
    </access>  
  </field>  
  <field name="PlanNext" value="Лоты (следующий год)" >  
    <access>  
      <group name="УОЗД ФСК ЕЭС"/>  
      <group name="УОЗД Янтарьэнерго"/>  
    </access>  
  </field>  
</block>  
<scripts>  
  <script file="~/Scripts/Content/Search/searchCustom.js" />  
</scripts>  
</root>
```

Тегом `block` описываются вкладки главного меню слева страницы по вертикали. В него вложены теги `field`, описывающие элементы зависимого меню, отображаемого по горизонтали сверху.

Для каждого элемента, зависимого меню, настраивается доступ с помощью групп доступа. Если ни один элемент зависимого меню не доступен, вкладка главного меню не отображается. В теге `script` указывается путь к файлу, содержащему скрипт, выполняемый после загрузки страниц поиска.

4.14.3. Настройка блока просмотра

В блоке просмотра описываются поля регистрационной карточки, которые отображаются при открытии страницы документа. Отображение состоит из строк `row`, которые располагаются друг под другом. Строки `row` состоят из колонок `column`, которые располагаются рядом друг с другом. Теги `<view>` и `</view>` указывают, где начинается и где заканчивается блок просмотра.

Пример части шаблона отображения регистрационной карточки с блоком просмотра:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
<view>
<rows>
  <row>
    <column>
      <block id="tlcInfo" name="Основные сведения">
        <rows>
          <row>
            <column>
              <field fieldType="field"
key="|Document|Организация_заказчик|Организация_заказчик_наименование"
name="Организация заказчик" valueType="text" />
            </column>
          </row>
          <row>
            <column>
              <table key="|Document|Бюджеты" edit-
name="ist_fin" height="300">
                <column fieldType="field"
key="|Источник_финансирования" valueType="text" isValue="true" name="Источник
финансирования" width="150"/>
                <column fieldType="field"
key="|КБК_бюджетные_средства" name="КБК бюджетные средства" valueType="text"
isValue="true" width="200" />
                <column fieldType="field"
key="|КВР_бюджета_код" name="КВР" valueType="text" isValue="true" width="100"/>
              </table>
            </column>
          </row>
        </rows>
      </block>
    </column>
  </row>
</rows>
</view>
</root>
```

```

        <column fieldType="field"
key="|Год_финансирования_бюджета" name="Год финансирования" valueType="text"
isValue="true" width="100"/>
        <column fieldType="field"
key="|Сумма_лимита_бюджета" name="Сумма лимита" valueType="money" isValue="true"
width="100"/>
        <column fieldType="field"
key="|Запланировано" name="Запланировано" valueType="money" isValue="true"
width="100"/>
        <column fieldType="field"
key="|Остаток_бюджета" name="Остаток" valueType="money" isValue="true" width="100"/>
        <column fieldType="field"
key="|Экономия" name="Экономия" valueType="money" isValue="true" width="100"/>
    </table>
    </column>
</row>
</rows>
</block>
</column>
</row>
</rows>
</view>

```

Внешний вид регистрационной карточки для шаблона с указанной в примере структурой блока просмотра приведен на рис.5.

Пример внешнего вида регистрационной карточки

Рисунок 5

Для корректного отображения полей при открытии страницы документа, в блоке просмотра настоятельно рекомендуется ограничивать количество столбцов в каждой строке:

- не более 4 столбцов с надписями;
- не более 12 столбцов без надписей.

Допускается добавлять неограниченное количество строк в структуру блока просмотра.

4.14.4. Настройка отображения таблиц

В состав блока просмотра шаблона отображения регистрационной карточки, кроме полей, также могут входить таблицы. Теги `<table>` и `</table>` указывают, где начинается и где заканчивается перечисление столбцов таблицы.

Именем тега поля, описывающего столбец таблицы, является **column**. Значение атрибута **name** для каждого поля равно названию соответствующего столбца таблицы в модели содержания регистрационной карточки. Значение атрибута **key** для таблицы (поля) равно полному пути к таблице (столбцу) в модели содержания регистрационной карточки.

Значение атрибута **height** задает максимальную высоту таблицы, а атрибута **width** — ширину столбца таблицы.

Пример части шаблона отображения регистрационной карточки с таблицей:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <table id="ZNZ" flow="Заявки на закупку" childLinksFlow="Позиции"
filter="doc_RegCard/rc_Index/integer_Планируемый_год_размещения = 2018">
    <settings>
      <setting key="canModify" value="true" />
      <setting key="canCreateLinks" value="true" />
      <setting key="terminalRoute" value="Удаление заявки из извещения"/>
    </settings>
    <columns>
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/longtext_Наименование_объекта_закупки" name="Наименование
закупки" valueType="longtext" weight="55" />
      <column fieldType="attribute" key="doc_RegCard/rc_Index/money_НМЦК"
name="НМЦК" valueType="money" weight="25" />
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/integer_Планируемый_год_размещения" name="Планируемый год
размещения" valueType="text" weight="20" />
    </columns>
    <sorts>
      <sort key="doc_RegCard/rc_Index/text_Статус" direction="ascending"
priorityValues="В работе; Черновик"/>
    </sorts>
  </table>
  <table id="ZNR" flow="Запросы на разъяснение">
    <settings>
      <setting key="canModify" value="true" />
      <setting key="canCreateLinks" value="true" />
      <!-- <setting key="terminalRoute" value="Согласование плана закупок"/>-->
    </settings>
    <columns>
      <column fieldType="attribute"
key="doc_RegCard/rc_Index/text_Регистрационный_номер" name="Рег. номер"
valueType="text" weight="20" isOptional="true" />
    </columns>
  </table>
</root>
```

```

        <column fieldType="attribute"
key="doc_RegCard/rc_Index/date_Дата_регистрации" name="Дата" valueType="date"
weight="25" />
        <column fieldType="attribute"
key="doc_RegCard/rc_Index/longtext_Описание_запроса" name="Описание запроса"
valueType="longtext" weight="55" />
    </columns>
</table>
<table>
    <settings>
        <setting key="canModify" value="true" />
        <setting key="canCreateLinks" value="true" />
        <setting key="terminalRoute" value="Удаление заявки из извещения"/>
    </settings>
    <columns>
        <column fieldType="attribute"
key="doc_RegCard/rc_Index/longtext_Наименование_объекта_закупки" name="Наименование
закупки" valueType="longtext" weight="55" />
        <column fieldType="attribute" key="doc_RegCard/rc_Index/money_НМЦК"
name="НМЦК" valueType="money" weight="25" />
        <column fieldType="attribute"
key="doc_RegCard/rc_Index/integer_Планируемый_год_размещения" name="Планируемый год
размещения" valueType="text" weight="20" />
    </columns>
</table>
</root>

```

Внешний вид таблицы с указанной в примере структурой приведен на рис. 6. Заявки на закупку ППЗ

Пример внешнего вида таблицы

Рисунок 6

Существует возможность настроить работу с таблицами в легком клиенте таким образом, чтобы при выборе какой-либо строки таблицы, на странице документа отображались дополнительные поля или таблица, связанная с текущей (далее — вложенная таблица). Для этого используются теги `<view>` и `</view>`, которые указывают, где начинается и где заканчивается перечисление дополнительных полей (или где находится вложенная таблица). Тег `<field>` описывает строку, при выборе которой должны отображаться дополнительные поля или вложенная таблица.

Примечание 4. При выборе строки дополнительные поля (или вложенная таблица) всегда отображаются ниже текущей таблицы.

Пример части шаблона отображения регистрационной карточки с вложенной таблицей:

```
<view>
  <rows>
    <row>
      <column>
        <field name="Контакты" isLabel="true" />
        <table key="|Таблица_контактов">
          <column fieldType="field" key="|ФИО_адресата" name="ФИО адресата" valueType="text"
/>
          <column fieldType="field" key="|Почтовый_адрес" name="Почтовый адрес"
valueType="text" />
          <column fieldType="field" key="|Телефон" name="Телефон" valueType="text" />
          <column fieldType="field" key="|Факс" name="Факс" valueType="text" />
          <column fieldType="field" key="|Email" name="Email" valueType="text" />
        </table>
      </column>
    </row>
  </rows>
</view>
```

Есть возможность включить перенос текста в заголовке таблицы, для этого необходимо добавить следующий атрибут в описание отображаемой колонки `headerNoEllipsis="true"`.

Пример описания переноса текста заголовка в таблице:

```
<table key="|Document|Типовые_контракты" height="300">
  <column fieldType="field" key="|Номер_типового_контракта" valueType="longtext"
name="Реестровый номер типового контракта" width="150" headerNoEllipsis="true" />
  <column fieldType="field" key="|Тип_типового_контракта" valueType="text" name="Вид
документа" width="200" headerNoEllipsis="true"/>
  <column fieldType="field" key="|Номер_документа_утвердивший_ТК" valueType="text"
name="Номер документа, утвердившего типовой контракт" width="140"
headerNoEllipsis="true"/>
```

```

    <column fieldType="field" key="|Наименование_документа_утвердившего_ТК"
valueType="text" name="Наименование документа, утвердившего типовой контракт"
width="300" headerNoEllipsis="true"/>
    <column fieldType="field" key="|Дата_документа_утвердившего_ТК"
valueType="datetime" name="Дата документа, утвердившего типовой контракт" width="150"
headerNoEllipsis="true"/>
</table>

```

4.14.5. Выбор значений из полей /таблиц и других форм (FormdataDict)

Элемент выбора предназначен для выбора из списка значений, который формируется на основании значений полей других документов и/или таблиц.

Настройки форм элемента выбора осуществляются аналогично элементу из справочника за исключением атрибута **edit-tool**, который в данном случае прописывается следующим образом:

```
edit-tool="formdata:Ключ Настроек"
```

где Ключ Настроек - идентификатор настроек, которые определяют, откуда и каким образом взять данные для списка значений. Также этот ключ используется в качестве заголовка окна выбора значений.

Настройки хранятся в файле \Templates\Document\FormdataDict.xml, который имеет следующий формат:

```

<?xml version="1.0" encoding="utf-8" ?>
<Data>
  <Settings key="Ключ Настроек">
    <Select>
      <Select ... />
    </Select>
    <DictColumns ...>
      <Column ... />
      <Column ... />
    </DictColumns>
  </Settings>
  <Settings key="Другой Ключ Настроек">
    <Select>
      <Select ... />
    </Select>
    <DictColumns ...>
      <Column ... />
      <Column ... />
    </DictColumns>
  </Settings>
</Data>

```

Каждый блок настроек состоит из двух разделов Select и DictColumns

Раздел Select настраивается аналогично одноименному разделу в Hooks.xml.

Раздел **DictColumns** содержит перечень столбцов (**Column**), которые будут выводиться пользователю для выбора и в которых описан способ получения данных.

Атрибуты перечня столбцов (**DictColumns**):

- **sourceTag** - ссылка на документ или набор документов из раздела **Select** (атрибут **tag**);

- **tableName** - имя таблицы, из которой будут использоваться значения полей, если нужны значения полей непосредственно с документа, атрибут опускается. Если указывается подтаблица, то имя должно содержать {NUM}.

Пример: `tableName="|Document|Характеристики_ПКТРУ|{NUM}|Допустимые_значения_характеристики"`;

- **parentTableKeyAttr** - ключевой реквизит родительской таблицы (используется, если в **tableName** указана подчиненная таблица). По этому реквизиту происходит связка строк в дочерней таблице и строк в родительской таблице;

- **rowFilter** - формула фильтрации строк таблицы, если данная формула фильтрации не указана - используются все строки, если указана - только строки, подходящие под условия.

Пример: `rowFilter="{Тип_контракта}={editName=contractType}"`

- **rowFilterCondition** - условие на фильтр. Задаётся в том же формате, что и **rowFilter**. Если условие выполняется, то применяется **rowFilter**, иначе **rowFilter** не применяется.

Атрибуты столбца (**Column**):

- **fieldName** - имя поля документа или столбца таблицы, из которого будут использованы данные для столбца списка значений для выбора. Подстановка "%key%" позволяет получить ключ текущего документа;

- **formula** - формула, результат вычисления которой будет записан в ячейку столбца. Применяется вместо **fieldName**.

Пример: `formula="{Цена}*{editName=totalAmount}"`

- **visibleName** - заголовок столбца, который отображается пользователю при выборе значений, вместе с тем поле должно совпадать с атрибутом **dict-column-name** при настройке шаблона документа, если требуется, чтобы это поле было заполнено значением столбца;

- **isCode** - "true" - признак того, что это столбец с кодом, то есть указание **visibleName** не требуется, а при настройке шаблона документа для такого столбца указывается **dict-column-name="code"** (аналогично настройке выбора из справочника);

aggregate - операция агрегации, которая будет производиться над значением столбца. Пример файла настроек:

```
<?xml version="1.0" encoding="utf-8" ?>
<Data>
  <Settings key="testcase">
    <Select>
      <Select searchString="document[doc_Id = '{editName=kontractnumber}']"
tag="Контракты" />
    </Select>
  </Settings>
</Data>
```

```

</Select>
<DictColumns sourceTag="Контракты"
tableName="|Document|Этапы_исполнения_контракта"
rowFilter="{Этап_закрыт_контракта}!=1">
  <Column fieldName="Дата_окончания_этапа_контракта" isCode="true" />
  <Column fieldName="Сумма_платежей" visibleName="Сумма" aggregate="sum" />
</DictColumns>
</Settings>

```

4.14.6. Группировка дат для контроля хронологии

В шаблонах для полей ввода дат (`valueType="date"` или `valueType="datetime"`) можно включить группировку, внутри которой будут автоматически проставляться ограничения, контролирующие их хронологию.

Для того, чтобы включить группировку полям ввода дат (`valueType="date"` или `valueType="datetime"`) необходимо:

1) в шаблоне для тега `<field valueType="date" ...>` назначить наименование группы атрибутом **date-group-id** (строка);

2) на странице происходит контроль всех полей с одинаковой группой, поэтому наименование должно быть уникально для группы;

3) в шаблоне для тегов `<field valueType="date" ...>` одной группы назначены индексы с помощью атрибута **date-group-index** (целое число больше нуля, пропуски значений допускаются)

Поведение группировки:

- поля ввода дат на форме можно объединять в группы;

- внутри одной страницы для каждой группы назначается уникальное название;

- группы полей с разными названиями друг на друга не влияют.

Внутри одной группы для полей назначаются индексы, обозначающие следование значений полей по хронологии по следующим правилам:

- Поле не влияет на другие поля, если его индекс равен нулю или он не указан, если индексы в группе не указаны, аналогично влияние поля на другие поля не происходит;

- поле с меньшим индексом не может иметь значение больше, чем поле, имеющее больший индекс;

- поля с равным индексом друг на друга не влияют;

- поля с равным индексом влияют на другие поля совместно: минимальное (максимальное) значение полей одного индекса влияют на ограничение предшествующих (последующих) по хронологии полей соответственно;

- пустые значения полей в одной группе при пересчёте ограничений игнорируются, но на них так же накладываются ограничения.

Пример группы, состоящей из двух полей, (со следующим результатом `Дата_от <= Дата_по`):

```
<field fieldType="field" valueType="date" isValue="true" key="Дата_от" name="Дата от"
edit-name="Date_since" date-group-id="dateRangeConditional" date-group-index="1"/>
<field fieldType="field" valueType="date" isValue="true" key="Дата_по" name="Дата до"
edit-name="Date_until" date-group-id="dateRangeConditional" date-group-index="2"/>
```

Пример группы из трех полей с разными индексами, значения дат по хронологии будут идти друг за другом (индексы возможно пропускать) со следующим результатом: (Дата_от <= Дата_ср <= Дата_по)

```
<field fieldType="field" valueType="date" isValue="true" key="Дата_от" name="Дата от"
edit-name="Date_since" date-group-id="dateRangeConditional" date-group-index="1"/>
<field fieldType="field" valueType="date" isValue="true" key="Дата_ср" name="Дата ср"
edit-name="Date_middl" date-group-id="dateRangeConditional" date-group-index="2"/>
<field fieldType="field" valueType="date" isValue="true" key="Дата_по" name="Дата до"
edit-name="Date_until" date-group-id="dateRangeConditional" date-group-index="5"/>
```

Пример группы, состоящей из трех полей, два из которых имеет одинаковый индекс, совместно влияющие на третье поле (его значение значение может быть меньше обоих значений первых полей) имеет следующий результат: Дата_с1 <= Дата_по **И** Дата_с2 <= Дата_по

```
<field fieldType="field" valueType="date" isValue="true" key="Дата_с1" name="Дата с1"
edit-name="Date_since1" date-group-id="dateRangeConditional" date-group-index="1"/>
<field fieldType="field" valueType="date" isValue="true" key="Дата_с2" name="Дата с2"
edit-name="Date_since2" date-group-id="dateRangeConditional" date-group-index="1"/>
<field fieldType="field" valueType="date" isValue="true" key="Дата_по" name="Дата по"
edit-name="Date_until1" date-group-id="dateRangeConditional" date-group-index="2"/>
```

4.14.7. Настройка видимости кнопок верхней панели

Настройка видимости кнопок верхней панели находится в файле Templates/Document/RegisterRules.xml относительно корня легкого клиента. Существуют тэги с условиями и проверками. Тэги могут содержать в себе другие тэги с условиями или проверками, но не оба сразу.

Пример Отображения кнопки согласно условиям:

```
<removeSettings docCondition="(doc_Reg<And>
  <And>
    <WhenUserInGroup name="Инициаторы закупки" />
    <WhenHasNotProcess name="Согласование заявки на закупку" />
  </And>
  <Or>
    <WhenFieldEquals path="|Document|Статус" value="Черновик" />
    <WhenFieldEquals path="|Document|Статус" value="Доработка новой заявки" />
  </Or>
</And>
```

Реализованные условия:

- Or - Любая проверка внутри условия должна быть успешной;
- And - Все проверки внутри условия должны быть успешными.

4.14.8. Настройка блока редактирования

В блоке редактирования описываются поля регистрационной карточки, которые отображаются при открытии окна **Редактировать регистрационную карточку** и при создании нового документа средствами легкого клиента. Теги `<edit>` и `</edit>` указывают, где начинается и где заканчивается блок редактирования. Чтобы при регистрации документа корректно формировался его регистрационный номер, в блоке редактирования вводятся *переменные*.

Пример части шаблона отображения регистрационной карточки с блоком редактирования:

```

<edit>
  <rows>
    <row>
      <column>
        <field name="Регистрационный номер" isLabel="true" />
      </column>
    </row>
    <row>
      <column>
        <field fieldType="field" key="|Document|Префикс_регномаера" name="Префикс
регномаера" valueType="text" isValue="true" edit-name="prefix" edit-autocomplete="Prefix" />
      </column>
      <column>
        <field fieldType="field" key="|Document|Число_регномаера" name="Число регномаера"
valueType="text" isValue="true" edit-name="regnum" edit-readonly="true" />
      </column>
      <column>
        <field fieldType="field" key="|Document|Суффикс_регномаера" name="Суффикс
регномаера" valueType="text" isValue="true" edit-name="suffix" edit-autocomplete="Suffix" />
      </column>
    </row>
    <row>
      <column>
        <field name="Краткое содержание" isLabel="true" />
      </column>
    </row>
    <row>
      <column>
        <field fieldType="field" key="|Document|Краткое_содержание"
name="Краткое содержание" valueType="longtext" isValue="true" edit-
name="description" edit-required="true"/>
      </column>
    </row>
  </rows>
  <variables>
    <variable variable-name="regNumber" variable-format="{prefix}{number}{suffix}">
<field fieldType="field" key="|Document|Префикс_регномаера" name="Префикс регномаера"
valueType="text" variable-name="prefix" variable-format="{value}-" />
      <field fieldType="field" key="|Document|Число_регномаера" name="Число регномаера"
valueType="text" variable-name="number" variable-format="{value}" />
      <field fieldType="field" key="|Document|Суффикс_регномаера" name="Суффикс
регномаера" valueType="text" variable-name="suffix" variable-format="-{value}" />
    </variable>
  </variables>
</edit>
</root>

```

Внешний вид регистрационной карточки для шаблона с указанной в примере структурой блока редактирования приведен на рис.7.

Пример внешнего вида регистрационной карточки

Позиции плана графика, Редактировать Копирование ППГ Показать ППЗ Создать контракт

<p>Основные сведения</p> <p>Обоснование закупки</p> <p>Особенности размещения</p> <p>Требования обеспечения</p> <p>Информация о поставке</p> <p>Количество (объем)</p> <p>Победители</p> <p>Позиции 1</p> <p>Извещения о закупочной процедуре 1</p> <p>Задачи на размещение закупки 1</p> <p>ЭП</p>	<p>Общие сведения</p> <table border="1"> <tr> <td>Регистрационный номер</td> <td>13.06.2018</td> <td>Статус</td> <td>Закупка завершена</td> </tr> <tr> <td>Дата регистрации</td> <td>18233150013953315010010011920241</td> <td>Дата публикации</td> <td>2018</td> </tr> <tr> <td>ИКС</td> <td></td> <td>Планируемый год размещения</td> <td>2018</td> </tr> <tr> <td>Реестровый номер ПГ</td> <td></td> <td>Реестровый номер позиции ПГ</td> <td>1</td> </tr> <tr> <td>Номер закупки в ЕИС</td> <td></td> <td>Номер закупки в ПГ</td> <td>1</td> </tr> <tr> <td>Номер редакции</td> <td>0</td> <td></td> <td></td> </tr> </table> <p>Организационные сведения</p> <table border="1"> <tr> <td>Организация-заказчик</td> <td>МКУК "ДВОРЕЦ КУЛЬТУРЫ И СПОРТА" (ИНН 2907088630)</td> <td>ГРЭС</td> <td>АДМИНИСТРАЦИЯ МУНИЦИПАЛЬНОГО ОБРАЗОВАНИЯ "ВЕЛЬСКОЕ"</td> </tr> <tr> <td>Организатор закупки</td> <td>МКУК "ДВОРЕЦ КУЛЬТУРЫ И СПОРТА" (ИНН 2907088630)</td> <td></td> <td></td> </tr> <tr> <td>Роль организатора закупки</td> <td>Заказчик</td> <td></td> <td></td> </tr> <tr> <td>Централизованная закупка</td> <td><input type="checkbox"/></td> <td>Совместные торги</td> <td><input type="checkbox"/></td> </tr> </table> <p>Сведения о закупке</p> <table border="1"> <tr> <td>Наименование объекта закупки</td> <td>Карандаш</td> <td>Описание объекта закупки</td> <td>Карандаши простые</td> </tr> <tr> <td>ОКПД</td> <td>19.20.21.100</td> <td>ОКПД наименование</td> <td>Бензин автомобильный</td> </tr> <tr> <td>КБР</td> <td>241</td> <td>КБР наименование</td> <td>Научно-исследовательские и опытно-конструкторские работы</td> </tr> <tr> <td>Содержит несколько ОКПД</td> <td><input type="checkbox"/></td> <td>Угруппенная позиция</td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>Используется КТРУ</td> <td><input checked="" type="checkbox"/></td> <td>Наименование товара, работы, услуги по КТРУ</td> <td>Бензин автомобильный (розничная реализация)</td> </tr> <tr> <td>Код товара, работы, услуги по КТРУ</td> <td>19.20.21.100.00001</td> <td>Валюта</td> <td>Российский рубль</td> </tr> <tr> <td>Тип особой закупки</td> <td></td> <td>% аванса</td> <td></td> </tr> <tr> <td>НМЦК</td> <td>100000.00</td> <td>Максимальный размер оплаты</td> <td></td> </tr> <tr> <td>Размер аванса</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Банковское сопровождение контракта</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Содержит несколько позиций ОКПД/КТРУ</td> <td><input type="checkbox"/></td> <td></td> <td></td> </tr> </table> <p>Дополнительные сведения</p> <table border="1"> <tr> <td>Дополнительные сведения</td> <td>Информация о контрактной службе, контрактном управляющем</td> </tr> <tr> <td>Информация о возможности одностороннего отказа от исполнения контракта</td> <td></td> </tr> <tr> <td>Электронная закупка</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Общественное обсуждение</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Лекарственный препарат</td> <td><input type="checkbox"/></td> </tr> <tr> <td></td> <td>Контракт жизненного цикла</td> </tr> </table>	Регистрационный номер	13.06.2018	Статус	Закупка завершена	Дата регистрации	18233150013953315010010011920241	Дата публикации	2018	ИКС		Планируемый год размещения	2018	Реестровый номер ПГ		Реестровый номер позиции ПГ	1	Номер закупки в ЕИС		Номер закупки в ПГ	1	Номер редакции	0			Организация-заказчик	МКУК "ДВОРЕЦ КУЛЬТУРЫ И СПОРТА" (ИНН 2907088630)	ГРЭС	АДМИНИСТРАЦИЯ МУНИЦИПАЛЬНОГО ОБРАЗОВАНИЯ "ВЕЛЬСКОЕ"	Организатор закупки	МКУК "ДВОРЕЦ КУЛЬТУРЫ И СПОРТА" (ИНН 2907088630)			Роль организатора закупки	Заказчик			Централизованная закупка	<input type="checkbox"/>	Совместные торги	<input type="checkbox"/>	Наименование объекта закупки	Карандаш	Описание объекта закупки	Карандаши простые	ОКПД	19.20.21.100	ОКПД наименование	Бензин автомобильный	КБР	241	КБР наименование	Научно-исследовательские и опытно-конструкторские работы	Содержит несколько ОКПД	<input type="checkbox"/>	Угруппенная позиция	<input checked="" type="checkbox"/>	Используется КТРУ	<input checked="" type="checkbox"/>	Наименование товара, работы, услуги по КТРУ	Бензин автомобильный (розничная реализация)	Код товара, работы, услуги по КТРУ	19.20.21.100.00001	Валюта	Российский рубль	Тип особой закупки		% аванса		НМЦК	100000.00	Максимальный размер оплаты		Размер аванса				Банковское сопровождение контракта				Содержит несколько позиций ОКПД/КТРУ	<input type="checkbox"/>			Дополнительные сведения	Информация о контрактной службе, контрактном управляющем	Информация о возможности одностороннего отказа от исполнения контракта		Электронная закупка	<input type="checkbox"/>	Общественное обсуждение	<input type="checkbox"/>	Лекарственный препарат	<input type="checkbox"/>		Контракт жизненного цикла
Регистрационный номер	13.06.2018	Статус	Закупка завершена																																																																																														
Дата регистрации	18233150013953315010010011920241	Дата публикации	2018																																																																																														
ИКС		Планируемый год размещения	2018																																																																																														
Реестровый номер ПГ		Реестровый номер позиции ПГ	1																																																																																														
Номер закупки в ЕИС		Номер закупки в ПГ	1																																																																																														
Номер редакции	0																																																																																																
Организация-заказчик	МКУК "ДВОРЕЦ КУЛЬТУРЫ И СПОРТА" (ИНН 2907088630)	ГРЭС	АДМИНИСТРАЦИЯ МУНИЦИПАЛЬНОГО ОБРАЗОВАНИЯ "ВЕЛЬСКОЕ"																																																																																														
Организатор закупки	МКУК "ДВОРЕЦ КУЛЬТУРЫ И СПОРТА" (ИНН 2907088630)																																																																																																
Роль организатора закупки	Заказчик																																																																																																
Централизованная закупка	<input type="checkbox"/>	Совместные торги	<input type="checkbox"/>																																																																																														
Наименование объекта закупки	Карандаш	Описание объекта закупки	Карандаши простые																																																																																														
ОКПД	19.20.21.100	ОКПД наименование	Бензин автомобильный																																																																																														
КБР	241	КБР наименование	Научно-исследовательские и опытно-конструкторские работы																																																																																														
Содержит несколько ОКПД	<input type="checkbox"/>	Угруппенная позиция	<input checked="" type="checkbox"/>																																																																																														
Используется КТРУ	<input checked="" type="checkbox"/>	Наименование товара, работы, услуги по КТРУ	Бензин автомобильный (розничная реализация)																																																																																														
Код товара, работы, услуги по КТРУ	19.20.21.100.00001	Валюта	Российский рубль																																																																																														
Тип особой закупки		% аванса																																																																																															
НМЦК	100000.00	Максимальный размер оплаты																																																																																															
Размер аванса																																																																																																	
Банковское сопровождение контракта																																																																																																	
Содержит несколько позиций ОКПД/КТРУ	<input type="checkbox"/>																																																																																																
Дополнительные сведения	Информация о контрактной службе, контрактном управляющем																																																																																																
Информация о возможности одностороннего отказа от исполнения контракта																																																																																																	
Электронная закупка	<input type="checkbox"/>																																																																																																
Общественное обсуждение	<input type="checkbox"/>																																																																																																
Лекарственный препарат	<input type="checkbox"/>																																																																																																
	Контракт жизненного цикла																																																																																																

Рисунок 7

Для корректного отображения полей при открытии окна **Редактировать регистрационную карточку** и при создании нового документа, в блоке редактирования настоятельно рекомендуется ограничивать количество столбцов в каждой строке:

- не более 2 столбцов с надписями (в зависимости от ширины столбца);
- не более 4 столбцов без надписей.

Допускается добавлять неограниченное количество строк в структуру блока редактирования.

В блоке редактирования допускается использовать дополнительные атрибуты полей, описанные в таблице 6. Атрибуты, которые могут принимать значения типа «true/false», по умолчанию всегда принимают значение «false».

Таблица 7— Дополнительные атрибуты полей в блоке редактирования

Название атрибута	Значение
isLabel	При значении, равном «true», будет отображаться только название поля
isValue	При значении, равном «true», будет отображаться только значение поля

edit-name	Уникальное название поля, с помощью которого автоматически определяется отредактированное значение, которое следует присвоить полю
edit-readonly	При значении, равном «true», поле будет недоступно для редактирования, при этом значение поля будет доступно для копирования. Атрибут не рекомендуется использовать для полей, которые описывают пополняемые словари или адресную книгу
edit-disabled	При значении, равном «true», поле будет недоступно для редактирования и копирования указанного в поле значения
edit-required	При значении, равном «true», поле будет обязательно для заполнения
edit-hidden	При значении, равном «true», поле будет скрыто для редактирования
edit-isInvisible	При значении, равном «true», поле будет являться невидимым
edit-isMultiply	При значении, равном «true», в поле будет допустим множественный выбор значений
edit-isIgnored	При значении, равном «true», значение поля не будет обрабатываться при регистрации документа, редактировании документа или формировании отчета
edit-IsScriptValue	При значении, равном «true», значение поля будет нередактируемым полем и будет заполняться автоматически средствами скриптов
edit-dictionary	Название непополняемого словаря, из которого будет выбираться значение поля
edit-autocomplete	Название словаря, из которого можно выбрать значение поля, либо указать значение вручную
edit-isReplenishableDictionary	При значении, равном «true», словарь, название которого указано в качестве значения атрибута edit-autocomplete , будет пополняемым
edit-dictionaryGroup	Название группы полей, используемых для работы с иерархическими словарями
edit-dictionaryValue	Название столбца иерархического словаря, которое будет храниться в поле
edit-dictionaryDisplay	Название столбца иерархического словаря, которое будет отображаться в поле

edit-default-value	<p>Значение поля по умолчанию, которое автоматически присваивается при регистрации. Может принимать следующие значения: - «{now}» — текущее время;</p> <ul style="list-style-type: none"> - «{today}» — текущая дата; - «{currentUser}» — текущий пользователь; - любое текстовое значение
Название атрибута	Значение
edit-value	<p>Значение поля, которое автоматически присваивается при регистрации или редактировании документа. Может принимать следующие значения:</p> <ul style="list-style-type: none"> - «{now}» — текущее время; - «{today}» — текущая дата; - «{currentUser}» — текущий пользователь; - любое текстовое значение
edit-addressbook-groups	<p>Список типов элементов адресной книги, значения из которых будут доступны для выбора. Элементы в списке разделяются запятыми. Элементы могут принимать следующие значения:</p> <ul style="list-style-type: none"> - «all» — все элементы адресной книги, кроме корневого подразделения; - «SelectableTypes» — все элементы адресной книги, кроме списков рассылки и служб; - «Person» — сотрудники; - «Department» — подразделения; - «Role» — роли; - «WorkGroup» — рабочие группы; - «SystemRole» — системные роли; - «AccessGroup» — группы доступа; - «Position» — должности; - «ObservablePerson» — контролеры заданий; - «Admin» — администраторы комплекса; - «CurrentUser» — текущий пользователь; - «MailList» — список рассылки; - «Service» — службы.
edit-isCalculated	<p>При значении, равном «true», поле будет вычисляемым, то есть значение этого поля может быть взято из другого поля. В этом случае значение атрибута name текущего поля должно совпадать со значением атрибута name того поля, значение которого будет взято. Вычисляемое поле всегда скрыто</p>

4.14.9. Переменные в блоке редактирования

Чтобы при регистрации документа корректно формировался его регистрационный номер, в блоке редактирования вводятся *переменные*. Теги `<variables>` и `</variables>` указывают, где начинается и где заканчивается перечисление переменных. Переменные делятся на *глобальные* и *локальные*. Каждая локальная переменная вводится для одного поля. Глобальные переменные содержат несколько локальных. Теги `<variable>` и `</variable>` указывают, где начинается и где заканчивается перечисление локальных переменных, входящих в состав глобальной.

Допускается использовать дополнительные атрибуты полей для переменных, описанные в таблице 8.

Таблица 8— Дополнительные атрибуты полей для переменных

Название атрибута	Значение
variable-name	Имя переменной, служащее для обращения к ней
variable-format	Внешний вид значения поля. Может принимать следующие значения: - «{value}» — значение поля; - «{key}» — ключевое значение поля; - «{name}» — название поля; - «{variable-name}» — название переменной; - «{имя_локальной_переменной}» — значение локальной переменной (для глобальных переменных)

4.15. Порядок создания шаблона отображения новой модели содержания или потока документов

Шаблон для новой модели содержания или потока документов создается в следующем порядке:

- 1) откройте файл «_Sample.xml», расположенный в соответствующей папке (см. п. 5.3);
- 2) сохраните файл в той же папке под следующим именем:
 - «<имя_модели_содержания>.xml» — если файл создается для модели содержания регистрационной карточки;
 - «<имя_файла>.xml» — если файл создается для потока. Имя файла должно отличаться от имен существующих моделей содержания регистрационной карточки;
- 3) в созданном файле отредактируйте блоки просмотра и редактирования (см. пп. 5.11.2, 5.11.4);

- 4) если шаблон создан для потока документов, установите соответствие между шаблоном и потоком документов (см. п. 5.13);
- 5) проверьте корректность работы легкого клиента (см. документ «Легкий клиент. Руководство пользователя»).

4.16. Настройка соответствия шаблонов отображения регистрационных карточек документов конкретным потокам

4.16.1. Механизм определения шаблона отображения регистрационной карточки

Программный механизм определения шаблона, который будет использоваться для отображения регистрационной карточки, следующий:

- 1) проверка наличия в папке «\Templates\Document\RegCard\CustomTemplates» файла с названием «<название модели содержания>.xml». Если файл найден, то он будет использоваться в качестве шаблона отображения регистрационной карточки документа;
- 2) если файл «<название модели содержания>.xml» не найден, то выполняется проверка файла «Flows.xml» на наличие в нем сведений о соответствии между потоком и шаблоном отображения;
- 3) если в файле «Flows.xml» отсутствуют сведения о соответствии между потоком и шаблоном отображения, то для отображения регистрационной карточки документа будет использоваться шаблон «RegCard.xml».

4.17. Настройка шаблонов форм поиска

Шаблоны отображения форм поиска документов или задач на вкладке **Поиск** во вкладках второго уровня **По документам** или **По задачам** имеют однотипную структуру. Пример шаблона отображения формы поиска документов на вкладке второго уровня **По документам**:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
<edit>
  <rows>
    <row>
```

```
<column>
  <field name="Регистрационный номер" isLabel="true" />
</column>
<column>
  <field fieldType="field" key="|Document|Префикс_регномера" name="Префикс
регномера" valueType="text" isValue="true" edit-name="prefix" edit-
autocomplete="Prefix"/>
```

1

```

        </column>
        <column>
            <field fieldType="field" key="|Document|Число_регномаера" name="Число регномаера"
valueType="text" isValue="true" edit-name="regnum" />
        </column>
        <column>
            <field fieldType="field" key="|Document|Суффикс_регномаера" name="Суффикс
регномаера" valueType="text" isValue="true" edit-name="Suffix" edit-
autocomplete="Suffix"/>
        </column>
    </row>
    <row>
        <column>
            <field name="Дата регистрации от" isLabel="true" weight="3" />
        </column>
        <column>
            <field fieldType="field" key="|Document|Дата_регистрации_от" name="Дата
регистрации от" valueType="date" isValue="true" edit-name="registerDateFrom"
weight="300"/>
        </column>
        <column>
            <field name="до" isLabel="true" />
        </column>
        <column>
            <field fieldType="field" key="|Document|Дата_регистрации_до"
name="до" valueType="date" isValue="true" edit-name="registerDateTo" />
        </column>
    </row>
    <row>
        <column>
            <field name="Поток" isLabel="true"/>
            <field fieldType="field" key="|Поток" name="Поток" edit-name="flowKey"
valueType="text" isValue="true" edit-dictionary="#Потоки"
editdictionaryDisplay="Название" edit-dictionaryGroup="flow" edit-isMultiple ="true" />
        </column>
    </row>
    <row>
        <column>
            <field name="Автор" isLabel="true" />
        </column>
    </row>
    <row>
        <column>
            <field fieldType="field" key="|Document|Автор" name="Автор"
valueType="AddressBook" isValue="true" edit-name="creator" edit-isMultiple ="true"
list="content-addressbook-suggestions"/>
        </column>
    </row>
    <row>
        <column>

```

```

    <column>
      <field name="Регистратор" isLabel="true" />
    </column>
  </row>
  <row>
    <column>
      <field fieldType="field" key="|Document|Регистратор" name="Регистратор"
valueType="AddressBook" isValue="true" content-addressbook-suggestions="control"
editname="registrator" edit-isMultiple ="true" />
    </column>
  </row>
  <row>
    <column>
      <field name="Краткое содержание" isLabel="true" />
    </column>
  </row>
  <row>
    <column>
      <field fieldType="field" key="|Document|Краткое_содержание" name="Краткое
содержание" valueType="longtext" isValue="true" edit-name="description"
list="contentaddressbook-suggestions" />
    </column>
  </row>
  <row>
    <column>
      <field name="Поиск по контексту" isLabel="true" />
    </column>
  </row>
  <row>
    <column>
      <field fieldType="field" key="|Document|Поиск_по_контексту" name="Поиск по
контексту" valueType="longtext" isValue="true" edit-name="context" />
    </column>
  </row>
</rows>
</edit>
<search>
  <field fieldValue="(doc_RegCard/rc_Index/text_Префикс_регномера = @prefix and
doc_RegCard/rc_Index/integer_Число_регномера = @regnum and
doc_RegCard/rc_Index/text_Суффикс_регномера = @Suffix) and ((doc_RegCard/rc_FlowKey =
@flowKey) and (doc_RegCard/rc_Index/date_Дата_регистрации &gt;= @registerDateFrom) and
(doc_RegCard/rc_Index/date_Дата_регистрации &lt;= @registerDateTo)) and ((doc_Creator =
@creator) and (doc_Registrator = @registrator)) and (xn-
contains(doc_RegCard/rc_Index/longtext_Краткое_содержание ,@description)) and
(doc_Parts/doc_Part/part_Versions/part_Version/ver_Data =
attachment[xncontains(bptc_Data, @context)]/bptc_Id)" />
</search>
</root>

```

Внешний вид формы поиска документов для шаблона с указанной в примере структурой приведен на рис. 8.

Внешний вид формы поиска документов на вкладке Поиск во вкладке второго уровня По документам

Рисунок 8

В шаблонах отображения форм поиска именем тега поля (см. п. 4.2) является **row**. Каждое поле описывает один столбец. Теги `<rows>` и `</rows>` указывают, где начинается и где заканчивается перечисление полей.

В шаблонах отображения списков допускается использование следующих дополнительных атрибутов:

- **weight** — отношение ширины столбца к ширине окна интернет-обозревателя в процентах;
- **isOptional** — столбец, для которого значение данного атрибута равно «true», не будет отображаться при работе с легким клиентом на мобильном устройстве.

При необходимости возможно добавить в структуру шаблона любое количество полей или удалить поля из структуры шаблона.

Шаблон в целом повторяет используемые в системе шаблоны, но нужно заметить, что в шаблоне присутствует раздел `<search>`, в поле `<field>` которого размещается шаблон поисковой строки. Шаблон поисковой строки используется для задания структуры поискового запроса. Структура этого шаблона представляется в виде дерева, элементами этого дерева являются операции сравнения, выглядящие как "элемент_запроса [= значение_запроса | функция(значение_запроса)] например:

```
doc_RegCard/rc_Index/text_Префикс_регномера = @prefix
```

Вместо `@prefix` будет подставлено значение, полученное из поисковой формы. Элемент `@prefix` представляет собой значение-идентификатор и должен начинаться со знака '@' если в шаблоне поискового запроса использовать строковое значение 'prefix'.

- `doc_RegCard/rc_Index/text_Префикс_регномера = 'prefix'` , то оно будет подставлено без изменений, в случае если поле в поисковой форме является полем множественного выбора, в этом случае в поисковый шаблон остается таким же как и при единственном выборе, но в самом же поисковом запросе значения будут объединены через подходящую функцию, Общий вид шаблона поисковой формы по задачам такой же как и по документам, за той лишь разницей, что использование шаблона для поиска по задачам является опциональным, и поля добавленные в шаблон по задачам будут добавляться к текущим фиксированным полям.

Есть возможность задавать дополнительный набор полей для поиска (фильтрации) и для поисковых форм. Набор будет по умолчанию скрыт и не будет использоваться при поиске. При задании набора дополнительных полей рядом с кнопкой **Найти** появится кнопка **Расширенный поиск**, который отвечает за скрытие и показ дополнительного набора полей.

Пример дополнительного добавления поля в шаблон для поиска

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <edit>
  </edit>
  <extraSearchForm>
  </extraSearchForm>
  <search>
    <field fieldValue="поисковый запрос" />
  </search>
</root>
```

На одном уровне с элементом `edit`, в котором описываются поля для поиска, добавляется элемент `extraSearchForm`, структура содержимого, которого совпадает со структурой, которая используется в `edit`.

Примечание. При указании дополнительного набора полей их также необходимо учитывать в поисковой строке, если поля не будут учтены, то значения этих полей не будут учитываться.

4.18. Формат элемента иерархического словаря

Словари используются для стандартизации заполнения predetermined значений полей в форме РК документа. Чтобы избежать опечаток при заполнении полей, которые могут иметь только фиксированные значения, предпочтительно не вводить их с клавиатуры, а выбирать из списка. Список допустимых значений для конкретного поля называется словарем. Привязка словаря к полям выполняется в модуле «Дизайнер форм». Каждому

словарю соответствует определенная *схема данных*, которая отражает структуру словаря, а также список допустимых значений для конкретного поля. *Элементы схемы данных* словаря распределены по *уровням*. На каждом уровне могут находиться не только элементы, но и подуровни. Существует возможность создавать словари, схема данных которых содержит как один уровень, так и неограниченное количество уровней. Готовая схема данных заполняется *записями*, состоящими из информационных элементов. Информационные элементы записи разделяются по параметрам — *атрибутам*. Среди атрибутов уровня выделяют *ключевые атрибуты*. Информационные элементы, соответствующие ключевому атрибуту, называются *ключевыми элементами*. Ключевые элементы выполняют служебную функцию идентификации записи в словаре и к ним привязываются *подчиненные словари*, содержащие записи нижних уровней. С помощью многоуровневого словаря можно организовывать на экранной форме связанные поля со списками. При выборе какого-либо значения в списке такого поля изменяется набор значений в списках других полей. Списки таких полей заполняются записями из подчиненных словарей, подчиненных выбранному информационному элементу.

Словарь является списком возможных значений для какого-либо элемента модели содержания или формы. Возможно изменение и удаление существующих словарей, а также создание пользовательских словарей.

Для добавления словаря (приведен пример для карточки Позичий). Необходимо добавить следующие поля в структуру XML. Главное поле словаря, которое служит для идентификации, что тег <columns> описывает словарные поля. Все поля словаря должны находиться в одном columns.

```
<field fieldType="field" key="|Document|Основание_закупки_у_ЕП_пункт" name="Основание
закупки у ЕП пункт" valueType="text" isValue="true" edit-name="registerOsnZakP" dict-
column-name="code" edit-tool = "Основание закупки у ЕП" />
```

Атрибуты, которые содержат Поля, представлены в таблице 9.

В том же блоке columns могут находиться другие поля, которые заполняются по результату выбора словарной строки.

```
<field fieldType="field"
key="|Document|Организация_заказчик|Организация_заказчик_наименование"
name="Организация-заказчик наименование" valueType="text" isValue="true" dict-column-
name="Наименование" edit-name="registerOrgZaName"
edit-required="true" edit-default-value="ПАО &quot;Россети&quot;" />
```

Атрибут dict-column-name должен быть обязательно заполнен. По нему определяется, какой словарной колонкой заполняется поле.

Таблица 9 — Атрибуты, применяемые в Полях

Атрибут	Тип значения	Обязательность	Описание
edit-tool	string	Указывать только для главного поля	Тип и имя инструмента. Для получения данных из словаря edit tool="dictionary:Имя Словаря"
dict-column-name	string	Обязательно для всех полей	
display-field-part	bool: true, false	Необязательно	
dict-edit-name	string	Обязательно для всех полей	Значение - edit-name поля с атрибутом edit-tool. Описывает к какому словарю относится данное поле в случае если в теге <column /> описано несколько словарей
one-level-parent	string	Необязательно	Значение - edit-name поля родительского словаря. К примеру "Организация-заказчик" родитель для "Способа закупки", значит нужно указать edit-name поля " Document Организация_заказчик Организация_заказчик"
is-select-one-level	int	Необязательно	Уровень на котором расположен текущий словарь в словаре родителе. К примеру "Организация-заказчик" и "Способ закупки" расположены в словаре "Организации заказчики", Но Организация-заказчик расположен на 1ом уровне а "Способ закупки" на 2ом уровне
dictionary-selection-start-level	int/string	Необязательно	Минимальный уровень с которого пользователю разрешено выбирать значение. Имеет место в иерархических словарях. При указании значения "last" - разрешено выбирать только последний элемент (у которого нет дочерних элементов), независимо от его уровня вложенности
dict-subdictionary	string	Необязательно	Название подчиненного словаря. Имеет место в иерархических словарях. Если не указан - берется первый подчиненный словарь
dict-column-width-percentage	int	Необязательно	Ширина столбца в процентах от общей ширины. Если не указан - высчитывается средняя.
show-dict-column	bool	Необязательно	Показывает, отображать столбец или нет. Если не указан, столбец отображается.

Блоки, в которых находятся эти параметры, выглядят следующим образом:

<row>

```

<column>

<field fieldType="field" key="|Document|Организация_поставщик|Организация_поставщик"
  name="Организация поставщик" valueType="text" edit-required="true" isValue="true"
edit-name="orgpost2" hierar-field- name="|Document|Организация_поставщик"
related-field- name="|Document|Организация_поставщик|
Организация_поставщик_наименование"
edit-isScriptValue="true" dictionary-selection-start-level="1" edit-
tool="dictionary:Контрагенты"
  dict-column-name="code"/>
  <field fieldType="field"
key="|Document|Организация_поставщик|Организация_поставщик_OГРН" name="OГРН"
  valueType="text" isValue="true" edit-name="orgpostOGRN" edit-
isScriptValue="true"
  dict-column-name="OГРН" edit-hidden="true" />
</column>
</row>

```

Правила которые учитываются в блоках:

- главное поле словаря, которое служит для идентификации, что тег `<columns>` описывает словарные поля;
 - главное поле должно содержать атрибут `edittool=dictionary: Имя Словаря`;
 - главное поле может содержать дополнительные атрибуты:
 - `is-select-one-level` - уровень словаря, с которого выбираются данные);
 - `one-level-parent` (используется для связи между словарями);
 - `dictionary-selection-start-level` (номер уровня, с которого начинается выборка из словаря);
 - `guid-value` (для связи между словарями).
 - по главному полю генерируются кнопка отображения словаря и поле `display-field` в котором отображаются выбранные значения;
 - главное поле всегда скрыто (**атрибут `edit-hidden` не нужен**) и может быть отображено с помощью `display-field-part=true`;
 - каждое поле словаря должно содержать атрибут `dict-column-name` который указывает соответствие поля колонке;
 - каждое поле может содержать дополнительные атрибуты:
 - атрибут `display-field-part= true`, говорит о том будет ли значение этого поля отображаться в `display-field`. Этот атрибут работает только если поле скрыто `edit-hidden=true`.
 - все поля словаря должны находиться в одном блоке `column`.
- Поля в шаблоне могут быть расположены в произвольном месте шаблона, атрибут `dict-edit-name` является связующим для определения принадлежности заполняемого поля к словарю. Например:

```

<row>
  <column>

```

```

        <field name="Способ закупки" isLabel="true" />
        <field fieldType="field" key="|Document|Способ_закупки_код"
name="Способ закупки" valueType="text" isValue="true" edit-
name="registerSpZakup" edit-required="true" dict-column-name="code" edit-
tool = "dictionary:Организации заказчики" is-select-one-level="2" edit-
isScriptValue="true" one-level-parent="registerOrgZa" dict-edit-
name="registerSpZakup" />
        <field fieldType="field" key="|Document|Способ_закупки" name="Способ
закупки наименование" valueType="text" isValue="true" edit-
name="registerSpZakupName" edit-required="true" edit-isScriptValue="true"
dict-column-name="Наименование" edit-hidden="true" display-field-part="true"
dict-edit-name="registerSpZakup"/>
        <field name="Способ закупки по положению" isLabel="true" />
        <field fieldType="field" key="|Document|Способ_закупки_положение"
name="Способ закупки положение" valueType="text" isValue="true" edit-
name="registerSpZakupPosition" edit-isScriptValue="true" dict-column-
name="Способ по положению" dict-edit-name="registerSpZakup"/>
        <field fieldType="field" key="|Document|Вид_закупки" name="Вид
закупки" valueType="text" isValue="true" edit-name="registerVidZakupki" edit-
hidden="true" edit-isScriptValue="true" dict-column-name="Вид закупки" dict-
edit-name="registerSpZakup"/>
        <field fieldType="field" key="|Document|Количество_этапов"
name="Количество этапов" valueType="text" isValue="true" edit-
name="registerCountEtap" edit-hidden="true" edit-isScriptValue="true" dict-
column-name="Количество этапов" dict-edit-name="registerSpZakup"/>
    </column>
</row>
<row>
    <column>
        <field name="Способ закупки" isLabel="true" />
        <field fieldType="field" key="|Document|Способ_закупки_код"
name="Способ закупки" valueType="text" isValue="true" edit-
name="registerSpZakup" edit-required="true" dict-column-name="code" edit-
tool = "dictionary:Организации заказчики" is-select-one-level="2" edit-
isScriptValue="true" one-level-parent="registerOrgZa" dict-edit-
name="registerSpZakup" />
    </column>
</row>
<row>
    <column>
        <field fieldType="field" key="|Document|Способ_закупки" name="Способ
закупки наименование" valueType="text" isValue="true" edit-
name="registerSpZakupName" edit-required="true" edit-isScriptValue="true"
dict-column-name="Наименование" edit-hidden="true" display-field-part="true"
dict-edit-name="registerSpZakup"/>
    </column>
    <column>
        <field name="Способ закупки по положению" isLabel="true" />
        <field fieldType="field" key="|Document|Способ_закупки_положение"
name="Способ закупки положение" valueType="text" isValue="true" edit-
name="registerSpZakupPosition" edit-isScriptValue="true" dict-column-
name="Способ по положению" dict-edit-name="registerSpZakup"/>
    </column>
</row>
<row>
    <column>

```

```

    <field fieldType="field" key="|Document|Вид_закупки" name="Вид
закупки" valueType="text" isValue="true" edit-name="registerVidZakupki" edit-
hidden="true" edit-isScriptValue="true" dict-column-name="Вид закупки" dict-
edit-name="registerSpZakup"/>
    <field fieldType="field" key="|Document|Количество_этапов"
name="Количество этапов" valueType="text" isValue="true" edit-
name="registerCountEtap" edit-hidden="true" edit-isScriptValue="true" dict-
column-name="Количество этапов" dict-edit-name="registerSpZakup"/>
  </column>
</row>

```

4.19. Формат шаблона присоединенных файлов

Каждый элемент описывает одну таблицу для соответствующего потока документов. В атрибуте `category` содержится название фильтруемой категории - в таблице будут отображаться присоединенные файлы только с указанной категорией. Если атрибут `category` не указан, то будут выведены все присоединенные файлы. Каждый элемент `column` в `view/columns` описывает 1 столбец в режиме просмотра. Описывается совершенно так же, как и в результатах поиска. Каждый элемент `column` в `edit/columns` описывает 1 строку при редактировании свойств присоединенного файла. Описывается совершенно так же, как и элемент `field` при редактировании регистрационной карточки за одним исключением: в атрибутах не должно быть `isLabel=true`, а также каждый элемент представляется в виде 2 строк - заголовка и соответствующего поля под ним. Для `fieldType=property` доступны следующие свойства:

- `AttachmentTitle` - наименование файла с иконкой
- `AttachmentLastModifier` - автор последнего изменения
- `AttachmentLastModify` - дата последнего изменения
- `AttachmentCategory` - категория
- `AttachmentCreatedTime` - дата создания
- `AttachmentVersion` - номер версии
- `AttachmentFileName` - название файла

Редактировать можно только `AttachmentTitle`, `AttachmentFileName` и `AttachmentCategory`. Для показа и редактирования дополнительных добавленных свойств необходимо использовать `fieldType="attribute"`

```

<?xml version="1.0" encoding="utf-8" ?>
<root>
  <table category="Документация">
    <view>
      <columns>
        <column fieldType="property" key="AttachmentTitle" name="Наименование
файла" valueType="text" weight="55" />

```

```

    <column fieldType="property" key="AttachmentLastModifier"
name="Автор изменения" valueType="addressbook" weight="15" />
    <column fieldType="property" key="AttachmentVersion" name="Версия"
valueType="text" weight="15" />
    <column fieldType="property" key="AttachmentLastModify" name="Дата
изменения" valueType="datetime" weight="15" />
  </columns>
</view>
<edit>
  <columns>
    <column fieldType="property" key="AttachmentTitle" edit-
name="attachmentTitle" name="Описание файла" isValue="true"
valueType="text"/>
    <column fieldType="property" key="AttachmentFileName" edit-
name="attachmentFileName" name="Название файла" isValue="true"
valueType="text" edit-required="true"/>
  </columns>
</edit>
</table>
<table category="Протоколы">
  <view>
    <columns>
      <column fieldType="property" key="AttachmentTitle" name="Наименование
файла" valueType="text" weight="55" />
      <column fieldType="property" key="AttachmentLastModifier"
name="Автор изменения" valueType="addressbook" weight="15" />
      <column fieldType="property" key="AttachmentVersion" name="Версия"
valueType="text" weight="15" />
      <column fieldType="property" key="AttachmentLastModify" name="Дата
изменения" valueType="datetime" weight="15" />
    </columns>
  </view>
  <edit>
    <columns>
      <column fieldType="property" key="AttachmentTitle" edit-
name="attachmentTitle" name="Описание файла" isValue="true"
valueType="text"/>
      <column fieldType="property" key="AttachmentFileName" edit-
name="attachmentFileName" name="Название файла" isValue="true"
valueType="text" edit-required="true"/>
    </columns>
  </edit>
</table>
</root>

```

5. ТИПЫ ПОЛЕЙ

5.1.1. Адресная книга

При помощи адресной книги в большинстве случаев может быть выбран не только пользователь или несколько пользователей, но также группа доступа или роль. Состав групп и ролей определяется настройкой в модуле «Администратор». Переход к спискам групп доступа, ролей и списков рассылки осуществляется на соответствующих вкладках. При выборе имени сотрудника этот сотрудник становится адресатом. При выборе названия подразделения адресатами становятся все сотрудники этого подразделения. При выборе роли адресатом становится сотрудник, исполняющий эту роль в данный момент. Имя сотрудника указано в скобках после названия роли. При выборе должности адресатом становится сотрудник, занимающий эту должность в данный момент. Имя сотрудника указано в списке ниже названия должности. При выборе группы доступа адресатами становятся все пользователи, входящие в группу.

Используется два варианта адресной книги. Один вариант — полный — предназначен для выбора одновременно нескольких пользователей, например, при назначении пользователей, для которых данный пользователь будет являться заместителем. Другой вариант — упрощенный — используется для выбора одного пользователя, например, при назначении контролирующего по маршруту. В упрощенном варианте отсутствует список выбранных пользователей (правая часть окна). Полная адресная книга имеет вид, представленный на 9.

Выбор значений из адресной книги

Системное имя	Полное имя	Электронная почта	Администрирование комплекса	Администрирование документооборота	Контроль исполнения заданий	Расширенный контроль
DTLN-E12-TEST1\Администратор	Администратор1 А.А.		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
DTLN-E12-TEST2\admin1	Администратор2 А.А.		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
DTLN-E1-CLIENT1\admin	Администратор3 А.А.		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
DTLN-E1-CLIENT6\admin	Администратор4 А.А.		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Anisimov	Анисимов И.Б.		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
arbuзова	Арбузова А.Н.	arbu@gmail.ru	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
batagova	Батагова И.А.	batan@gmail.ru	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bezak	Безак В.В.		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Vishin	Вишин Н.О.		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
vodnik	Водник В.В.	vodnik@gmail.ru	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Gashin	Гашин А.Н.		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Gorbunov	Горбунов В.И.		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
grekova29	Грекова Н.А.	grekova@chinaland.ru	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Grishin	Гришин Н.Н.		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Gurevich	Гуревич Е.В.		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Dobosh	Добощ Н.И.		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Общие | Подразделение | Наборы прав доступа

Личные данные

Фамилия: Администратор1
Имя: Администратор1
Отчество: Администратор1
Системное имя: DTLN-E12-TEST1\Администратор
Позиция:
Комната:
Телефоны:
Факс:
Электронная почта:
Примечание:
Пароль: *****
Отпечаток:

Особые полномочия

- Администрирование комплекса
- Администрирование документооборота
- Делегирователь
- Контроль исполнения заданий
- Расширенный контроль
- Создание отчетов
- Создание регистрационных форм

Является уполномоченным для:

Является заместителем для:

Конфигурация рабочего места

- Локальная конфигурация (по умолчанию)
- Расширенная конфигурация

Сотрудник: _____ Дата начала: _____

Именованная лицензия

- Заблокирован
- Деактивирован

Рисунок 9

Таблица 10 — Значения атрибутов адресной книги

Атрибут	Тип значения	Обязательность	Описание	Пример
address-book-tool	string	обязательно	добавление этого атрибута заменяет стандартный линейный контрол адресной книги на модальный диалог	address-book-tool = "Address", где Address идентифицирует адресную книгу в шаблоне, может быть произвольной строкой, но должны отличаться для разных полей
edit-addressbook-groups	string	обязательно	в этом атрибуте при добавленном атрибуте address-book-tool задается перечисление типов адресных книг, которые будут присутствовать в контроле	edit-addressbook-groups = "person, Service, WorkGroup, AccessGroup, Role"
address-book-tool-selectable	string	необязательно	в этом атрибуте при добавленном атрибуте address-book-tool задается тип адресной книги, для которой будет осуществляться выборочное отображение элементов по уровням	address-book-tool-selectable = "Department, Person 108,15, 16"
addressbook-main-field	string	необязательно	Атрибут указывается для связанных полей адресной книги. значение атрибута - edit-name поля, к которому привязан контрол адресной книги. Нужен для определения списка связанных полей при выборе элемента	addressbook-main-field="IN"

Атрибут	Тип значения	Обязательность	Описание	Пример
addressbook-property	string	необязательно	название атрибута элемента адресной книги. Указывается совместно с addressbook-main-field. В поле с таким атрибутом запишется значение указанного атрибута элемента адресной книги, который выбран в поле, edit-name которого совпадает со значением, указанным в addressbook-main-field	addressbook-property="Chief"
children-selection	bool	необязательно	Если разрешен множественный выбор, при выборе элемента выбираются также все его дочерние элементы	children-selection="true"или children-selection="false"

В дальнейшем по тексту документа выбор имени пользователя в адресной книге подразумевает возможность выбора подразделения, группы доступа или роли. Вкладка **Последние выбранные** содержит список десяти адресатов, выбранных в адресной книге последними за время текущего сеанса с модулем «Администратор». Если за время текущего сеанса с модулем «Администратор» выбор адресата в окне адресной книги осуществляется впервые, то вкладка **Последние выбранные** не отображается. Вкладка **Сотрудники** содержит список всех имен пользователей. Для того чтобы назначить пользователя адресатом: 1) в левой части окна выберите пользователя, которого следует назначить адресатом. Для быстрого поиска имени укажите первые буквы имени в строке поиска, расположенной в верхней части окна. Для выбора пользователя среди найденных в списке используйте кнопки ▼ и ▲.

Помимо адресной книги, описанной выше, в некоторых случаях также используется упрощенный ее вариант, допускающий выбор только одного пользователя, роли и т. д. (рис.10). Для того чтобы выбрать пользователя в окне такой адресной книги, выберите из списка имя пользователя, которого следует

назначить адресатом, и нажмите на кнопку **Выбрать**. В остальном упрощенная адресная книга аналогична полной.

Упрощенная адресная книга

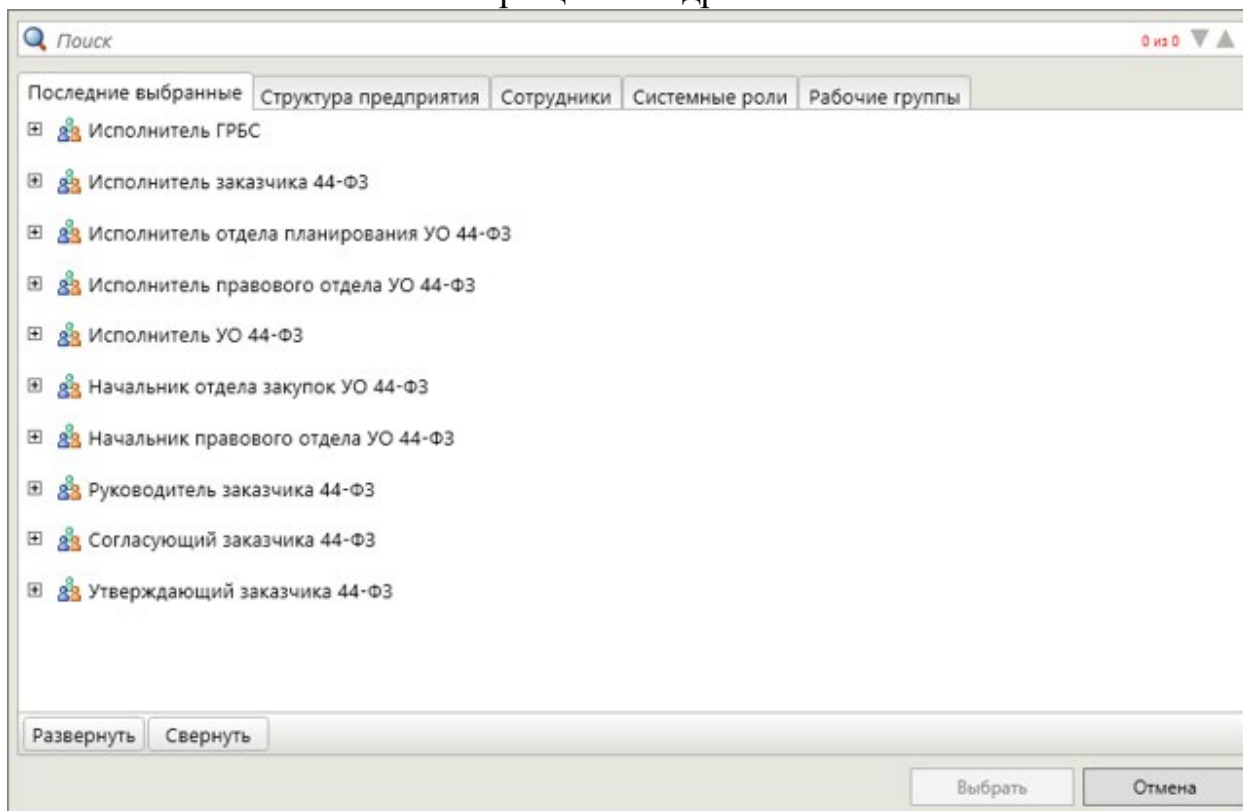


Рисунок 10

Пример части шаблона отображения регистрационной карточки адресной книги:

```
<root>
<edit>
  <rows>
    <row>
      <column>
        <field name="Исполнитель" isLabel="true" />
        <field fieldType="field" name="Исполнитель"
          valueType="AddressBook"
          isValue="true"
          edit-name="performer"
          address-book-tool = "performer"
          edit-addressbook-groups = "person, WorkGroup, Role" edit-isMultiple = "true"
        />
      </column>
      <column>
```

address-book-tool string обязательное добавление этого атрибута заменяет стандартный линейный контроль адресной книги на модальный диалог address-book-tool = "Address", где Address идентифицирует адресную

книгу в шаблоне, может быть произвольной строкой, но должны отличаться для разных полей.

Выбор значений из адресной книги.

Пример описания шаблона:

```
<field fieldType="field" key="|Document|Экспертная_комиссия|Эксперты"
name="Экспертная комиссия" valueType="addressbook" isValue="true" edit-
name="addressbook" address-book-tool = "Address" edit-addressbook-groups = "person,
Service, WorkGroup, AccessGroup, Role" address-book-tool-selectable = "Department,
108,15, 16"/>
```

Данный шаблон создаст контроль, в котором будут отображаться адресные книги person, Service, WorkGroup, AccessGroup, Role, а для книги Department будут доступны элементы для выбора 108,15, 16 и их дочерние.

Форма атрибута **address-book-tool-selectable**:

- первое значение - тип которого происходит выбор значений адресной книги, например Department;

- второе значение - тип значений, выбираемых из адресной книги, например Person;

- последующие перечисления – идентификаторы Первого значения, из которых происходит выбор Второго значения.

Например, если для поля указать атрибуты addressbook-main-field="IN" и addressbook-property="Chief", то при выборе значения в поле с edit-name="IN", к которому привязан контрол показа словаря, в текущее поле запишется значение атрибута "Chief" у выбранного в "IN" элемента адресной книги.

Настройка полей адресной книги в карточках

Допустимы следующие подстановки в атрибуте address-book-tool-selectable поля:

- {currentUserParents} - возвращает родительские элементы текущего пользователя указанного типа (подразделения, группы доступа, списки рассылки);

- {currentOrganization} - возвращает дочерние элементы для текущей организации пользователя (не включая текущую организацию). Тег {currentOrganization} оставлен для совместимости. Лучше использовать {currentOrganizationChilds};

- {currentOrganizationChilds} - возвращает дочерние элементы для текущей организации пользователя (не включая текущую организацию). Т.к. название подстановки {currentOrganization} не соответствует сути, поэтому сделал дубль с более осмысленным названием {currentOrganizationChilds}. Тег {currentOrganization} оставлен для совместимости;

- {currentOrganizationWithChilds} - возвращает текущую организацию и дочерние элементы для текущей организации.

Конвертеры адресной книги. В запросах к полям адресной книги типа «Пользователь» и «Группа» допустимо использовать следующие специальные выражения:

- eq-or-sub - получить элемент равный данному или являющийся его подчиненными элементами.

- Пример: (doc_RegCard/rc_Index/integer_Организация_заказчик xn-in-or-sub @registerOrgZa) ;

- xn-in-or-sub - получить элементы равные данным или являющиеся их подчиненными элементами;

- sub - получить элементы являющиеся его подчиненными элементами.

Пример: (doc_RegCard/rc_Index/integer_Организация_заказчик sub @registerOrgZa).

5.1.2. Текстовое поле

Текстовые поля не ограничивают ввод данных, для таких полей есть необходимость задавать параметры ввода, например, вводить только буквы, цифры или текст в определённом формате

Большинство полей форм создаётся с помощью одиночного тега <input>. У этого тега два обязательных атрибута:

- type задаёт тип поля;

- name задаёт имя поля.

Тип поля влияет на то, как оно будет отображаться. Самый распространённый тип — это text, который обозначает *текстовое поле*.

Пример:

```
<form action="https://echo.htmlacademy.ru" method="get">
  <input type="text" name="search">
</form>
```

Имя поля нужно, чтобы правильно обработать данные на сервере. Как правило имя поля должно быть уникальным, в пределах формы, возможны исключения. Для задания имени поля используют латинские буквы и цифры. Расположение полей описывается по столбцам. Для обозначения отдельных столбцов в каждой строке используется пара тегов <column> и </column>.

5.1.3. Числовое поле инфа

Для ввода чисел предназначено специальное поле, которое допускает ограничения по нижней и верхней границе, а также устанавливает шаг приращения. Само поле для ввода чисел похоже на обычное текстовое поле, но со стрелками, которые позволяют увеличивать и уменьшать значение.

Для ограничения введённого числа предназначены атрибуты **min** и **max**, они могут принимать отрицательное и положительное значение.

Для ввода чисел используется элемент `type=number`. Он создает числовое поле, которое возможно настроить с помощью следующих атрибутов:

- **min**: минимально допустимое значение;
- **max**: максимально допустимое значение;
- **readonly**: доступно только для чтения;
- **required**: указывает, что данное поле обязательно должно иметь значение;
- **step**: значение, на которое будет увеличиваться число в поле;
- **value**: значение по умолчанию.

6. ХУКИ

6.1.1. Работа с транзакциями

Хуки могут выполняться в двух режимах:

- 1) в отдельных транзакциях с сохранением документа;
- 2) в одной транзакции с сохранением документа.

В отдельных транзакциях с начала 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.

С одной транзакцией, 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.

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 к элементу, описывающему событие

Пример настроек:

```
<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 секции настроек на конкретное действие.

Примечание: Для действий, в которых сохранение документа не происходит (OnRouteStarted), атрибут transaction значения не имеет.

Особенности работы с текущим документом. При выполнении хуков в отдельных транзакциях, текущий документ уже сохранен в базе данных, поэтому для доступа к его полям используется 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 тэг "current" может использоваться как для доступа к документу в действиях, так и для использования в качестве подзапроса в качестве подзапроса в других селектах.

При выполнении хуков в одной транзакции с сохранением документа, в базе данных на этот момент либо еще нет документа, либо храниться еще не измененная версия. Данным способом можно воспользоваться в случае отдельных транзакций.

Доступ к текущему документу в select-ах. 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 нет – 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 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>
```

В случае возникновения необходимости выбора нескольких документов в одном запросе, включая текущий. Необходимо выбрать документы кроме текущего, а затем объединить их с текущим с помощью следующей конструкции:

Пример Объединения документа с текущим

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

```
</Select>
```

Доступ к текущему документу в действиях. Текущий документ неявно добавляется к списку документов, получаемых с помощью select-ов, с тэгом "{current}". Таким образом его можно использовать в действиях.

Примеры использования текущего документа в действиях:

```
<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 тэг "{oldCurrent}" который позволяет получать доступ к данным текущего документа, которые были до его изменения.

6.2. Конфигурирование файлов

Хуки конфигурируются для каждого потока отдельно. Для этого, аналогично с шаблонами, создаются xml-файлы в папке по пути:

\\Templates\Document\Hooks\CustomTemplates\<имя_рег.формы>.xml.

Имя файла хуков для потока совпадает с именем регистрационной формы.

Структура файла хуков представлена в таблице 11.

Таблица 11 - Структура файла хуков

Название хука	Описание
<Hooks>	Корневой элемент всех хуков потока
<OnDocumentCreated/>	событие создания документа
<NamedConstants>	формирование именованного счетчика
<CommonConditions>	условия, необходимые для того, чтобы счётчики отработали
<Condition path="" operation="" value="" />	
<Constant/>	
<Conditions>	условия, необходимые для отработки счётчика
<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

Название хука	Описание
<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
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

Название хука	Описание
<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
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. Может быть более одного
<ConditionalAction/> <Actions/> <SetField/>	условные и действия
targetField	поле целевого потока, в которое записывается значение
value=""	значение, которое записывается в targetField
<Conditions/> <Condition/>	условия выполнения маршрута. Может быть более одного
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/>	проверяет, что не существует созданных планов с указанными статусами, если статусы не указан, то используются статусы: Отправлено в ЕИС, Опубликован, Размещен, Загружен, Неактуальный.
<ThrowExceptionIfDocsExists>	проверяет наличие документов, удовлетворяющих запросу и выбрасывает исключение, если хотя бы один документ найден. Перечень атрибутов
query	запрос (возможно указывать параметризованный запрос)

Название хука	Описание
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 связи
<OnLinkAdded>	событие добавления связи
<Aggregation/>	тип обработки события. Выборка всех связанных документов и набор действий
<Select>	выборка документов. В результате получаем набор документов исходного потока
<CurrentDoc />	текущий документ

Настройка конфигурационного файла. Конфигурационный файл DocsLoader представляет собой xml-файл. Для каждого элемента перечисляется список возможных атрибутов и список возможных вложенных элементов.

Если около элемента стоит пометка (мн.) - это означает, что данный элемент множественный, то есть таких элементов можно указать несколько подряд.

6.2.1. Агрегатные функции

Агрегатные функции предназначены для того, чтобы вычислять некоторые для заданного множества строк. Таким множеством строк может быть группа строк, если агрегатная функция применяется к сгруппированной таблице, или вся таблица. Для всех агрегатных функций, порядок вычислений следующий: на основании параметров агрегатной функции из заданного множества строк производится список значений. Затем по этому списку значений производится вычисление функции.

Поддерживаемыми агрегатными функциями являются

- Max – вычисление максимума из заданных значений;
- Min – вычисление минимума из заданных значений;
- Sum – вычисление суммы для заданных значений.

Пример использования в Hooks.xml

```
<Actions>
  <Sum sourceFlow="Лоты_связь_СМП" targetFlow="{current}"
  targetField="|Document|Объем_закупок_СМП" sourceField="|Document|НМЦ_с_НДС"
  useSqlAggregate="yes"/>
```

```
<Min sourceFlow="Заявка на размещение закупки 223Ф3" targetFlow="current"
sourceField="|Document|Плановая_дата_размещения"
targetField="|Document|Порядок_проведения_процедуры|Плановая_дата_размещения" />
</Actions>
```

Для более быстрой выборки рекомендуется использовать атрибут **useSqlAggregate="yes"**, в таком случае агрегатная функция будет вычисляться средствами СУБД (через Хранилище). В данный момент атрибут **useSqlAggregate** реализован только для функции **Sum**.

Если атрибут не указан, вычисление агрегатной функции происходит традиционным способом - путем загрузки всех документов кодом и «ручного» извлечения значений.

6.2.2. Вычисление значения выражений

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

Пример настроек Вычисления значения выражений

```
<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

- **Variables** – содержит 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 Select-ax (атрибут tag);

- path – путь до реквизита, на основе которого вычисляется значение переменной;

- aggregation – тип агрегации, который надо применить к значениям, полученным из реквизита, указанного в path.

Допустимые значения:

- none – вернет первое непустое значение. используется по умолчанию, если атрибут aggregation не указан;

- min – минимальное значение;

- max – максимальное значение;

- Sum – сумма;

- Average – среднее арифметическое;

- Count – количество.

- Expression – выражение, значение которого необходимо вычислить.

6.2.3. Изменение (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 тэге будет отобрано несколько документов-источников, то все Action-ы будут выполняться соответствующее количество раз (таким образом, например, можно подсчитать сумму). При желании в документах-источниках можно указать таблицу-источник (с помощью необязательного элемента SourceTable). В этом случае все Action-ы будут выполняться для каждой строки таблицы в каждом документе. Настройки атрибутов TableFieldOperations представлены в таблице 12.

Таблица 12 - Настройки Атрибутов TableFieldOperations

Наименование Атрибута	Значение Атрибута	Пример
sourceTag	документ(ы)-источник(-и) данных (атрибут tag из раздела Select)	

Наименование Атрибута	Значение Атрибута	Пример
targetTag	целевой(-ые) документ(ы) (атрибут tag из раздела Select)	
targetTable	имя целевой таблицы	
targetRowFilter	формула фильтрации строк целевой таблицы, если не указана - используются все строки, если указана - только строки, подходящие под условия (Подробнее о формулах). В формуле можно использовать подстановки	
{ИмяСтолбцаТаблицы}	Значение берется из указанного поля текущей строки	targetRowFilter="{Этап_закрыт}=1 And {Количество}=0"
{field=ИмяПоля}	значение поля текущего документа-источника	targetRowFilter="{Этап_закрыт}=1 And {field= Document Количество}=0"
{fieldTarget=ИмяПоля}	значение поля целевого документа	formula="{Цена}*({fieldTarget= Doc ument Количество}+100)"
{field=ЗначениеТэга:Имя Поля}	аналогично предыдущему, но значение поля берется из первого документа найденного набора документов (атрибут tag из раздела Select)	targetRowFilter="{Этап_закрыт}=1 And {field=Лоты: Document Количество}=0"
{sourceTableColumn=Имя Столбца}	работает только при наличии элемента SourceTable, значение поля берется из указанного столбца текущей строки таблицы-источника (ИмяСтолбца должно соответствовать атрибуту name в элементе Column)	
checkFormula	формула проверки, которая будет применена к отобранным строкам целевой таблицы (Подробнее о формулах). Если любая строка удовлетворяет формуле, пользователю будет выдано заданное в checkMessage сообщение и сохранение документа будет отменено. В формуле можно использовать подстановки аналогичные targetRowFilter	
checkMessage	сообщение, которое будет выдано пользователю при непрохождении проверки	
Атрибуты таблицы-источника (SourceTable)		

Наименование Атрибута	Значение Атрибута	Пример
tableName	имя таблицы	Возможно указать подчиненную таблицу в следующем виде: " Document Таблица * Подтаблица " - в этом случае будут перебираться все строки всех подтаблиц во всех строках основной таблицы
Атрибуты столбца (Column)		
name	имя столбца таблицы	
aggregate	Операция агрегации, которая будет производиться над значением столбца	
Атрибуты действия (Action)		
column	имя столбца целевой таблицы, в который будет вноситься значение	
formula	формула, результат вычисления которой будет записан в ячейку столбца. В формуле можно использовать подстановки:	
{ИмяСтолбцаТаблицы}	значение берется из указанного поля текущей строки целевой таблицы	formula="{Цена}*{Количество}"
{field=ИмяПоля}	значение поля документа-источника	formula="{Цена}*({field= Document Количество}+100)"
{fieldTarget=ИмяПоля}	значение поля целевого документа	formula="{Цена}*({fieldTarget= Document Количество}+100)"
{field=ЗначениеТэга:ИмяПоля}	аналогично предыдущему, но значение поля берется из первого документа найденного набора документов (атрибут tag из раздела Select)	formula="{Цена}*({field=Лоты: Document Количество}+100)"
{sourceTableColumn=ИмяСтолбца}	работает только при наличии элемента SourceTable, значение поля берется из указанного столбца текущей строки таблицы-источника (ИмяСтолбца должно соответствовать атрибуту name в элементе Column)	

Элементами TableFieldOperations являются:

- **SourceTable** - таблица-источник. Если этого элемента нет, список действий будет выполняться один раз для каждого документа-источника, если есть - будет выполняться 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>
```

Внимание! Чтобы обновлять разные строки по – разному, добавляйте несколько блоков TableFieldOperations с разными фильтрами (targetRowFilter).

6.2.4. 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 тэгам, 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 записываются 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 - добавляется новая.

Таблица 13- Настройки Элементов и Атрибутов

Наименование	Значение
Элементы TableConsolidation	
SourceTable (множественный)	описывает таблицы, из которых будут консолидироваться данные
TargetTable	описывает таблицу, в которую будет записан результат
Элементы таблицы-источника (SourceTable)	
Column (множественный)	описание столбца таблицы
Атрибуты таблицы-источника (SourceTable)	
flow	имя тэга выборки документов (из Select или текущий документ)
tableName	имя таблицы
useDocsAsTable	true - признак того, что в качестве столбцов будет использоваться не таблица, а поля самого документа (используется вместо tableName)
Атрибуты столбца таблицы-источника (Column)	
name	имя столбца таблицы или поля документа
Элементы таблицы-приемника (TargetTable)	
Column (множественный)	описание столбца таблицы
Атрибуты таблицы-приемника (TargetTable)	
flow	имя тэга выборки документов (из Select или текущий документ)
tableName	имя таблицы
Атрибуты столбца таблицы-приемника (Column)	
name	имя столбца таблицы
findExisting	"true" - в данном столбце будет производиться поиск уже существующего значения

Наименование	Значение
aggregate	операция агрегации, которая будет производиться над значением столбца (см. Агрегирование значений)

Пример настроек TableConsolidation

```

<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>

```

6.2.4.1. Настройка правил переноса значений из базового документа в новый и наоборот.

Настройка правил переноса значений из базового документа в новый и наоборот. Находится в файле Templates/Document/Hooks.xml относительно корня легкого клиента. Реализовано две точки изменения:

- FieldsModify - работает в момент сборки шаблона на отображение в браузере, иначе говоря предзаполняет поля.

Далее описаны команды настройки правил переноса значений из базового документа в новый и наоборот:

1) CopyField[@src][@dst] - Копирует значение из поля @src документа-источника в поле @dst нового документа. Если @dst не указан, то он совпадает с @src.

2) SetField[@dst][@value] - Записывает строку @value для поля @dst нового документа.

3) PurchasePlanFieldsModify - Проверяет, что не существует созданных планов с указанными статусами, если статусы не указаны, то используются статусы: Опубликован в ЕИС, Размещен, Загружен, Неактуальный.

Пример:

При нажатии «Сохранить» при создании «План закупок», если уже существует план закупок со статусом перечисленным в атрибуте statuses (выходит сообщение об ошибке).

PurchasePlanFieldsModify

```
<Hooks>
  <FieldsModify>
    <PurchasePlanFieldsModify>
      <ThrowIfExis statuses="Отправлено в
ЕИС;Опубликован;Размещен;Загружен;Неактуальный" message="Формирование нового плана не
доступно при не размещенной версии плана закупок в базе"/>
    </PurchasePlanFieldsModify>
  </FieldsModify>
```

- LinksModify - работает в момент сборки шаблона, добавляет связи документа с другими документами.

Команды:

1) LinkCurrentDoc - связывает документ-источник на новый документ.
 2) LinkAllLinksWhereFlow[@name] - связывает все связи документа-источника у которых имя потока @name на новый документ. Т.е. необходимо скопировать все связи документа-источника у которых имя потока @name в новый документ.

Реализованны следующие проверки для выборки документов:

- WhenParentFieldEquals - проверяет значение поля **path** в родительском документе;

- WhenParentLinksCount - проверяет количество ссылок в потоке **flow** родительском документе.

```
<SetField dst="|Document|Объект_протокола" value="fcsProtocolOK1">
  <And>
    <WhenParentLinksCount flow="Заявки участников" value="any" />
    <WhenParentLinksCount flow="Заявки победителей" value="4" />
    <WhenParentFieldEquals path="|Document|Способ_определения_поставщика_код"
value="OK44" />
  </And>
</SetField>
```

- проверяет, что в родительском документе есть ссылки на документы из потока «Заявки участников»;

- проверяет, что в родительском документе ровно 4 ссылки на документы из потока «Заявки победителей»;

- проверяет, что в родительском документе «Статус» равен «Подача заявок».

6.2.4.2. CopyTable

Данная настройка позволяет копировать таблицы из базового документа в создаваемый.

```
<FieldsModify source="Протоколы">
  <CopyTable src="|Document|Список_документов" />
</FieldsModify>
```

Данная настройка работает в двух режимах - безусловное и настраиваемое копирование. Безусловное копирование представляет собой копирование таблицы, указанное в атрибуте **src**, с базового документа в создаваемые по тому же реквизиту.

Примечание. Безусловное копирование используется при создании документа одного потока. Например, создание протоколов на основании уже созданного протокола.

Настраиваемое копирование представляет собой копирование таблицы с указанием реквизита базового документа и реквизита документа куда происходит копирование. Настраиваемое копирование позволяет копировать таблицы с разных потоков. В атрибуте **src** указана информация откуда брать данные, в атрибуте **dst** указано куда копировать. Для колонок необходимо указывать реквизит относительно таблицы

Пример:

```
<CopyTable src="|Document|Критерии_оценки_заявки_участников"
dst="|Document|Критерии_оценки_заявки_участников">
  <Column src="|Код_критерия" dst="|Код_критерия" />
  <Column src="|Критерий" dst="|Критерий" />
  <Column src="|Значимость_критерия" dst="|Значимость_критерия" />
  <Column src="|Дополнение_критерия" dst="|Дополнение_критерия" />
  <Column src="|Лимит_критерия" dst="|Лимит_критерия" />
  <Column src="|Порядок_оценки_критерия_код"
dst="|Порядок_оценки_критерия_код" />
  <Column src="|Порядок_оценки_критерия" dst="|Порядок_оценки_критерия" />
  <Column src="|Код_показателя_оценки" dst="|Код_показателя" />
  <Column src="|Наименование_показателя" dst="|Наименование_показателя" />
  <Column src="|Код_порядка" dst="|Код_порядка" />
  <Column src="|Порядок_оценки_по_показателю"
dst="|Порядок_оценки_по_показателю" />
  <Table src="|Document|Участники_комиссии" dst="|Оценка_комиссии">
    <Column src="|Номер" dst="|Член_комиссии_осуществляющий_оценку"/>
  </Table>
</CopyTable>
```

Так же поддерживается копирование во вложенные таблицы, для этого используется тэг **Table**.

Правила отображения.

Правила настройки отображения поддерживаются так же и для остальных хуков. Данные правила должны быть записаны в тэге Visibility внутри тэга CopyTable.

Пример Правила отображения:

```
<CopyTable src="|Document|Критерии_оценки_заявки_участников"
dst="|Document|Критерии_оценки_заявки_участников">
    <Column src="|Код_критерия" dst="|Код_критерия" />
    <Column src="|Критерий" dst="|Критерий" />
    <Column src="|Значимость_критерия" dst="|Значимость_критерия" />
    <Visibility>
    <And>
        <WhenAlwaysTrue />
    </And>
    </Visibility>
</CopyTable>
```

6.2.4.3. Нумерация лотов/позиций плана (SetLotSequentialNumbers)

Хук `<SetLotSequentialNumbers>` производит автонумерацию лотов/позиций для плана. Входные параметры:

- атрибут `planInnoFlagPropertyName` определяет наименование атрибута плана с флагом инновационной продукции;
- атрибут `planYearPropertyName` определяет наименование атрибута плана «Год»;
- атрибут `planOrgZakPropertyName` определяет наименование атрибута плана с организацией заказчиком;
- вложенный тег `<MaxLotNumQuery>` содержит запрос для выборки документов (лотов/позиций) и определения максимального номера позиции в них;
- вложенный тег `<FirstTimeInPlanQuery>` содержит запрос для выборки документов (лотов/позиций) которые требуется пронумеровать, начиная с максимального номера из выборки `MaxLotNumQuery`.

Во вложенные запросы подставляются параметры:

- {0} - год закупки;
- {1} - флаг инновационной продукции;
- {2} - идентификатор организации заказчика.

Пример использования хука (xml)

```
<SetLotSequentialNumbers
planInnoFlagPropertyName="doc_RegCard/rc_Index/boolean_Инновационная_продукция"

planYearPropertyName="doc_RegCard/rc_Index/integer_Год"

planOrgZakPropertyName="doc_RegCard/rc_Index/integer_Организация_заказчик">
    <MaxLotNumQuery>
```

```

        (doc_RegCard/rc_FlowName = 'Позиции плана закупок 223ФЗ')
and (doc_RegCard/rc_Index/integer_Год_закупки = '{0}') and
(doc_RegCard/rc_Index/boolean_Инновационная_продукция = '{1}') and
(doc_RegCard/rc_Index/integer_Организация_заказчик = '{2}')
    </MaxLotNumQuery>
    <FirstTimeInPlanQuery> (doc_RegCard/rc_FlowName = 'Позиции плана
закупок 223ФЗ') and (doc_RegCard/rc_Index/integer_Год_закупки = '{0}') and
(doc_RegCard/rc_Index/boolean_Инновационная_продукция = '{1}') and
(doc_RegCard/rc_Index/integer_Организация_заказчик = '{2}')
        and not (xn-in(doc_RegCard/rc_Index/text_Статус,'Подготовка
ППЗ','Доработка ППЗ','Подготовка лота','Согласование','Согласовано руководителем
инициатора','Согласовано для вынесения на ЦЗК','Согласовано','Отклонено','Согласование
лота')) and (not((doc_RegCard/rc_Index/integer_Номер_позиции_плана)))
    </FirstTimeInPlanQuery>
</SetLotSequentialNumbers>

```

6.2.4.4. Проверка выражений

Данное действие выполняет проверку истинности какого-либо выражения, в котором участвуют значения из разного набора документов.

Пример настроек:

```

<VerifyExpression>
    <Variables>
        <Variable name="СуммаППГ" source="Позиции плана-
графика" path="|Document|Сумма_всего" />
        <Variable name="НМЦКППГ" source="Позиции плана-
графика" path="|Document|НМЦК" />
    </Variables>
    <Expression value="$NullToZero({НМЦКППГ}) &lt;=
$NullToZero({СуммаППГ})" />
    <FailureText value="НМЦК превышает допустимую сумму в
рамках ППГ" />
</VerifyExpression>

```

Variables- содержит 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 Select-ах (атрибут tag);
- path - путь до реквизита, на основе которого вычисляется значение переменной;
- aggregation - тип агрегации, который надо применить к значениям, полученным из реквизита, указанного в path.

Допустимые значения:

- none - вернет первое непустое значение. используется по умолчанию, если атрибут aggregation не указан;
- min - минимальное значение;
- max - максимальное значение;

- sum – сумма;
- averagge - среднее арифметическое;
- count – количество.
- expression - выражение, истинность которого необходимо проверить. В данном значении описывается формула.
- failtureText - текст исключения в случае, когда выражение не истинно.

6.2.4.5. Проверка количества найденных документов (DocsCountCheck).

Действие позволяет проверить количество документов, отобранных в Select-е с конкретным тэгом.

Элемент выполняет операцию проверки количества отобранных документов. В случае, если операция выполнена успешно, текущая транзакция отменяется (если проверка выполняется, например, при сохранении документа, документ сохранен не будет, на экране появится запрограммированный текст ошибки).

Внимание! Для того, чтобы проверка могла отменить изменения документа, хуки должны выполняться в одной транзакции.

Для элемента DocsCountCheck используются такие атрибуты как:

- **sourceTag** - ссылка на атрибут tag из Select-а, отбирающего нужные документы;

- **operation** - операция проверки:

- **empty** - если список пустой;

- **notEmpty** - если список не пустой.

- **message** - если операция выполняется успешно, пользователю выводится указанный текст ошибки. В сообщении возможно использовать подстановки из полей текущего документа (в фигурных скобках), например, message="Документы со статусом {Document|Статус} уже существуют"

Пример:

```
<OnDocumentCreated>
  <Aggregation>
    <Select>
      <CurrentDoc />
      <Link flow="Контракты" />
    </Select>
    <Actions>
      <SetField targetFlow="Контракты" targetField="|Document|Статус"
value="Заключение договора" />
      <SetFieldGuid targetFlow="current"
targetField="|Document|ИД_в_ЕИС" />
      <CopyField sourceFlow="Контракты" targetFlow="current"
sourceField="%key%" targetField="|Document|ИД_договора" />
    </Actions>
  </Aggregation>
</OnDocumentCreated>
```

```
</Aggregation>
</OnDocumentCreated>
```

Примечание. В случае, если данное действие для проверки существования документов, аналогичных сохраняемому, при редактировании необходимо исключить текущий (сохраняемый) документ из поиска, иначе возможно ложное срабатывание, если проверяемые параметры не менялись:

```
<OnDocumentCreated>
  <Aggregation>
    <Select>
      <Select searchString="document[doc_Id != {current_Id} and .....]" />
```

6.2.4.6. Проверка количества строк в таблице (TableRowsCountCheck)

Действие позволяет проверить количество строк в таблице в текущем документе, либо в отобранном документе.

Элемент выполняет операцию проверки количества строк в таблице отобранного документа. В случае, если операция выполнена успешно, текущая транзакция отменяется (если проверка выполняется, например, при сохранении документа, документ сохранен не будет, на экране появится запрограммированный текст ошибки).

Внимание! Для того, чтобы проверка могла отменить изменения документа, хуки должны выполняться в одной транзакции.

Для элемента TableRowsCountCheck используются атрибуты такие как:

- **sourceTag** - ссылка на атрибут tag из Select-а, отбирающего нужные документы;

- **tableName** - название таблицы;

- **operation** - операция проверки:

- **equals** - число строк равно value;

- **less** - число строк меньше value;

- **greater** - число строк больше value;

- **value** - значения для сравнения;

- **ifCheckedAttrs** (**необязательный параметр**) - проверка выполняется только в случае, если реквизиты в ifCheckedAttrs true (реквизиты перечисляются через запятую);

- **message** - если операция выполняется успешно, пользователю выводится указанный текст ошибки. В сообщении возможно использовать подстановки других параметров хука (в фигурных скобках), например message="В таблице {tableName} число строк {operation} {value}".

Пример:

```
<OnDocumentCreated>
```

```

<Aggregation>
  <Select>
    ...
  </Select>
  <Actions>
    <TableRowsCountCheck sourceTag="{current}"
    tableName="|Document|Позиции_КТРУ" operation="less" value="2"
    ifCheckedAttrs="|Document|Содержит_несколько_позиций_ОКПДКТРУ"
    message="В таблице {tableName} число строк {operation} {value}" />
  </Actions>
</Aggregation>
</OnDocumentCreated>

```

6.2.4.7. Сравнение с нормированным значением (RationingCheck)

Нормированное значение для текущего документа находится в документах, указанных в атрибуте "**rationingDocsTag**". Необходимо обработать каждый документ поочередно, до того, как будет найдено нормированное значение.

Для того чтобы осуществить поиск нормированного значения в документе необходимо осуществить:

- 1) поиск в таблице документа, указанной в тэге **<RationingTable/>**;
- 2) в таблице найти поля, содержащие ключ организации и нормированное значение (эти поля соответственно указываются в атрибутах **orgKeyColumn** и **valueColumn**)
- 3) если ключ организации в таблице равняется ключу организации, указанной у текущего документа, то значение в колонке указанной **valueColumn** является нормированным;
- 4) если в таблице отсутствует значение, необходимо перейти к следующему документу;
- 5) если ни в одной таблице нормированных документов не найдено значения, то выберите нормированные документы только те, у которых в поле **orgKey** элемента **<RationingDocument/>** указан ключ организации определенной в тэге **<RootOrganization/>** (атрибут **name**);
- 6) если документы найдены необходимо использовать первый из них и считать значение поля, указанного в атрибуте **field** элемента **<RationingDocument/>** нормированным;
- 7) если нормированное значение не найдено - ничего не осуществляем. Если значение найдено, то приступаем к сравнению нормированного значения и со значением документа, указанного в атрибуте **field** элемента **<CurrentDocument/>**.

Для определения типа сравнения необходимо указать атрибут **condition** у элемента **<RationingCheck>..</RationingCheck>**. Поддерживаемые типы условий:

- Less (меньше);
- LessOrEqual (меньше или равно);

- More (больше);
- MoreOrEqual (больше или равно);
- Equal (равно).

Например, если указан тип «Less», это говорит о том, что текущее значение обязано быть меньше нормированного.

В случае возникновения ошибки, сообщение об ошибке указывается в атрибуте **message** элемента **RationingCheck**. Может содержать реквизиты текущего документа, а также специальный реквизит **{rationing}**, который будет заменен на полученное нормированное значение.

6.2.4.8. Условные действия (ConditionalActions)

Для условных действий в разделе **Select**, производится выборка документов.

Внутри элемента **Actions** содержится список действий (элементы **ConditionalAction**), для каждого из которых определены условия (содержимое элемента **Conditions**) и действия (**Actions**), выполняемые при выполнении условий.

Во многих проверках присутствует атрибут **tag**, значение которого должно соответствовать значению атрибута **tag** элемента **Select**, в котором производится выборка документов. Значение этого атрибута определяет набор документов, по отношению к которым будут проверки. Если атрибут не указан - набором документов для проверки будут все отобранные документы.

Реализованные проверки для выборки документов:

- **WhenAllHasActiveProcess** - проверяет, что у всех отобранных документов есть активный маршрут с названием **name**;

- **WhenFieldsEquals** - проверяет, что у всех отобранных документов значения реквизитов, указанных в **path**, равны;

- **WhenFieldsNotEquals** - проверяет, что у какого-либо документа из отобранных значение реквизита по пути **path** отличается от других;

- **WhenFieldsStartsWith** - проверяет, что у всех отобранных документов первые **length** символов значения реквизита по пути **path** совпадают;

- **WhenAnyFieldMassCheck** - массовая проверка значений (правило проверяет указанное поле в наборе документов, если любое из полей удовлетворяет условиям, правило дает положительный ответ);

- **WhenFormula** - проверяет, что указанное в формуле выражение выполняется для всех участвующих в ней документов из параметров. Правила определения формул:

- каждый параметр определяется **xml**-атрибутом, начинающимся с префикса *"param-"*, после которого следует само название параметра из формулы. Например, параметр *"someParam1"* задается атрибутом *"param-someParam1"*

- значение параметра формируется в следующем формате *"<тег выборки документов из выражения Select либо предопределенные выборки,*

например `{current}`><путь к реквизиту документа для подстановки в формулу>".
Примеры: `"{current}|Document|Номер_редакции"`,
`"Первая_Версия_Плана|Document|Год"`.

- параметры документов для подстановки в формулы должны быть *вещественного* типа. Если же параметр является строкой - будет произведена попытка конвертации в вещественное число.

- если реквизиты документов, подставляемые в формулу, содержат пустое значение (`null`), условие не будет выполняться. В таких случаях возможно использование уже существующих подстановок `$NullToZero`, `$NullToEmpty`

- формула определяется атрибутом `"formula"` и может содержать некоторые арифметические операции.

Примеры каждой из проверок. Предполагается, что в разделе `Select` есть выборка с `tag="Позиции"`.

Пример:

```
<WhenFieldsNotEquals tag="Позиции" path="|Document|Единица_измерения"/>
<WhenFieldsEquals tag="Позиции" path="|Document|Код_классификатора"/>
<WhenAllHasActiveProcess tag="Позиции" name="Согласование заявки"/>
<WhenFieldsStartsWith tag="Позиции" path="|Document|ОКПД2" length="5"/>

<WhenFormula param-someParam1="Первая_Версия_Плана|Document|Год" param-
someParam2="{current}|Document|Номер_редакции"
formula="{someParam1}+0.1*{someParam2}>=999999"/>
```

6.3. Сервисные плагины и утилиты

`DocsLoader` – библиотека для унифицированной загрузки документов E1 из различных внешних источников.

Библиотека может создавать новые или обновлять поля существующих документов E1, в том числе создание/обновление подчиненных/связанных документов, в том числе и с несколькими уровнями подчинения).

На основе библиотеки создаются плагины автоимпорта и утилиты, предназначенные для загрузки из разных источников, с различным интерфейсом.

В том числе библиотека кроме режима загрузки документов E1, имеет возможность работать в режиме сравнения. В режиме сравнения не производится фактического сохранения документов, вместо этого полученные значения сравниваются с уже имеющимися в документе E1 и выдается список полей, значения которых не совпадают.

Загрузка документов управляется конфигурационным файлом, который задает:

- параметры для источника данных для документов;
- параметры предварительных проверок источника на пригодность;

- способы создания/поиска документов Е1;
- маппинг - соответствие заполнения полей документов из источника;
- дополнительные параметры маппинга;
- описание справочников Е1.

В том числе для определенных действий при загрузке могут быть использованы модули кода, которые выполняют действия, которые невозможно настроить с помощью конфигурационного файла.

Конфигурация настраивается единообразно для всех использующих библиотеку утилит и плагинов, за исключением настроек для получения данных из источника.

DictsLoader загрузка справочников

Visual Mapper программа для просмотра интеграционных схем для отправки документов в ЕИС

6.4. Настройка ЭП

6.4.1. Общие сведения

В состав комплекса «Е1-Закупки» включена поддержка криптографической системы на базе системного компонента «КриптоПро ЭЦП Runtime». Это позволяет устанавливать подлинность электронных документов, зарегистрированных в комплексе. Под подлинностью документа понимается его неизменность и авторство. Подлинность документа устанавливается на основании проверки ЭП, которая проставляется пользователем с помощью средств криптосистемы. ЭП может быть сформирована для любого электронного документа, содержащего присоединенные файлы, с сохранением уникальных признаков подписавшего документ пользователя.

Электронная подпись (ЭП) — это последовательность символов, позволяющая установить, что исходный текст не был изменен с тех пор, как его подписал (сформировал ЭП этого текста) указанный пользователь. Следует подчеркнуть, что с точки зрения применяемых алгоритмов и принятой логики работы в комплексе «Е1-Закупки» ЭП не является выражением отношения пользователя, подписавшего документ, к содержанию документа (согласен или не согласен). ЭП — это средство проверки неизменности документа. Пользователь, поставивший в электронном документе свою ЭП, подтверждает этим, что видел документ именно таким. В комплексе «Е1-Закупки» один и тот же документ может быть подписан несколькими пользователями. Впоследствии как пользователь, подписавший документ, так и другие пользователи могут проверить каждую из ЭП документа. Проверка показывает, верна подпись или неверна. Первое

означает, что документ не изменялся и его подписал именно тот сотрудник, имя которого указано в комплексе. Второе означает, что документ был изменен либо был подписан не тем сотрудником, имя которого указано в комплексе. Когда пользователь комплекса ставит в электронном документе свою ЭП, фактически подпись формируется для каждого из присоединенных файлов данного документа. При проверке ЭП документа криптосистема делает заключение о подлинности ЭП пользователя только в случае, если верны все подписи, сформированные для присоединенных файлов. ЭП также может быть сформирована для присоединенных файлов. В этом случае аналогично документу, ЭП может быть сформирована отдельно для присоединенных файлов. Процедуры подписания и проверки ЭП присоединенных файлов аналогичны подписанию и проверке ЭП документа. ЭП создается путем преобразования текста электронного документа. Само преобразование, а также ряд других действий, осуществляются специальным программным модулем — криптопровайдером. При формировании ЭП используются *алгоритмы подписи* и *несимметричного шифрования* данных. Метод несимметричного шифрования характеризуется тем, что шифрование и дешифровка текста производятся с помощью различных ключей, составляющих ключевую пару или, другими словами, пару «*закрытый-открытый ключ*». Названия ключей «закрытый» и «открытый» отражают тот факт, что закрытый ключ, принадлежащий какому-либо пользователю, сохраняется втайне. Тогда как соответствующий ему открытый ключ распространяется среди пользователей, проверяющих ЭП. Фактически при формировании ЭП производится шифрование *хэша* документа закрытым ключом пользователя. При проверке ЭП происходит следующее: создается хэш проверяемой версии документа; хэш подписанной версии документа расшифровывается открытым ключом подписавшего пользователя; два полученных хэша сравниваются, и в результате устанавливается, был ли изменен исходный текст с тех пор, как в нем поставлена ЭП.

Для работы с ЭП необходима тождественность следующих параметров на компьютерах пользователей, которые будут подписывать документы или проверять ЭП других пользователей:

- используемого криптопровайдера;
- алгоритма подписи;
- длин ключей (открытого и закрытого ключей, используемых в алгоритме несимметричного шифрования).

Подписание электронных документов и присоединенных файлов, а также проверка ЭП осуществляются на рабочих местах пользователей. При проверке ЭП электронного документа возможны случаи, когда не удается определить, верна ЭП или неверна. В этом случае результатом диагностики является сообщение: «Не удалось проверить ЭП». Такая ситуация возникает вследствие смены криптопровайдера или изменения параметров криптопровайдера (см. п. 3.4.4).

Комплекс «Е1-Закупки» поддерживает работу *неквалифицированной* ЭП. Для работы с неквалифицированной ЭП достаточно на рабочем месте каждого из пользователей, которые будут подписывать ЭП документы или проверять ЭП других пользователей, установить криптопровайдер (например, «КриптоПро CSP») и приобрести лицензию на него для каждого рабочего места.

6.4.2. Общий порядок настройки ЭП

При первоначальной настройке ЭП рекомендуется придерживаться определенного порядка действий.

Общий порядок настройки ЭП:

8) при установке инсталляционного пакета «Е1-Закупки. Клиент» на рабочих местах пользователей, которые будут подписывать документы или проверять ЭП других пользователей, установите системный компонент «КриптоПро ЭЦП Runtime»;

9) выполните настройку ЭП в модуле «Администратор»;

10) на рабочих местах пользователей, которые будут подписывать документы или проверять ЭП других пользователей, установите криптопровайдер и другие компоненты (при необходимости) и настройте параметры этих компонентов;

11) на рабочих местах пользователей, которые будут подписывать документы или проверять ЭП других пользователей, сформируйте сертификаты ключа подписи (см. п. 3.4.3).

Криптопровайдер устанавливается на все компьютеры, где предполагается применять ЭП для зарегистрированных в комплексе «Е1-Закупки» документов. Допустимо использование криптопровайдера «КриптоПро 3.9», который совместим с системным компонентом «КриптоПро ЭЦП Runtime».

Далее рассмотрим действие для проверки ЭЦП средствами КриптоПро. Для того, чтобы проверить ЭЦП прикрепленного файла необходимо:

1) создать новый документ, перейти в созданный документ, присоединить файл;

2) подписать присоединенный файл;

3) нажать «просмотреть ЭП» и скачать ЭЦП в этом диалоговом окне;

4) скачать утилиту КриптоПро `cryptcp`, для проверки ЭЦП;

5) скопировать оригинальный файл, ЭЦП и утилиту на диск C:\ (или в другую общую папку);

6) запустить командную строку от имени администратора (правой кнопкой мышки «Запустить от имени администратора»);

7) перейти в директорию с файлом, ЭЦП, утилитой;

8) выполнить команду:

`-cryptcp.x64.exe -vsignf OriginalFile.docx.`

9) убедиться, что команда выполнена успешно, отображается верная информация о сертификате, присутствует запись, что ЭЦП верна, и код ошибки [0x00000000], т.е. что ошибки нет и утилита отработала штатно;

Пример успешной валидации подписи с использованием утилиты `cryptcp.x64`.

```

Администратор: Командная строка
c:\>cryptcp.x64.exe -vsignf Test10.docx
CryptCP 5.0 (c) "КРИПТО-ПРО", 2002-2017.
Утилита командной строки для подписи и шифрования файлов.

Найдено сертификатов: 2
Цепочки сертификатов проверены.
Папка '.\':
Test10.docx... Проверка подписи...

Автор подписи: RU, Москва, org, dep, Фамилия Имя Отчество, test@test.ru
Подпись проверена.
[ErrorCode: 0x00000000]

c:\>
  
```

Рисунок 11

Аналогично, валидацию подписи можно произвести с помощью утилиты КриптоАртм. Преимуществом такого подхода является возможность просмотреть штамп времени (если при подписании использовался tsp сервер.

Для валидации ЭЦП через КриптоАРМ необходимо:

- 1) запустить утилиту КриптоАРМ;
- 2) нажать кнопку «Проверить ЭП»;

Валидация ЭЦП через КриптоАРМ

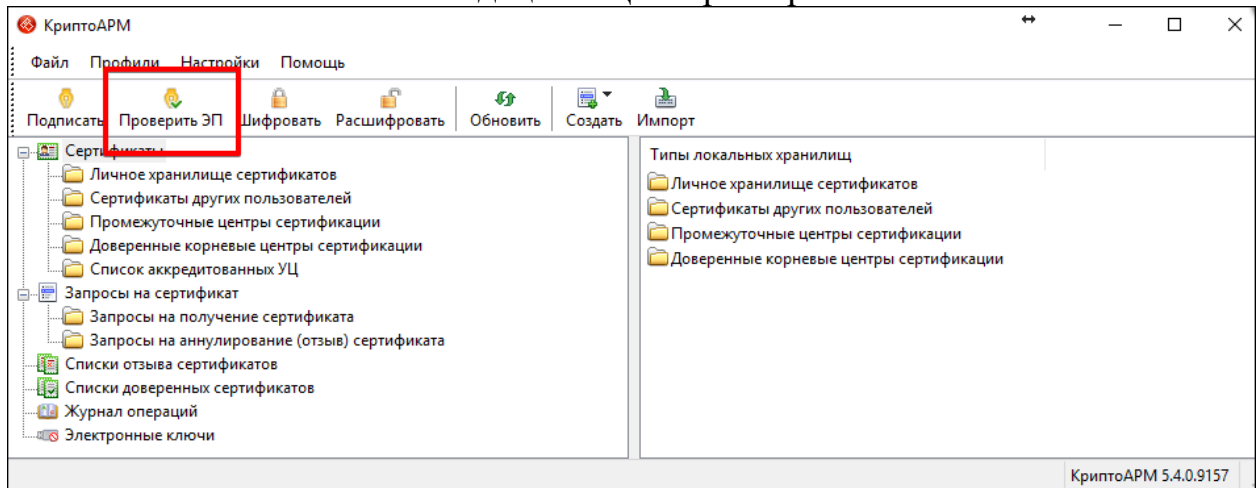


Рисунок 12

- 3) в диалоге выбора профиля, нажать «Далее»

Приветствие мастера

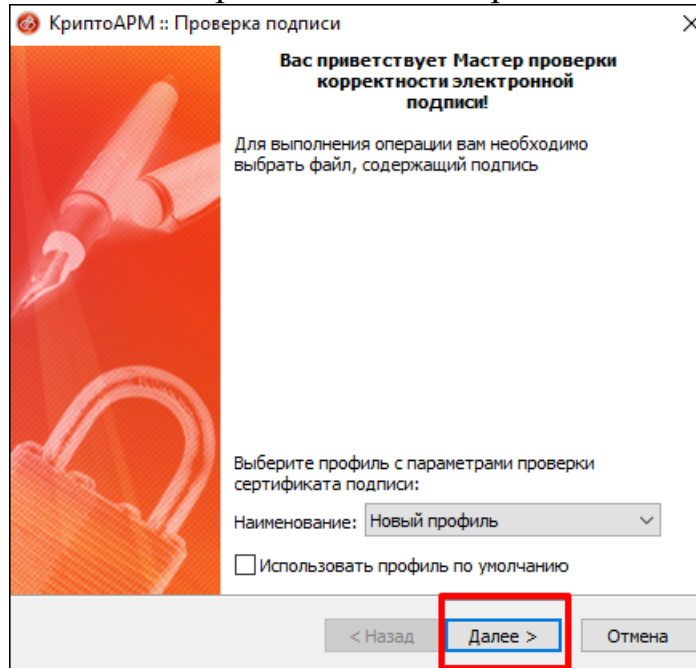


Рисунок 13

4) выбрать «.sgn» файл подписи;

Пример окна для выбора файла

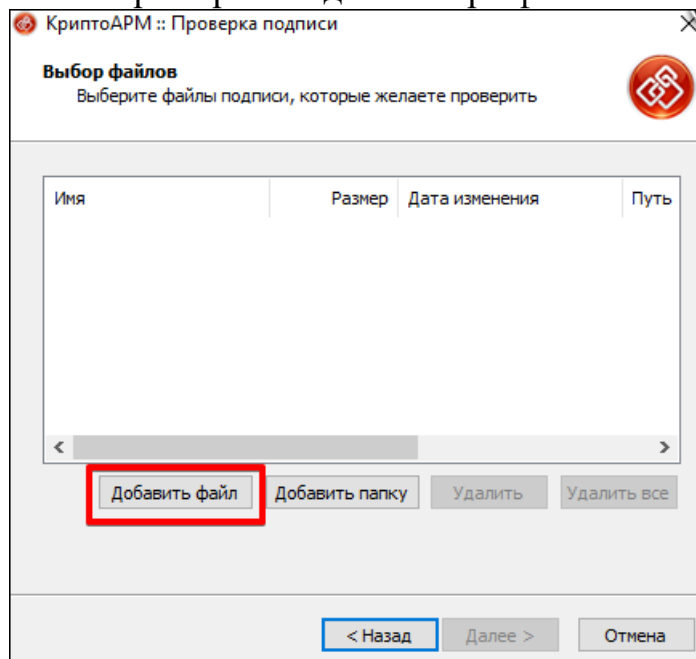


Рисунок 14

5) убедиться, что исходный файл был выбран автоматически, после выбора файла подписи;

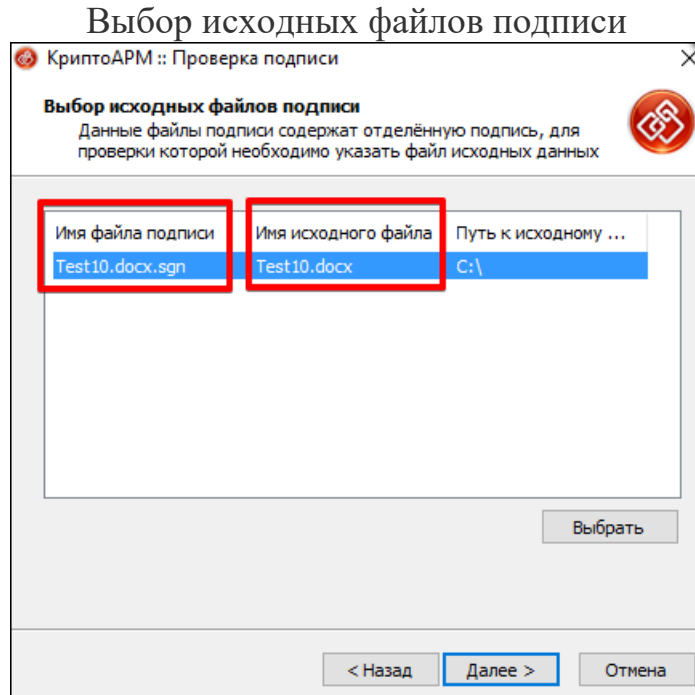


Рисунок 15

б) убедиться, что проверка завершилась со статусом «Успех»;

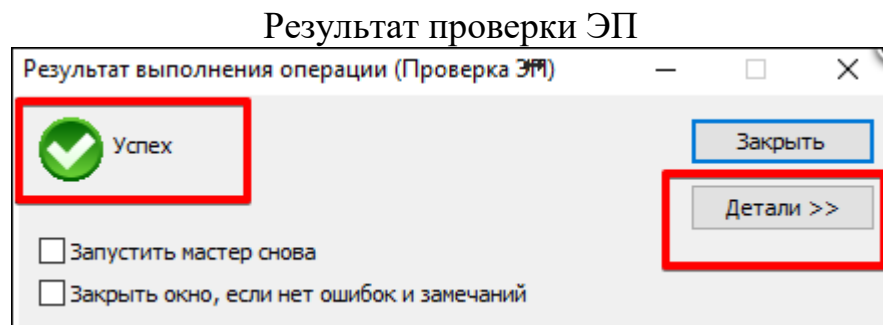


Рисунок 16

7) далее нажать «Детали» -> «Менеджер сообщения» -> выбрать подпись и нажать «Просмотреть», в появившемся окне перейти на вкладку «Штамп времени».

Просмотр штампа времени

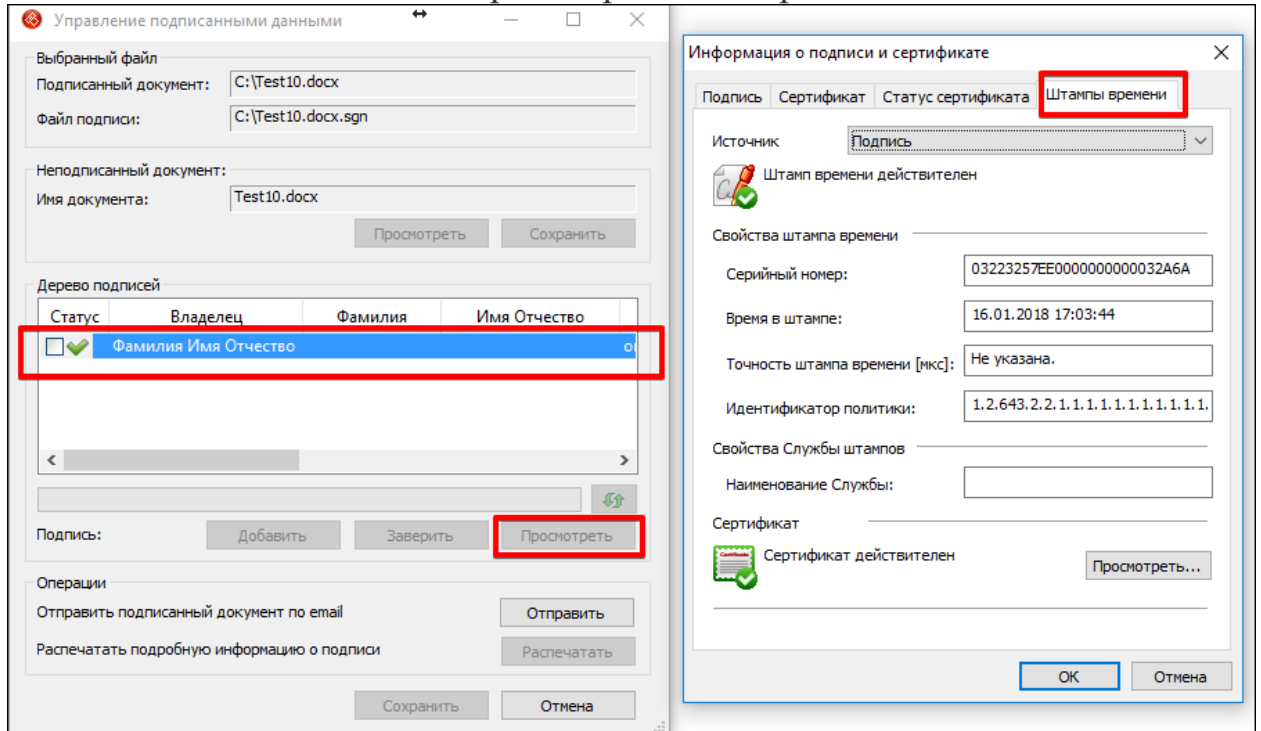


Рисунок 17

Для того, чтобы осуществить подготовку сервера и клиента к настройке необходимо:

- 1) установить КриптоПро CSP и КриптоПро browser plugin на клиенте;
- 2) установить КриптоПро CSP и КриптоПро .NET на сервере;
- 3) установить корневые сертификаты и csl списки отзыва (на сервере и на клиенте) из папки сертификаты.zip;

- после установки появится два сертификата с одинаковым названием, один сертификат лишний его необходимо удалить.

- 4) добавить личный сертификат через КриптоПро CSP из папки сертификаты.zip;

- для этого распаковать архив и скопировать содержимое папки «сертификат» на флэшку (в корне флэшки должен оказаться сертификат «Testov_Test_Testovich.cer» и папка «ibnokygi.000»);

- установить сертификат с флэшки;

- далее запустить утилиту КриптоПро CSP, перейти на вкладку «Сервис» и нажать «Установить личный сертификат».

Пример окна при установке личного сертификата

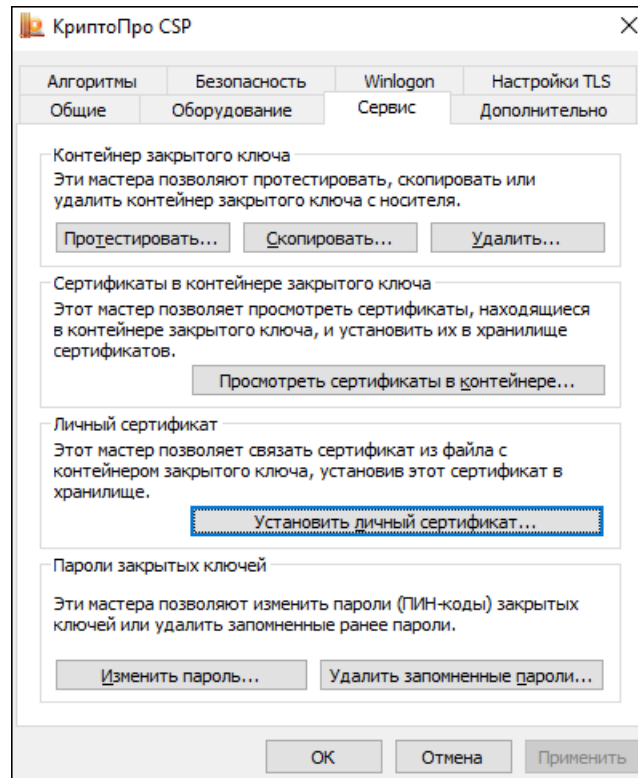


Рисунок 18

- в «Мастере установки» проставить чек-бокс «Найти контейнер автоматически»;

Окно Мастера установки личного сертификата

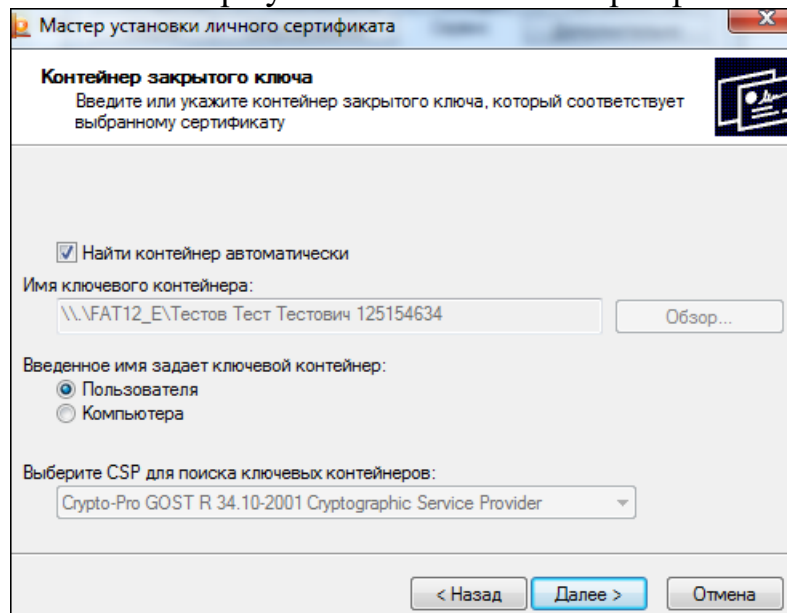


Рисунок 19

- пароль на сертификате «000000»;
 - скопировать в реестр (иногда срабатывает со второго раза);
 - запустить утилиту КриптоПро CSP, перейти на вкладку «Сервис» и нажать «Скопировать», и далее «По сертификату...» и выбрать сертификат «ГАУ АО «Управление ИКТ АО»».

Окно Мастера копирования контейнера закрытого ключа

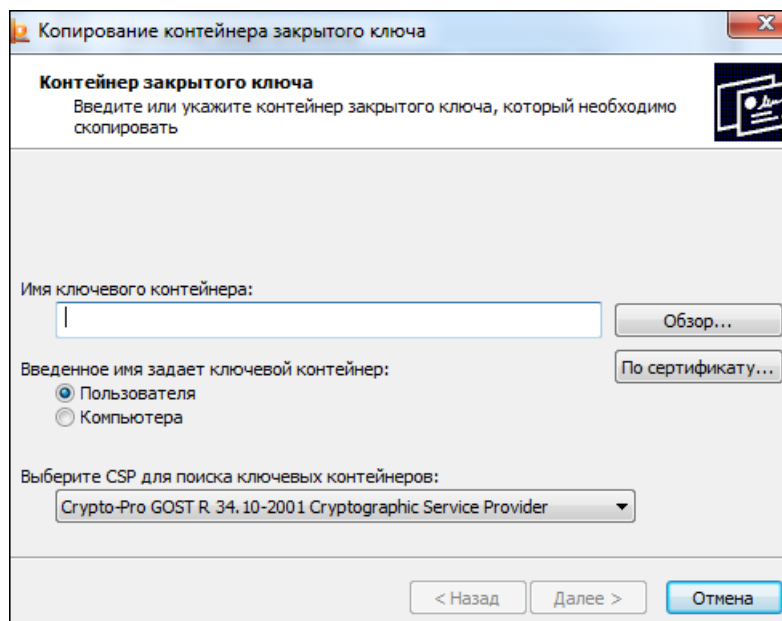


Рисунок 20

- введите любое имя Ключевого контейнера и нажмите «Готово»
- в появившемся диалоге выберите «Реестр» и нажмите «Ок»
- установите новый пароль и нажмите «Ок».

5) прописать пользователю почту из сертификата, чтобы заработал вход по ЭП

6) в двух конфигурационных файлах (Web.config и EdsAccessGroup.xml) прописать адрес TSP сервера заказчика <http://tsp.dvinaland.ru:8777/tsp>

- C:\Program Files (x86)\Cognitive\Euphrat\Server\Web.config

- C:\Program Files (x86)\Cognitive\Euphrat\Server\Templates\Plugins\EdsAccessGroup.xml

6.4.3. Формирование новых сертификатов ключей подписи

Каждому пользователю комплекса «Е1-Закупки», желающему ставить или проверять ЭП в электронных документах комплекса, требуется установить

сертификат ключа подписи на его рабочем месте. Сертификат ключа подписи содержит информацию о паре «закрытый-открытый ключ». Закрытый ключ используется при формировании ЭП. Этот ключ недоступен другим пользователям и хранится на компьютере своего владельца, что обеспечивается криптопровайдером. С другой стороны, для того чтобы остальные пользователи могли проверять ЭП пользователя, подписавшего документ, им должен быть доступен его открытый ключ. В этих целях открытые ключи сразу же после формирования сертификата ключа подписи автоматически публикуются на сервере комплекса. Сертификаты ключа подписи следует формировать после того, как на компьютерах пользователей установлены тождественные параметры криптопровайдера. В этих целях

пользователи вместо самостоятельной настройки криптопровайдера получают описание параметров с сервера комплекса.

Возможны следующие основные причины формирования нового сертификата ключа подписи:

- произошла смена криптопровайдера (см. п. 3.4.4);
- истек срок действия сертификата ключа подписи. Периодическая замена сертификатов ключей подписи (не реже одного раза в год) предусматривается общими правилами работы со всеми криптографическими системами. Для того чтобы получить новый сертификат ключа подписи, следует выполнить процедуру, предлагаемую при настройке криптопровайдера. В некоторых случаях существует возможность получить такой сертификат через интернет с помощью специального сервиса либо на отдельном носителе.

6.4.4. Особенности, связанные со сменой криптопровайдера или изменением параметров криптопровайдера

В процессе эксплуатации поддерживаемой в комплексе «Е1-Закупки» криптосистемы может потребоваться сменить используемый криптопровайдер или какие-либо параметры шифрования (алгоритмы, длины ключей).

Для того чтобы сменить используемый криптопровайдер:

- 1) установите новое матобеспечение (новый криптопровайдер) на компьютерах всех пользователей, которые будут подписывать документы или проверять ЭП других пользователей;
- 2) сформируйте новые сертификаты ключа подписи (см. п. 3.4.3) и установите их для каждого пользователя, которые будут подписывать документы или проверять ЭП других пользователей. Это необходимо выполнить, поскольку после смены криптопровайдера отсутствует возможность проверить все полученные ранее сертификаты ключа подписи.

СПИСОК СОКРАЩЕНИЙ

БД	— база данных
ЕИС	— Единая информационная система в сфере закупок
ЕЭТП	— Единая электронная торговая площадка
ОС	— операционная система
Система	— Информационная система управления закупками
	— Главный распорядитель бюджетных средств
ГРБС	— получатель бюджетных средств
ПБС	— уполномоченный орган
УО	
ИС	— информационная система
НМЦД	— начальная максимальная цена контракта
ЭП	— электронная подпись
РК	— регистрационная карточка