



**ИСИДА-ИНФОРМАТИКА**

---

**Система управления бизнес-процессами  
ISIDA BPM**

*Версия 8.\**

**Руководство по использованию**

## Содержание

<b>1. ОБЩИЕ СВЕДЕНИЯ О СИСТЕМЕ .....</b>	<b>3</b>
1.1. ОБЪЕКТЫ И ПРОЦЕССЫ .....	3
1.2. ЗАДАЧИ .....	5
1.3. ПРИЛОЖЕНИЯ.....	6
1.4. РАБОТА С ЭЛЕКТРОННЫМИ ДОКУМЕНТАМИ.....	7
<b>2. КОНСТРУКТОР ОБЪЕКТОВ .....</b>	<b>9</b>
2.1. ОБЩИЕ СВЕДЕНИЯ .....	9
2.2. СОЗДАНИЕ ОБЪЕКТОВ .....	11
2.3. АДАПТАЦИЯ ОБЪЕКТОВ.....	15
2.4. МАКЕТ УВЕДОМЛЕНИЯ .....	16
<b>3. ЭЛЕКТРОННЫЕ ДОКУМЕНТЫ.....</b>	<b>17</b>
3.1. ОБЩИЕ СВЕДЕНИЯ .....	17
3.2. СОЗДАНИЕ И НАСТРОЙКА ТИПА ЭЛЕКТРОННОГО ДОКУМЕНТА.....	18
<b>4. СПРАВОЧНИКИ.....</b>	<b>21</b>
4.1. СПРАВОЧНИК ПОДРАЗДЕЛЕНИЙ .....	21
4.2. СПРАВОЧНИК СОТРУДНИКОВ.....	22
4.3. РАБОЧИЙ КАЛЕНДАРЬ .....	24
4.4. СПРАВОЧНИКИ ПРИЛОЖЕНИЙ.....	25
<b>5. ДИЗАЙНЕР ПРОЦЕССОВ.....</b>	<b>27</b>
5.1. ОБЩИЕ СВЕДЕНИЯ .....	27
5.2. СПИСОК ПРОЦЕССОВ .....	28
5.3. СОЗДАНИЕ ПРОЦЕССА .....	29
5.4. НАСТРОЙКА ТОЧЕК МАРШРУТА .....	33
5.5. ВЕТВЛЕНИЯ И СЛИЯНИЯ В ПРОЦЕССЕ .....	36
5.6. ИСПОЛЬЗОВАНИЕ ПОДПРОЦЕССОВ .....	42
5.7. МЕНЕДЖЕР ЗАДАЧ .....	44
5.8. УНИВЕРСАЛЬНАЯ ЗАДАЧА GENERIC TASK.....	49
5.8.1. <i>Общая структура интерфейса универсальной задачи</i> .....	49
5.8.2. <i>Настройка универсальной задачи в Дизайнере процессов</i> .....	51
5.8.3. <i>Настройка маршрутной кнопки для задачи</i> .....	52
5.8.4. <i>Настройка возвратных кнопок</i> .....	61
5.8.5. <i>Настройка кнопок без движения</i> .....	61
5.8.6. <i>Задача «Изменение полей объекта»</i> .....	62
5.9. НАСТРОЙКА ЗАДАЧ ДЛЯ ЭЛЕКТРОННЫХ ДОКУМЕНТОВ .....	64
5.10. ДОЧЕРНИЕ ТОЧКИ .....	68
5.11. ВЫБРОС ПРИКЛАДНЫХ СОБЫТИЙ .....	70
5.12. ПЛАНОВЫЕ СОБЫТИЯ .....	73
<b>6. ДИСПЕТЧЕР ПРОЦЕССОВ.....</b>	<b>77</b>
6.1. ОСНОВНЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ .....	77
6.2. ЗАПУСК ЭКЗЕМПЛЯРА ПРОЦЕССА.....	78
6.3. ВЫПОЛНЕНИЕ ЭКЗЕМПЛЯРА ПРОЦЕССА .....	78
6.4. ОСТАНОВКА ЭКЗЕМПЛЯРА ПРОЦЕССА .....	79
6.5. ИСПОЛНЕНИЕ ТОЧЕК .....	79
<b>7. МОНИТОР ПРОЦЕССОВ.....</b>	<b>79</b>
7.1. ПОИСК И ПРОСМОТР ПРОЦЕССОВ.....	80
7.2. УПРАВЛЕНИЕ ХОДОМ ИСПОЛНЕНИЯ ПРОЦЕССА .....	86
7.3. СРАВНЕНИЕ РЕДАКЦИЙ ОБЪЕКТОВ .....	90
<b>8. СОБЫТИЯ И ОПОВЕЩЕНИЯ .....</b>	<b>91</b>
8.1. ОБЩИЕ ПОНЯТИЯ .....	91
8.2. СПРАВОЧНИК СОБЫТИЙ .....	92
8.3. ПОДПИСКА НА СОБЫТИЯ .....	93
8.4. НАСТРОЙКА МАКЕТА УВЕДОМЛЕНИЯ .....	96
8.5. РЕГЛАМЕНТЫ ДЛЯ РАБОТЫ С СОБЫТИЯМИ И РАССЫЛКА УВЕДОМЛЕНИЙ .....	98
<b>ПРИЛОЖЕНИЕ 1. ПАРАМЕТРЫ НАСТРОЕК APPLICATION.PROPERTIES .....</b>	<b>102</b>

## 1. Общие сведения о системе

**ISIDA BPM** – это программная система управления бизнес-процессами, позволяющая описывать, исполнять и контролировать бизнес-процессы различных видов (управленческие, производственные, логистические и т. п.). Система ISIDA BPM является составной частью Платформы ISIDA (ISIDA Platform).

### 1.1. Объекты и процессы

Ключевые понятия, которыми оперирует ISIDA BPM, – «Бизнес-объект» и «Бизнес-процесс».

**Бизнес-объект**, или просто **Объект** является особым образом организованным объектом данных. Объекту обычно соответствует какая-либо реальная сущность – например, «сотрудник», «подразделение», «заявка», «изделие», «кредитный договор». Объект имеет логическую структуру, то есть тот или иной набор характеристик – **полей**, в которых хранится информация. Например, в объекте «сотрудник» могут быть поля «фамилия», «имя» «год рождения», а в объекте «заявка» – «номер», «дата», «статус» и «содержание».

Объект характеризуется **типом**. Именно тип объекта определяет его логическую структуру (набор полей).

Собственно данные хранятся во множестве **экземпляров объекта**, каждый из которых имеет тот или иной тип. Экземпляры объекта размещаются в базе данных.

*Примечание:* иногда под термином «объект» понимают его тип, а иногда – экземпляр объекта того или иного типа. Обычно, если нет уточнения, «объект» означает «экземпляр объекта», но в каждом случае необходимо учитывать контекст и явные указания на применяемое значение термина.

**Поле**, в свою очередь, тоже всегда имеет тип. Тип поля может быть простым (например, «число», «дата», «строка») либо сложным (в этом случае поле – это тоже объект иного, ранее определенного типа, например: «подразделение», «сотрудник», «контрольное событие»). Поле может быть единичным либо множественным, содержать ссылку на иной объект и т.п.

Объект может быть **собственным**, или «родным», либо **заимствованным**, или «приемным».

«Родной» объект создается с помощью специального визуального инструмента – «**Конструктора объектов**». Для каждого типа объекта задается требуемый набор полей с указанием их типов и иных параметров. После того, как объект сконструирован, он автоматически «маппируется» (преобразуется) в соответствующую структуру базы данных: для этого типа объекта создаются нужные таблицы, индексы, ключи для связывания таблиц и т.п. Экземпляры объекта затем размещаются именно в этой структуре базы данных.

«Приемный» объект принадлежит стороннему приложению, он внешний по отношению к ISIDA BPM и уже существует. Поэтому тип такого объекта только описывается в «Конструкторе объектов» (но не создается): указываются поля объекта и то, каким структурам «чужой» базы данных они соответствуют. При этом могут описываться не все поля объекта, а только те, которые нужны для работы. Таким образом, если для «родного» объекта логическая структура создается с помощью «Конструктора объектов» и затем автоматически преобразуется в соответствующую структуру базы данных, то для «приемного» объекта и его логическая структура, и соответствующая структура базы данных уже существуют, и в «Конструкторе объектов» лишь указывается, какие поля можно использовать, и в каких элементах структуры базы данных эти поля размещаются. «Приемные» объекты используются, если организуется взаимодействие либо совместная работа с независимо разработанными программными приложениями (чтобы использовать их данные), а также в некоторых специальных ситуациях – например, когда объект имеет очень сложную логическую структуру либо особую схему хранения в базе данных, и такой объект определяется не в Конструкторе объектов, а, например, низкоуровневыми системными средствами.

Другим ключевым понятием является Бизнес-процесс, или просто Процесс.

**Бизнес-процесс** – это совокупность (поток) взаимосвязанных действий (операций процесса), последовательность выполнения которых направлена на достижение установленного результата.

ISIDA BPM, являясь специализированной программной системой класса BPMS («системы управления бизнес-процессами»), отвечает за описание и выполнение бизнес-процессов.

В ISIDA BPM принято, что бизнес-процесс всегда строится для какого-либо бизнес-объекта, а выполнение бизнес-процесса понимается как «движение» этого объекта по «**маршруту**», зависящему от логики выполнения процесса. Такой объект называется «**управляемым объектом**».

Как и для объектов, для бизнес-процессов применяются понятия «**описание**» и «**экземпляр**». Описание процесса определяет логику его выполнения. В системе запускаются и выполняются экземпляры процесса, каждый из которых создан на основе того или иного описания.

*Примечание:* как и в случае с объектами, под термином «процесс» – в зависимости от контекста – могут пониматься как «описание процесса», так и «экземпляр процесса». Обычно, если нет уточнения, имеется в виду «экземпляр процесса».

Для конструирования бизнес-процессов (разработки описаний) в составе ISIDA BPM имеется визуальный инструмент – **Дизайнер процессов**. Для описания процесса используется специальная нотация, основанная на стандарте *Business Process Model and Notation (BPMN)*.

Описание создается Дизайнером процессов как диаграмма, похожая на ориентированный граф, состоящий из точек и переходов между ними. В точке процесса может выполняться определенная операция – **задача**, так и проверяться условия для определения дальнейшего движения – в этом случае точка представляет собой **шлюз**.

*Примечание:* в ISIDA BPM иногда вместо термина «задача» используется термин «**процедура**», в рассматриваемом контексте, эти термины являются синонимами.

ISIDA BPM поддерживает описание и исполнение как простых, так и сложных процессов: последовательных, параллельных, комбинированных; ветвящихся по условиям; имеющих возвраты/повторы; включающих подпроцессы и т.п.

Для каждой точки процесса, где предполагается выполнение действий пользователями, устанавливаются **исполнители** – пользователи, которые должны выполнять предусмотренные задачей действия. Поддерживаются различные правила назначения исполнителей: может быть указан конкретный пользователь, несколько пользователей или категория (группа) пользователей, причем с разными условиями выполнения операции в точке. Кроме того, для точки или для процесса в целом можно указать **контролеров** – пользователей, которые получают возможность следить за ходом процесса или его части.

ISIDA BPM допускает наличие точек процесса, в которых действия выполняются автоматически, без участия исполнителей (людей), подробнее об этом ниже.

Описание процесса используется как шаблон, на основе которого создаются конкретные экземпляры процесса. За создание экземпляров процессов, их запуск и выполнение отвечает центральный компонент ISIDA BPM – **Диспетчер процессов**. Он хранит экземпляры процессов и связанную с их выполнением оперативную информацию в специальной базе данных, обеспечивает «движение» объектов по маршрутам, вызов требуемых задач и т.д.

Еще один компонент ISIDA BPM – **Монитор процессов** – представляет собой специальную визуальную среду, предназначенную для администрирования выполняющихся экземпляров

процессов. Он дает обзор исполняющихся / завершенных / прерванных процессов и позволяет осуществлять «ручное» управление ими: приостановку и возобновление процесса, повторное исполнение и/или пропуск точек маршрута, корректировку сценария процесса в ходе исполнения.

## 1.2. Задачи

В ISIDA BPM **задача** – это специальный программный компонент, который содержит ту или иную бизнес-логику и, в соответствии с ней, выполняет предусмотренный набор действий. Задача **выполняется** в точке процесса.

Задача может быть «пользовательской» или «автоматической». **Пользовательские** задачи имеют экранный интерфейс и выполняют какие-либо действия по вводу, обработке и хранению данных – например, «Согласование заявки на получение кредита риск-менеджером», «Регистрация договора с клиентом», «Подтверждение получения комплекта материалов». **Автоматические** задачи – это программные роботы, которые работают без участия человека – например, «Отправка сообщения клиенту», «Расчет времени производственной операции», «Передача сведений о договоре в бухгалтерскую систему».

Часто задачи (обычно автоматические) являются **интеграционными** и обеспечивают требуемое взаимодействие со смежными корпоративными системами либо внешними системами и сервисами (например, «Получение из реестра информации об объекте недвижимости»).

Аналогично объектам, задачи делятся на две категории: (1) **собственные**, или «родные» задачи (2) **заимствованные**, или «приемные» задачи. «Родные» задачи создаются (разрабатываются) по установленной в ISIDA BPM технологии, развертываются по определенным правилам и автоматически регистрируются в системе. ISIDA BPM вызывает такие задачи в точках процесса в соответствии с его логикой, передавая им значения параметров и информацию о событиях. «Приемные» задачи – это те или иные участки бизнес-логики внешних (по отношению к ISIDA BPM) программных систем, которые формально объявляются как задачи (с определенными именами и параметрами). ISIDA BPM в нужных точках процесса передает управление внешней системе, которая запускает соответствующие участки своей функциональности, после чего сообщает результатах выполнения такой «задачи», и процесс выполняется дальше. Технология «приемных» задач позволяет использовать возможности существующих информационных систем с минимальным изменением последних.

Для того, чтобы задачи могли использоваться, они должны быть зарегистрированы в системе. Для этого в составе ISIDA BPM имеется специальный компонент – **Менеджер задач**. Он ведет каталог используемых задач и их параметров. Задачи могут регистрироваться как вручную, так и автоматически – через специальный программный интерфейс.

Дизайнер процессов позволяет в ходе конструирования процесса определять, какие задачи выполняются в его точках, и устанавливать различные параметры и условия их выполнения.

Как уже отмечалось выше, задача – как программный компонент – должна разрабатываться с соблюдением определенных требований. Однако разработка программного компонента – это сложный и дорогой процесс, поэтому создавать новые задачи целесообразно для особых и очень специализированных случаев. В большинстве же «обычных» ситуаций достаточно использовать имеющиеся в составе ISIDA BPM **готовые задачи**. Они обладают широкими возможностями настройки и удовлетворяют самым разным прикладным потребностям. Тогда никакой разработки вести не надо, следует только указать, что в точках процесса выполняются – с требуемыми параметрами и условиями – те или иные готовые задачи.

Одна из них так и называется – «**Универсальная задача**» («Generic Task»). В терминологии ISIDA BPM, это «родная» пользовательская задача. У «Generic Task» очень широкая область применения: ввод и отображение полей объекта, выполнение различных команд над значениями полей объекта и

параметров процесса, требуемое движение по маршруту – «вперед», «назад» или «без движения» – с выполнением необходимых действий по управлению данными. При этом визуальная часть «Универсальной задачи» (экранная форма) конструируется специальным визуальным средством – Конструктором форм (об этом будет рассказано ниже).

Другая готовая задача, входящая в состав ISIDA BPM – *«Изменение полей объекта»* («ChangeObjectFields»). В терминологии ISIDA BPM, это «родная» автоматическая задача. Она, в соответствии с заданными параметрами и условиями, автоматически изменяет значения тех или иных полей объекта при его прохождении данной точки маршрута.

Библиотека готовых задач постоянно пополняется по мере развития ISIDA BPM, расширяя функциональные возможности системы.

### 1.3. Приложения

Описанные выше программные инструменты сами по себе не обеспечивают получение какого-либо полезного результата. Целью всегда является решение какой-либо конкретной бизнес-задачи в той или иной предметной области. Например, в банке необходимо средствами системы ISIDA BPM автоматизировать процесс формирования и обработки кредитной заявки, включающий не только «движение» заявки по установленному маршруту согласования в службах банка и утверждения кредитным комитетом, но и возможности поиска заявок по различным критериям сотрудниками банка с учетом прав доступа, отображение списков найденных заявок и просмотр выбранной заявки, ведение справочников, формирование различных отчетов и т. п. Эта цель достигается созданием и использованием соответствующей прикладной программы, или *программного приложения*. Сотрудники банка работают в его среде: авторизуются в системе по паролю, получают кредитные заявки, пришедшие к ним для согласования, пользуются возможностями поиска информации, получают отчеты и т.п.

Программное приложение системы ISIDA BPM, или, для краткости, просто *«приложение»* («App») создается (разрабатывается) и затем функционирует в среде системы, решая ту или иную прикладную задачу.

Результатом разработки приложения является *«прикладная конфигурация»*. Это специальным образом организованный комплект описаний (дескрипторов) – файловый архив определенной структуры, который представляет собой *дистрибутив приложения*. Он содержит, как правило, дескрипторы типов объектов (управляемых, справочных, служебных) и описания бизнес-процессов, пользовательских меню, экранных форм, форм отчетов и др. Например, для описанного выше приложения для обработки кредитной заявки в комплекте могут быть дескрипторы объекта «Кредитная заявка», «Банковский продукт», «Клиент банка»; дескрипторы бизнес-процессов «Обработка кредитной заявки (физические лица)» и «Обработка кредитной заявки (юридические лица)» с описанием как логики процессов, так и параметров выполняемых в точках процессов задач («Универсальной задачи» и задачи «Изменение полей объекта»); дескрипторы различных экранных форм для создания, поиска и изменения объектов; дескрипторы аналитических отчетов по кредитным заявкам; описание состава меню для разных групп пользователей.

Для того, чтобы создать (разработать) приложение, используются различные инструменты системы ISIDA BPM и Платформы ISIDA в целом.

*Конструктор объектов*, который был описан ранее, предназначен для создания типов объектов, в том числе «управляемых объектов», для которых выполняются бизнес-процессы, а также иных категорий объектов (справочных, служебных, учетных и т. п.).

**Дизайнер процессов**, также описанный выше, служит для описания бизнес-процессов: конструирования процесса, логики движения объекта по маршруту, указания задач, которые выполняются в точках процесса, определения параметров и условий их выполнения и др.

За взаимодействие с пользователем через WEB-браузер отвечает клиентская (фронтальная) часть Среды выполнения – **Рабочее пространство**. В Рабочем пространстве пользователь выбирает доступные функции из меню, вводит и просматривает информацию в экранных формах, получает уведомления и отчеты и т. п.

*Примечание.* В текущей версии ISIDA BPM поддерживается работа пользователя в WEB-браузере настольного персонального компьютера, ноутбука или планшетного компьютера.

Серверная часть Среды исполнения состоит из нескольких компонентов.

За выполнение базовых операций с объектами – создание, поиск и получение, чтение и модификацию – отвечает **Процессор объектов**.

**Генератор форм** отображает экранные формы в соответствии с их дескрипторами.

В состав Среды исполнения входят и некоторые другие компоненты, которые выполняют различные функции обработки и хранения данных.

Среда исполнения при запуске приложения и в ходе его работы взаимодействует с компонентами системы ISIDA BPM, описанными выше (Диспетчером процессов, Монитором процессов), а также с иными составными частями Платформы ISIDA: **ISIDA Admin** (центром администрирования), **ISIDA Depot** (средой хранения электронного контента) и др.

Разработку приложений могут выполнять специалисты разной компетенции, например, системные аналитики, технологи, специалисты по оптимизации процессов, инженеры и т.п. В данном документе будет использоваться обобщенный термин **Администратор**, однако следует понимать, что это не традиционный администратор, отвечающий за функционирование системы, а специалист по управлению автоматизированными бизнес-процессами.

#### 1.4. Работа с электронными документами

ISIDA BPM поддерживает специальный тип управляемого объекта – **электронный документ**. Он имеет другую структуру, описывается иными средствами, но правила разработки и выполнения процессов в целом соответствует практике работы с объектами.

Для электронных документов используется своя система учета типов (описана в отдельной главе настоящего документа). Вместо полей, которые есть у объектов, для электронных документов используется понятие реквизитов, которые могут принадлежать только простым строковым или числовым типам.

Общую часть электронного документа составляет набор файлов произвольного формата, а также набор предусмотренных реквизитов. Особую часть документа составляет набор электронных цифровых подписей. Хранение всей информации о документе выполняется в специальном контейнере согласно внутреннему формату, основанному на языке XML. Для хранения документов, вне зависимости от их типов, используется встроенное хранилище электронных документов или выполняется интеграция с внешним хранилищем (системой класса Enterprise Content Management, ECM).

ISIDA BPM обеспечивает полный цикл обработки документа: создание, обращение, передачу на хранение. В ходе выполнения бизнес-процессов электронный документ проходит все этапы согласования, подписания и исполнения. Для фиксации решений, принятых уполномоченными

лицами, применяется технология электронной цифровой подписи, которая может работать в разных режимах: от простой электронной подписи (авторизованная отметка уполномоченного лица) до криптографических программных средств выработки и проверки электронной подписи, включая интеграцию с различными удостоверяющими центрами. Таким образом с помощью ISIDA BPM можно строить системы юридически значимого электронного документооборота.

В процессе обработки электронного документа могут быть предусмотрены автоматические точки маршрута, которые позволяют передать информацию о документе внешним информационным системам или запросить определенные сервисы, например, зарегистрировать документ в системе электронного делопроизводства или передать на хранение в архив организации.

При создании нового бизнес-процесса следует выбрать, что будет двигаться по маршруту – объект или электронный документ. Оба подхода могут применяться параллельно, каждый из них имеет свои преимущества и недостатки, выбор зависит от решаемой задачи и ограничений. Рекомендуется рассмотреть следующие аргументы для принятия решения:

- 1) Электронный документ более простой по своей структуре (файлы, реквизиты, подписи), а значит и более универсальный. Такую структуру может понять любая автоматизированная система, а значит документ может быть экспортирован из ISIDA BPM, преобразован к иному формату хранения, отправлен по некоторой системе доставки во внешнее окружение и т.п. Чтобы экспортировать структурированный объект, с каждой системой предстоит отдельно согласовывать его структуру. Поэтому при взаимодействии с большим числом внешних систем проще использовать документ.
- 2) Электронный документ – самый логичный ответ на необходимость авторизации данных при помощи простой или криптографической подписи (придание информации юридической значимости). Если отметок о прохождении в маршруте недостаточно и требуется более сильное средство авторизации и контроля целостности, то однозначно лучше использовать электронный документ.
- 3) Если результаты бизнес-процесса так или иначе должны быть переданы в архив организации и храниться там определенный срок, то для этого лучше использовать электронный документ.
- 4) Электронные документы обладают большей консервативностью. Их реквизиты как правило описывают содержимое, поэтому вторичны. Если задача предполагает, что данные должны легко модифицироваться в ходе процесса и всегда быть актуальными, лучше использовать объект.
- 5) Реквизиты документа относятся к простым типам (строки и числа), поэтому их сложно использовать для описания сложных структур данных. Для заполнения можно использовать справочники, выпадающие списки, но в итоге при создании электронного документа они все равно упростятся до строк и чисел (вместо полного справочного объекта в реквизиты будет занесено только его строковое наименование и т.д.). Это естественное продолжение свойств документа как простого универсального объекта, который может понять другая внешняя система или человек вне нашей системы. В задачах, где нужна аналитика, расчеты, высокая структурированность данных, лучше использовать объекты.
- 6) Процесс, связанный с документом, как правило, ограничивается жизненным циклом самого документа, т.е. он обычно короче, чем процесс у объекта. Например, естественными фазами документального процесса является его согласование, рассмотрение, исполнение. Если прикладная задача решается несколькими документами, то большой процесс разобьется на отдельные процессы по каждому документу, в то время как при работе с объектом можно построить один сквозной процесс. Можно сказать, что процессы для объектов более полные, непрерывные, охватывающие все стадии обработки и больше соответствуют духу BPM-систем.



7) В тех случаях, когда у объекта появляются поля с типом «большой текст», это повод задуматься, о применении электронного документа. Текст привычнее и удобнее писать внутри документа. Проблема не в том, что большие фрагменты текста хранить в базе данных нелогично, современные СУБД легко справляются с этой задачей. Гораздо сложнее предоставить адекватный интерфейс для написания и чтения такого текста. Написать пару абзацев можно в любом поле объекта, а написать и прочитать текст на несколько страниц гораздо удобнее в файле формата Microsoft Word. Конечно, можно допустить прикрепление к объекту произвольных файлов и весь текст будет там. Однако к произвольным файлам невозможно выдвинуть требования по их оформлению, содержанию, как правило, это слабо структурированные, часто хаотичные материалы. Документ в этом отношении строже, на него действуют нормы внешнего вида, законодательные требования и в конце концов шаблоны и образцы. Поэтому при работе с текстом лучше использовать электронный документ.

Нет однозначного решения, какой управляемый объект лучше использовать на маршруте – документ или объект. Выбор предстоит делать каждый раз, учитывая контекст задачи. Но ISIDA BPM поддерживает оба варианта работы.

## 2. Конструктор объектов

### 2.1. Общие сведения

Конструктор объектов предназначен как для создания «родных», так и для адаптации «приемных» объектов внешних систем. Следует отметить, что в качестве управляемых объектов могут выступать только «родные» объекты системы. После создания его структуры ISIDA BPM предоставляет пользователям возможность создать экземпляр такого объекта и продвигать его по маршруту бизнес-процесса.

Для каждого объекта администратор разрабатывает его структуру: поля и их идентификаторы, размерность, тип, признаки множественности, обязательности и т.д. Такой процесс называется *созданием объекта*.

Поля объекта могут быть простыми, например, строковыми, числовыми, датой, либо сложными: большой текст, вложенный объект, выпадающий список. На стадии конструирования объект имеет статус «Черновик», что позволяет изменять его структуру на протяжении продолжительного времени и не допускать использования, пока объект не будет полностью готов.

Следует помнить, что каждый идентификатор поля – это будущая колонка в таблице, а идентификатор объекта – сама таблица, поэтому к их именованию надо подходить осмысленно: использовать английский язык, давать полям понятные идентификаторы, отражающие назначение конкретного поля. Например, *requestType* – тип заявки, *responsibleExecutor* – ответственный исполнитель, *executingTerm* – срок исполнения и т.п. Хорошим тоном считается написание первого слова идентификатора с прописной буквы, а второго и последующих слов – с прописной. Все идентификаторы должны быть уникальными в рамках одного объекта.

Если объект предусматривает в качестве поля не просто строку или дату, а другой объект, то сначала должен быть разработан и опубликован вложенный объект, а уже затем – основной. Например, объект «Заявка» имеет поле «Информационная система», которое само по себе является отдельным сложным справочным объектом. В этом случае сначала необходимо сконструировать и опубликовать структуру объекта «Информационная система», а уже затем его можно использовать в качестве поля объекта «Заявка». В свою очередь, объект «Информационная система» также может в себе содержать сложное поле-объект, например, «Подразделение», или «Сотрудник». Таким образом ISIDA BPM поддерживает многоуровневую структуру сложных объектов.

После того, как объект сконструирован и сохранен, его следует опубликовать. В процессе публикации он автоматически «мапируется» (преобразуется) в соответствующую структуру базы данных: для

этого типа объекта создаются нужные таблицы, индексы, связи между таблицами и т.п. Экземпляры объекта затем размещаются именно в этой структуре базы данных. После публикации изменить объект будет не так просто: поскольку в базе данных уже заведены таблицы и колонки, изменить идентификаторы, размерности, связи будет невозможно. Для изменения структуры объекта его следует снова перевести в статус «Черновик» (отменить публикацию), и затем вносить изменения. После сохранения объект следует снова опубликовать.

После того, как структура управляемого объекта сконструирована и объект опубликован, им можно пользоваться: создавать для него бизнес-процесс, настраивать задачи процесса, которые, например, влияют или изменяют поля этого объекта. Появление нового управляемого объекта в системе (заявки, акта, события операционного риска, отчета) автоматически предполагает создание для него визуальной формы для заполнения полей объекта, просмотра объекта, его редактирования. В текущей версии функциональность ISIDA BPM не позволяет самостоятельно администратору системы создавать визуальную среду (формы) для создания, редактирования и просмотра управляемых объектов. Такие формы создаются командой разработчиков ISIDA BPM и поставляются в дистрибутивах ISIDA BPM. Однако в перспективе в состав ISIDA BPM будет оснащен специальным компонентом («Дизайнером форм»), который позволит создавать и настраивать форму без применения программирования.

Сказанное выше относится также и к справочным объектам, которые являются особой категорией «родных» объектов. Такие объекты создаются для накопления справочных данных, например, справочник информационных систем, справочник видов документов и т.п. Конструирование объекта и его публикация не означает автоматическое наполнение такого справочника данным – для этого нужна визуальная форма создания записи и просмотра всего справочника, а также возможность встроить пункт меню для работы с ним в главное меню системы. Пока такие формы создаются командой разработчиков.

Конструктор объектов предназначен также для адаптации «приемных» объектов, то есть описания средствами ISIDA BPM объекта, который принадлежит другой системе. Объект и его структура уже существует, поэтому тип такого объекта только описывается в конструкторе объектов (но не создается): указываются поля объекта и то, каким структурам «чужой» базы данных они соответствуют. При этом могут описываться не все поля объекта, а только те, которые нужны для работы. Этот процесс называется *адаптацией объекта*. Для выполнения этой работы ISIDA BPM должен обладать доступом к указанным таблицам внешней системы (настраивается средствами СУБД).

Например, некоторая внешняя система уже имеет в своей структуре справочник сотрудников и соответствующий ей объект в базе данных таблицу *employees*. В этом случае в системе ISIDA BPM не нужно создавать свой объект «Сотрудники», а воспользоваться готовой структурой *employees*, которая ведется во внешней системе. Для этого внешней системе нужно предоставить ISIDA BPM доступ на чтение таблицы *employees*, а затем в Конструкторе объектов администратор на базе имеющихся данных создает описание «приемного объекта», т.е. выполняет его адаптацию.

Важным функционалом ISIDA BPM является *импорт/экспорт объектов*. Довольно часть структура объекта разрабатывается на одном экземпляре ISIDA BPM, например, в тестовой зоне, а потом быстро и без искажений должна быть перенесена на другой экземпляр, например, в зону промышленной эксплуатации. Для этого миграции структуры объекта используется функция экспорта/импорта. Такой способ позволяет избежать ошибок при многократном создании с нуля одного и того же объекта, сэкономить время и гарантировать идентичность одних и тех же объектов на разных экземплярах систем.

В качестве формата для миграции данных используется файл формата JSON. Соответственно с помощью функции экспорта происходит выгрузка структуры объекта в JSON-файл, после чего с помощью функции импорта эта информация загружается в другой экземпляр ISIDA BPM.

## 2.2. Создание объектов

Рассмотрим основные возможности сеанса создания объектов (рис.2.2-1).

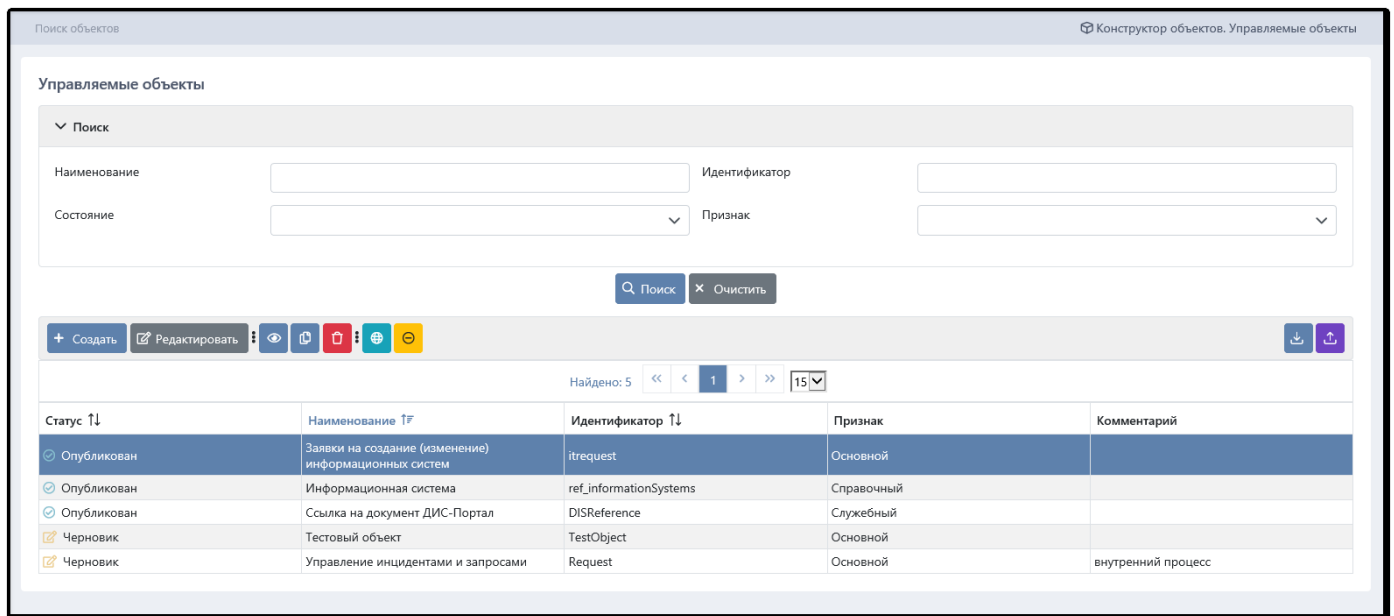


Рисунок 2.2-1 – Сеанс создания объектов

Перед описанием основных функций по созданию объектов необходимо пояснить некоторые термины и обозначения.

Каждый объект характеризуется идентификатором и наименованием, которые администратор придумывает самостоятельно. Уникальность идентификатора контролируется системой.

По умолчанию объект создается в статусе «**Черновик**». Это означает, что он находится в режиме активного редактирования: состав и наименование его полей, типы данных и прочие характеристики могут измениться. Объекты в таком статусе еще недоступны в Дизайнере процессов. Когда работа с объектом завершена, он публикуется, т.е. переходит в статус «**Опубликован**». Опубликованные объекты становятся доступны для описания и запуска бизнес-процессов.

В момент публикации в базе данных автоматически создаются соответствующие таблицы, индексы и прочие элементы, которые обеспечивают хранение экземпляров объектов, структура объекта «кристаллизуется» и объект уже не может так свободно менять свои характеристики, поэтому для опубликованных объектов многие операции редактирования заблокированы (например, нельзя менять идентификаторы и типы данных полей). В то же время некоторые операции доступны и для опубликованных объектов, например, в объект можно добавить новое поле или изменить наименование имеющихся полей, нельзя изменить идентификатор самого объекта. Чтобы внесенные изменения вступили в действие, объект нужно заново опубликовать.

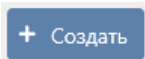
Если экземпляры объекта еще не созданы, публикацию объекта можно отменить. В этом случае объект вернется в статус «Черновик», все его характеристики могут быть изменены.

Создаваемым объектам присваивается определенный признак, который указывает на принадлежность объекта к одной из перечисленных групп:

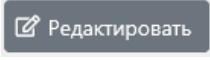
- **основной** – объекты с таким признаком могут использоваться в качестве управляемых объектов, например: заявка, кредитный договор, событие операционного риска и т.д., именно они движутся по маршруту;
- **служебный** – объект, который является частью других объектов и самостоятельно не используется, например, ссылка на документ внешней системы;

- **справочный** – объект, который используется для справочных полей, например, информационная система, вид документа.

Таким образом для создания объектов в ISIDA BPM предусмотрены следующие функции:



- создание нового объекта;



- редактирование существующего объекта;



- группа кнопок для просмотра, копирования и удаления объекта;



- публикация черновика. Объект принимает статус «Опубликован» и может быть использован в дальнейшей работе;



- отмена публикации. Объект принимает статус «Черновик» и в дальнейшем может быть отредактирован или удален;



- экспорт структуры объекта в файл формата JSON.



- импорт структуры объекта из формата JSON.

**Внимание:** Импортированный объект в целевой системе будет находиться в статусе черновика, поэтому для завершения процедуры миграции объект необходимо опубликовать.

С ходом эксплуатации количество типов объектов может существенно вырасти, поэтому для **фильтрации** списка предусмотрены специальные поля и функция поиска (рис. 2.2-2).

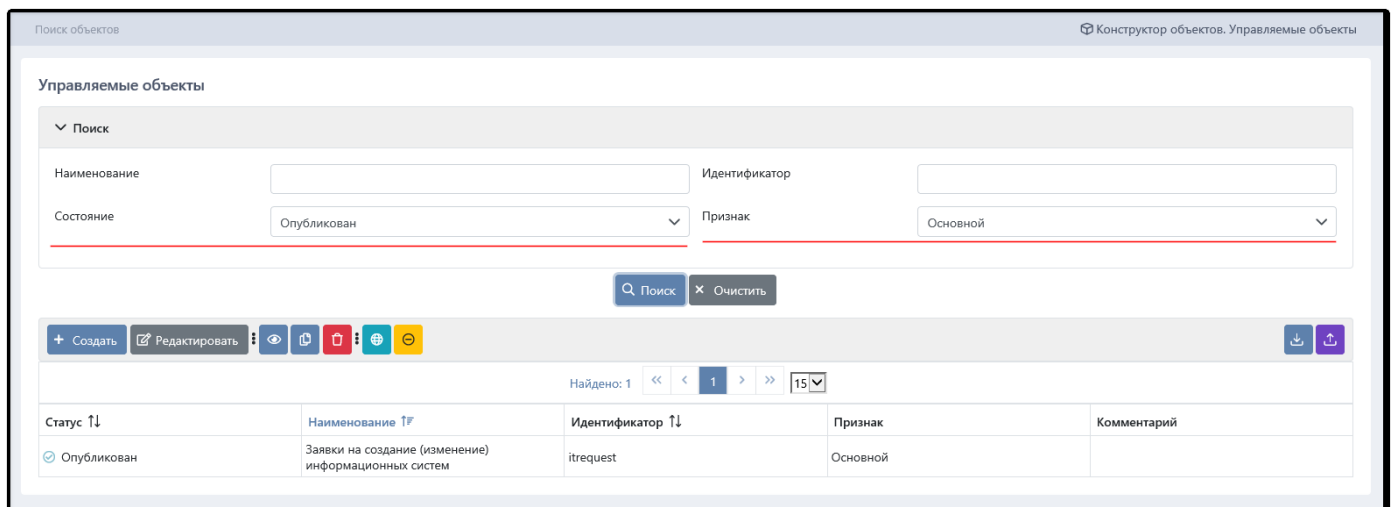


Рисунок 2.2-2. Фильтрация списка объектов

В полях «Наименование» и «Идентификатор» достаточно указать лишь часть значения, так как поиск работает по принципу «содержит». Для полей «Состояние» и «Признак» используются выпадающие списки с перечнем доступных значений.


При описании объекта три характеристики являются обязательными (рис. 2.2-3):

- **Наименование:** понятное участникам процесса обозначение объекта.
- **Идентификатор:** системная характеристика, которая задает таблицу в базе данных, а также используется в других компонентах, но на пользовательский уровень не выводится (тем не менее рекомендуется давать понятное и краткое значение). ISIDA BPM контролирует уникальность идентификаторов всех типов объектов;
- **Признак** – основной / служебный / справочный.

Поле «Комментарий» необязательно для заполнения и может использоваться администратором по своему усмотрению.

При сохранении объекта или поля объекта реализована проверка на **длину** наименований таблиц и колонок. Если длина превышает заданное администратором количество символов, то система выдает предупреждение и не сохраняет объект или поле. За длину наименований в *application.properties* отвечает настройка: *studio.sql.names.maxlength=30*.

Рисунок 2.2-3 – Создание нового объекта

Для управления полями объекта предусмотрен набор стандартных операций: добавить / отредактировать / удалить - .

Для описания поля необходимо указать следующие характеристики (рис. 2.2-4):

- **Наименование:** понятное участникам процесса обозначение поля.
- **Идентификатор:** системный идентификатор поля, который будет использован для идентификатора колонки в таблице.
- **Тип:** характеристика, указывающая на содержимое поля и способ его заполнения (подробное описано ниже).
- **Длина:** количество символов для строковых полей;
- **Является множественным:** флаг, указывающий на то, допускает ли данное поле несколько значений, например: простое строковое поле «Статус» не может иметь множество значений, поэтому данный флаг для него будет снят, а строковое поле «Характеристики договора» может допускать множество значений, поэтому для него флаг будет установлен.
- **Является обязательным:** флаг, указывающий на то, что заполнение данного поля будет обязательным при создании экземпляра объекта и должно контролироваться системой.
- **Значение по умолчанию:** при создании экземпляра объекта реквизит будет заполнен указанным заранее значением.

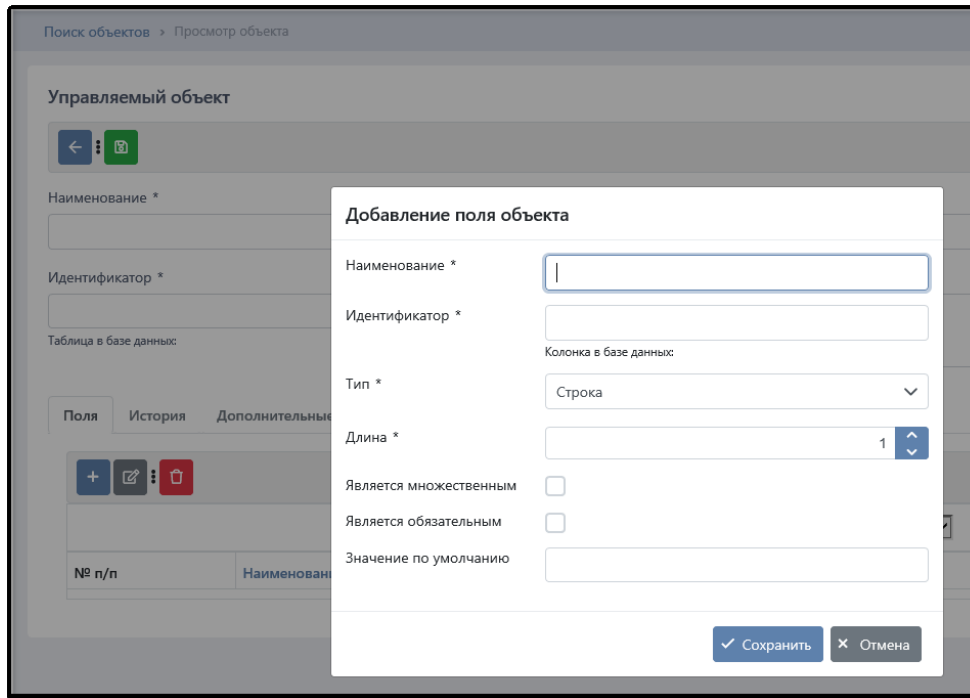


Рисунок 2.2-4 – Добавление поля объекта

ISIDA BPM поддерживает различные типы полей, чтобы обеспечить все многообразие данных, описывающих объект. При выборе определенного типа поля система может запросить дополнительные характеристики.

### Простые типы

Тип **«Строка»** предназначен для хранения строковых значений небольшой длины (длина строки указывается дополнительным параметром). Для заполнения этих полей системой будет предоставлен простой однострочный компонент. Максимальная длина строкового поля может быть равна 1000 символов.

Тип **«Большой текст»** предназначен для хранения больших текстовых фрагментов. Для заполнения поля будет предложен многострочный компонент неограниченной длины.

Тип **«Целое число»** предназначен для хранения целочисленных значений (без дробной части). В дополнительном параметре «Длина» указывать максимальное количество знаков.

Тип **«Дробное число»** предназначен для хранения дробных числовых значений. Для него предусмотрены дополнительные параметры «Длина» – максимальное количество знаков целой части, «Количество знаков после запятой» – размер дробной части.

Тип **«Дата и время»** предназначен для одновременного хранения даты и времени точно до миллисекунд.

Тип **«Дата»** предназначен для хранения календарного дня.

Тип **«Время»** предназначен для хранения времени с точностью до миллисекунд.

Тип **«Логический»** предназначен для хранения значений, которые могут принимать только два значения «Да» (true) или «Нет» (false). Для заполнения таких полей системой будет предложен чекбокс («шпичка»).

Тип **«Выпадающий список»** предназначен для хранения строковых значений, которые не допускают свободный ввод данных, а могут быть выбраны только из предложенного списка. Допустимые

варианты значений указываются в дополнительном параметре «Элементы списка» (необходимо ввести значение и нажать Enter).

### Сложные типы

К этой категории относятся все служебные, справочные и «приемные» объекты, которые уже имеются в системе. Любое из этих объектов может быть использован в качестве типа для поля другого объекта. Сложные типы также могут быть множественными и обязательными для заполнения.

ISIDA BPM предлагает несколько стандартных типов данных этой категории.

«**Подразделение**» используется для полей, которые используют значения из справочника подразделений.

«**Сотрудник**» используется для полей, которые используют значения из справочника сотрудников.

«**Двоичные данные**». Такой тип может применяться для хранения файлов произвольного формата, которые прикрепляются к объектам.

На форме описания объекта предусмотрена специальная закладка «**История**», которая содержит настройки для ведения редакций объекта (рис.2.2-5).

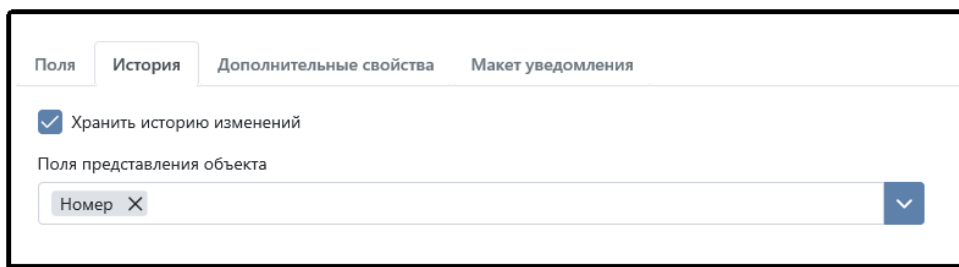


Рисунок 2.2-5 – Настройки для ведения истории объекта

Опция «**Хранить историю изменений**» указывает на необходимость сохранять все редакции объекта. Для каждой редакции фиксируется дата и время её создания, а также автор редакции. Две любые редакции можно сравнить между собой.

**Поля представления объекта** – те поля, объекта которые отображаются в сеансе сравнения редакций для идентификации объекта (рис.2.2-6).

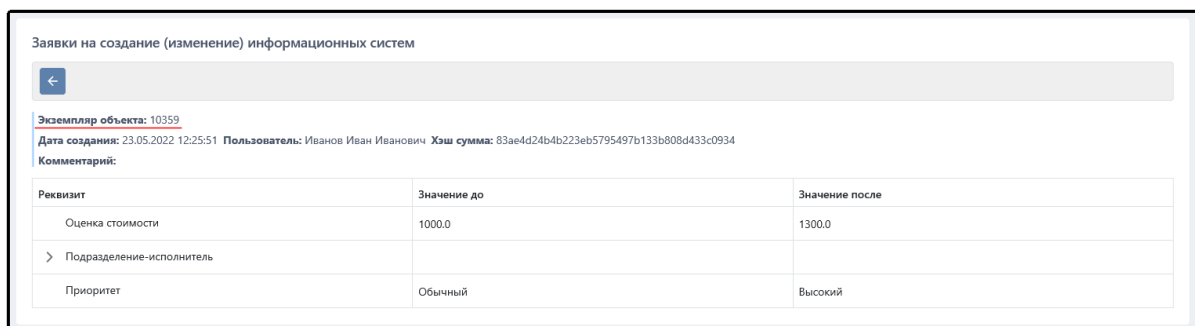


Рисунок 2.2-6 – Сеанс сравнения редакций

## 2.3. Адаптация объектов

Для «приемных» объектов предусмотрены стандартные функции создания, редактирования, просмотра, а также публикации и отмены публикации. Если список «приемных» объектов достаточно большой, его можно отфильтровать с помощью поисковых критериев (кнопка «Искать», рис. 2.3-1).

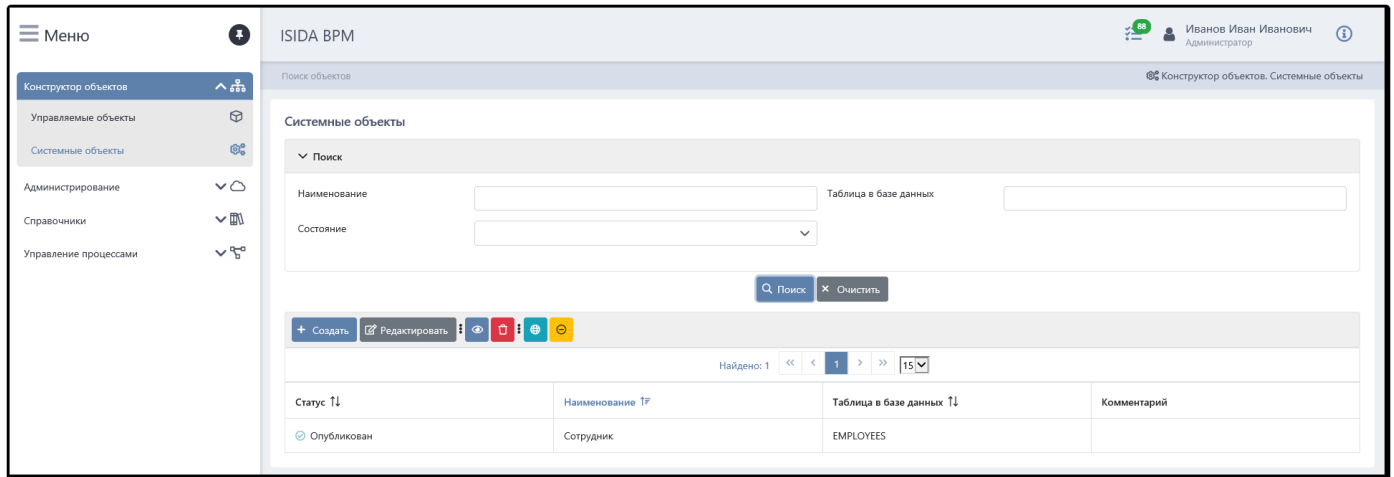


Рисунок 2.3-1 – Сеанс адаптации объектов

Для адаптации объекта соответствующие таблицы внешней системы должны быть доступны в рамках источника данных ISIDA BPM (data source). Процедура адаптации объекта начинается с выбора таблицы, после чего для описания полей будут предложены имеющиеся колонки (рис. 2.3-2). В отличие от «родных» объектов, в «приемных» можно изменить только наименование полей, т.к. все остальные настройки (тип поля, размерность, идентификатор) предопределены в БД.

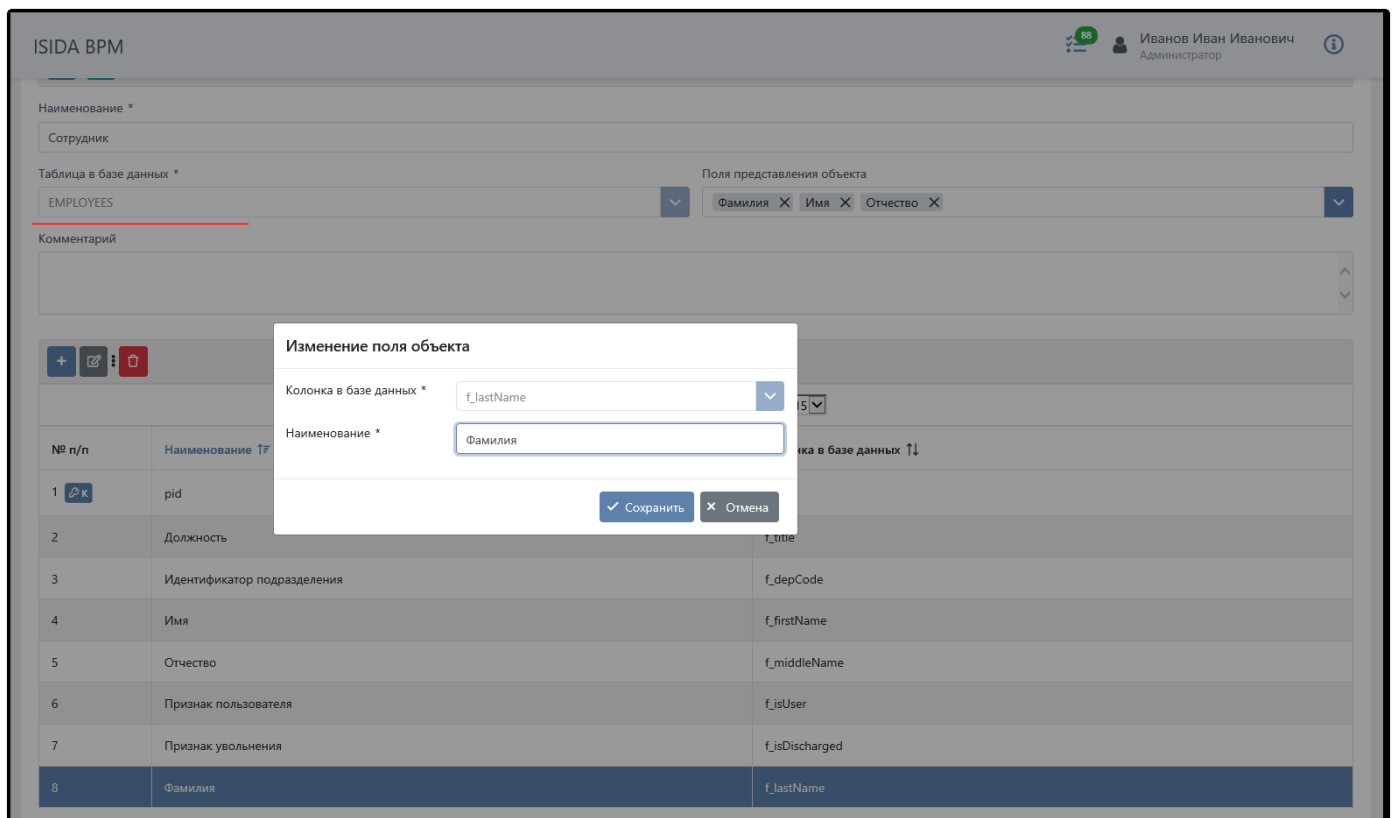


Рисунок 2.3-2 – Настройка полей для системных объектов

## 2.4. Макет уведомления

Закладка «Макет уведомления» служит для настройки шаблона для дальнейшего использования его в макете письма. Макеты уведомлений для событий ISIDA BPM настраиваются в два этапа. Сначала настраиваются шаблоны для темы и содержания письма, которые обычно содержат реквизиты объекта (рис.2.4-1). Заготовка шаблона для темы письма настраивается в секции *object.email.subject*, заготовка шаблона для содержания письма – в секции *object.email.body*.



Рисунок 2.4-1. Настройка макета уведомления

В заголовке письма можно указать, например, такой текст:

`${object.typeName} № ${object.number} от ${object.date|dd.MM.yyyy}`

Тогда в заголовке почтового уведомления будет сформирована такая фраза:  
*Заявки на создание (изменение) информационных систем № 40405 от 01.07.2022*

В заготовке для тела письма можно указать такой текст:

**Номер:** `${object.number}`  
**Дата:** `${object.date|dd.MM.yyyy}`  
**Содержание:** `${object.contents}`  
**Тип заявки:** `${object.requestType}`  
**Приоритет:** `${object.priority}`  
**Статус:** `${object.status}`  
**Подразделение:** `${object.department.name}`  
**Информационная система:** `${object.infSystem.fullName}`

Тогда в содержании почтового уведомления система перечислит все указанные реквизиты и их значения, если они не пусты. При создании шаблона можно пользоваться [справочником допустимых подстановок](#).

Настройка макета уведомления описана [в разделе 8.3](#) данного руководства.

## 3. Электронные документы

### 3.1. Общие сведения

Система ISIDA BPM позволяет использовать электронные документы (ЭД) в качестве управляемых объектов. Для того, чтобы подготовить такую технологию к работе, необходимо выполнить следующие действия:

- создать перечень необходимых реквизитов (или убедиться, что имеющихся реквизитов достаточно);
- создать новый тип электронного документа путем выбора реквизитов из готового перечня;
- настроить представление типа ЭД, в частности: установить на реквизите свойство, будет ли он включен в формы просмотра, редактирования и поиска документа, установить тип и длину реквизита и т.д.;

- создать новый процесс с указанием настроенного типа электронного документа (особенности разработки процесса для электронных документов описаны в [разделе 5.9](#) данного руководства).

**Примечание:** Для начала работы с электронными документами в конфигурационном файле *application.properties* необходимо указать настройку:

*bpm.flowcore.dispatcher.IProcessObjectService.eDocumentProcess=true*

### 3.2. Создание и настройка типа электронного документа

Любой тип электронного документа описывается набором реквизитов. Их состав у каждого типа индивидуален, но при этом все реквизиты всех типов вносятся в особый перечень (пункт меню «Управление реквизитами электронных документов», рис. 3.2-1).

Как и в случае с полями объектов идентификаторы реквизитам следует давать осмысленно, используя латинский алфавит (система контролирует их уникальность), наименование должно максимально полно отражать назначение реквизита.

РЕКВИЗИТЫ ДОКУМЕНТОВ						
Найдено реквизитов: 13						
Индекс	Идентификатор	Наименование	Тип	Длина	Кол-во десятичных знаков	Обязательность
4	Contents	Краткое содержание	строка	1000	1	Нет
5	Date	Дата	строка	10	1	Нет
11	department	Подразделение-заказчик	строка	500	0	Нет
3	depCode	Код подразделения	число	19	0	Да
1	docStartDate	Дата документа (дата начала периода)	строка	10	0	Да
0	eDocTypeID	Тип ЭД	строка	30	0	Да
9	infSystem	Информационная система	строка	500	0	Нет
2	lastVersion	Признак последней версии	строка	5	0	Да
6	Number	Номер	строка	10	0	Нет
7	priority	Приоритет	строка	50	0	Нет
8	requestType	Тип заявки	строка	30	0	Нет
10	responsibleEmployee	Ответственный работник	строка	100	0	Нет
12	status	Статус	строка	50	0	Нет

Рисунок 3.2-1 – Перечень реквизитов

Для каждого реквизита указывается тип его данных (строка или число), предельная длина, а для числовых реквизитов количество знаков после запятой (рис. 3.2-2).

РЕКВИЗИТЫ ДОКУМЕНТОВ						
Найдено реквизитов: 13						
Индекс	Идентификатор	Наименование	Тип	Длина	Кол-во десятичных знаков	Обязательность
4	Contents	Краткое содержание	строка	1000	1	Нет
5	Date	Дата	строка	10	1	Нет
11	department	Подразделение-заказчик	строка	500	0	Нет
3	depCode				0	Да
1	docStartDate				0	Да
0	eDocTypeID				0	Да
9	infSystem				0	Нет
2	lastVersion				0	Да
6	Number				0	Нет
7	priority				0	Нет
8	requestType				0	Нет
10	responsibleEmployee	Ответственный работник	строка	100	0	Нет
12	status	Статус	строка	50	0	Нет

**Добавление реквизита документов**

\*Идентификатор:

\*Наименование:

\*Тип:

\*Длина:

\*Кол-во десятичных знаков:

Обязательность в типе документов:  Применяется для вновь создаваемых типов документов!

Рисунок 3.2-2. Создание реквизита

После того, как реквизиты заведены, можно создавать новый тип электронного документа и наполнять его реквизитами (пункт меню «Управление типами электронных документов»).

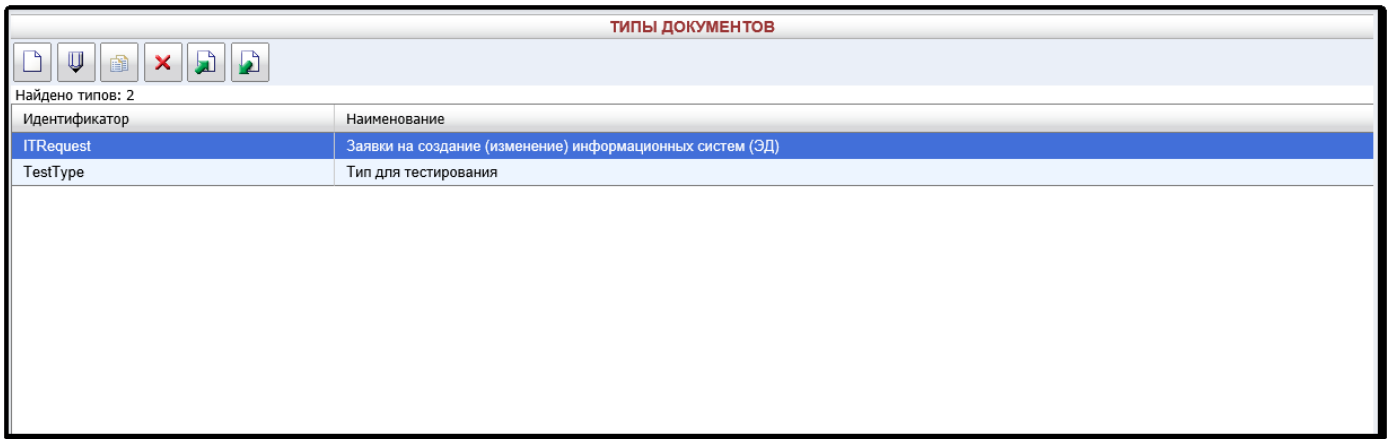


Рисунок 3.2-3 – Создание нового типа ЭД

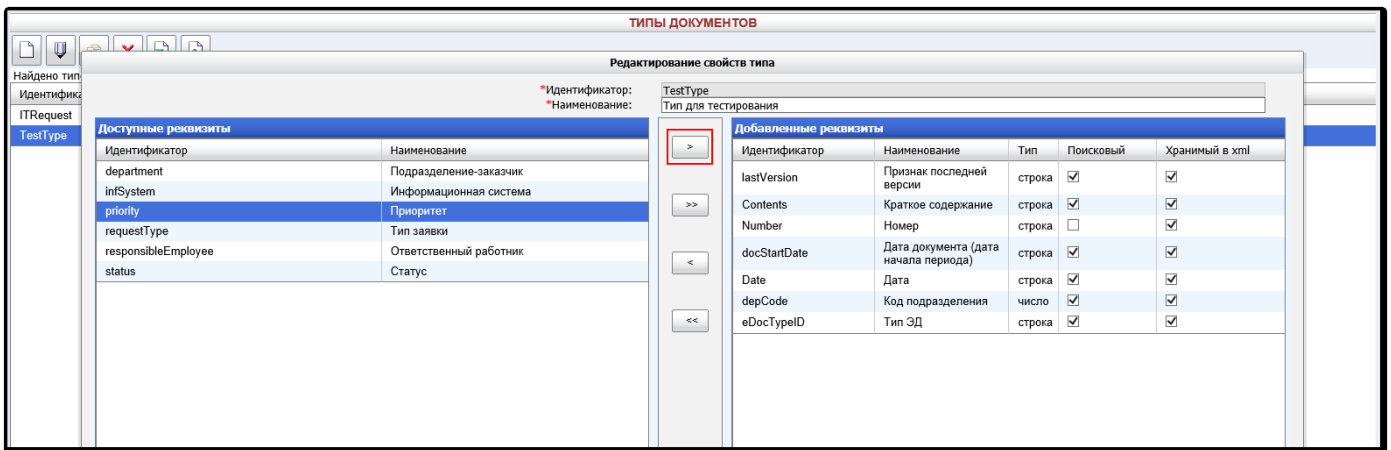


Рисунок 3.2-4 – Наполнение типа ЭД реквизитами

После сохранения типа необходимо настроить представление каждого реквизита (доступность для создания, редактирования, включение в форму просмотра, обязательность при заполнении и т.д.) (пункт меню «Настройка типов электронных документов»).

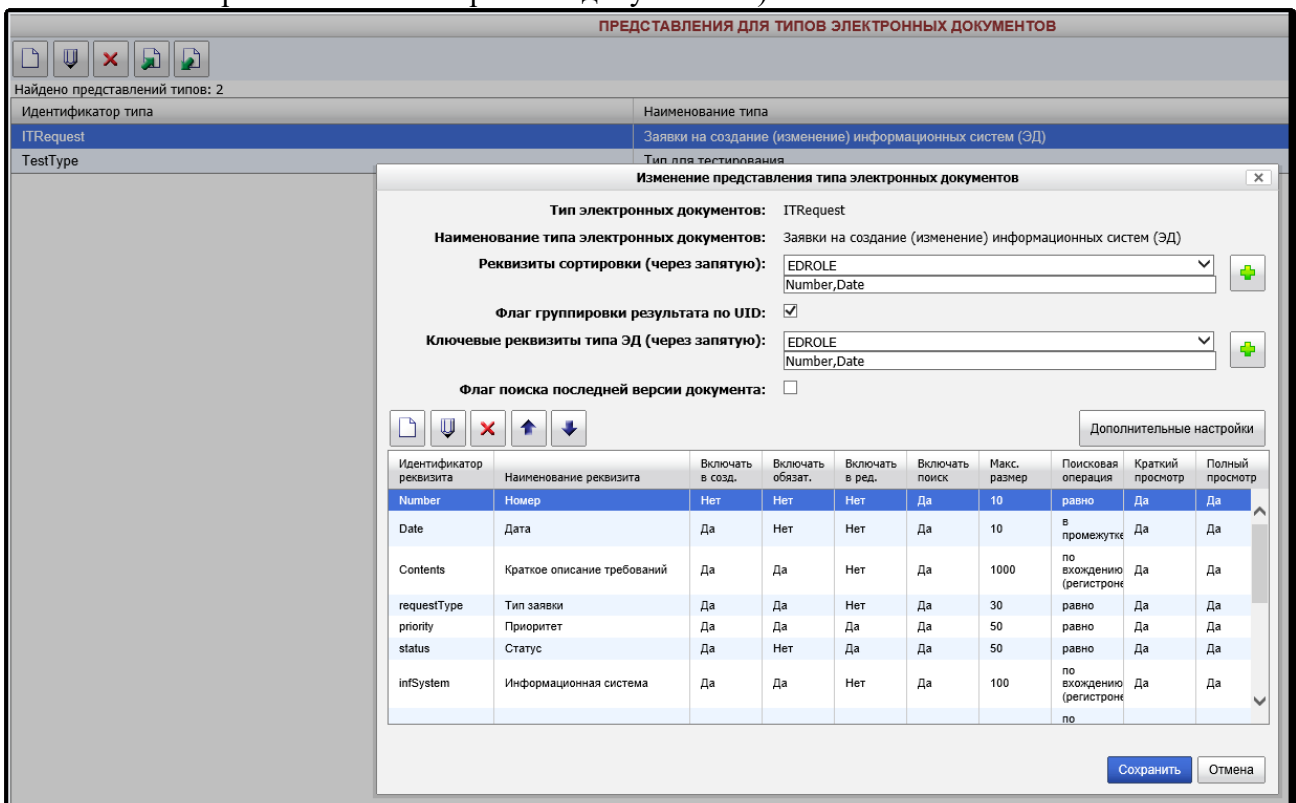


Рисунок 3.2-5 – Изменение представления типа

Изменение реквизита представления типа электронных документов

Тип электронных документов: ITRequest

Наименование типа электронного документа: Заявки на создание (изменение) информационных систем (ЭД)

Идентификатор реквизита: Number

Наименование реквизита: Номер

Флаг включения реквизита в краткий просмотр ЭД:

Флаг включения реквизита в полный просмотр ЭД:

Флаг доступности для создания:   Обязательность

Флаг доступности для изменения:

Флаг включения реквизита в поисковый запрос:

Флаг видимости имени реквизита:

Максимальная длина значения реквизита: 10

Знак операции при поиске: равно

Прикладной тип реквизита: Number

Валидация вводимого значения:

Комментарий-подсказка:

Псевдонимы значений реквизита:

**ПРИМЕЧАНИЕ**  
формат записи псевдонимов значений реквизита: значение1=псевдоним1;значение2=псевдоним2;

Сохранить Отмена

Рисунок 3.2-6 – Тонкая настройка реквизита электронного документа

Идентификатор реквизита фиксирован, а наименование реквизита в каждом типе может быть переопределено.

- **Флаг включения в краткий просмотр:** отображать ли данный реквизит в таблице со списком документов, отображаемых при поступлении на обработку.
- **Флаг включения в полный просмотр:** отображать ли данный реквизит при просмотре документа, поступившего на обработку.
- **Флаг доступности для создания:** должен ли данный реквизит заполняться на визуальной форме при создании документа. При этом может быть отдельно отмечена обязательность его заполнения.
- **Флаг доступности для изменения:** может ли это реквизит редактироваться в процессе обработки документа, если в маршруте будут предусмотрены соответствующие точки.
- **Флаг включения в поисковый запрос:** может ли этот реквизит участвовать в поиске. При выборе данного флага требуется уточнить операцию поиска (равно / по вхождению / в промежутке). Опция «регистронезависимый» означает, что поиск будет идти вне зависимости от регистра символов, указанных пользователем, например, поиск по значению «Краткий» и «краткий» будет давать одинаковый результат (данная опция используется в том случае, если по умолчанию СУБД не поддерживает регистронезависимый поиск).
- **Максимальная длина значения реквизита** позволяет переопределить (в сторону уменьшения) максимальное значение, которое может ввести пользователь при создании / редактировании реквизита.
- **Прикладной тип реквизита:** уточняет тип реквизита помимо базового типа строка или число.
  - *String* – простая строка, для заполнения будет предложен однострочное текстовое поле;
  - *Date* – дата в формате гггг.мм.дд, для заполнения реквизита будет предложен календарь;
  - *Number* – целое число в пределах до указанного значения в поле «Длина». Например, если указано «Длина = 10», то в поле с типом Number можно ввести только целые числа до 10 включительно.
- **Валидация вводимого значения:** маска в формате регулярных выражений, которая задает правильную структуру вводимых данных.
- **Псевдонимы значений реквизитов:** в данном поле указываются допустимые значения для выбора, для заполнения реквизита пользователю будет предоставлен выпадающий список. Варианты указываются следующим образом:  
значение1=псевдоним1;значение2=псевдоним2;

где «значение» – это данные, хранимые в базе данных, а «псевдоним» - данные, отображаемые пользователю.

Например: true=Да;false=Нет.

В этом случае значение реквизита в базе данных будет true или false, а пользователь на форме создания / редактирования реквизитов будет видеть выпадающий список «Да» / «Нет».


После того, как тип электронного документа создан, настроен и сохранен, можно создавать для него процесс (описано в [разделе 5.9](#) данного руководства).

## 4. Справочники

Система ISIDA BPM использует встроенные справочники, к которым относятся «Сотрудники», «Подразделения» (включая иерархию подразделений) и «Рабочий календарь». В процессе создания приложений перечень справочников может быть расширен по требованиям прикладных задач.

### 4.1. Справочник подразделений

Справочник подразделений служит для ведения записей о подразделениях организации и поддержания их иерархической структуры в актуальном состоянии. При создании новой учётной записи подразделения следует указать его полное и сокращенное наименование и индекс (это может быть любое обозначение, например, внутренний код подразделения, номенклатурный индекс и т.п.).

После создания записи подразделения его нужно поместить в дерево с помощью кнопки .

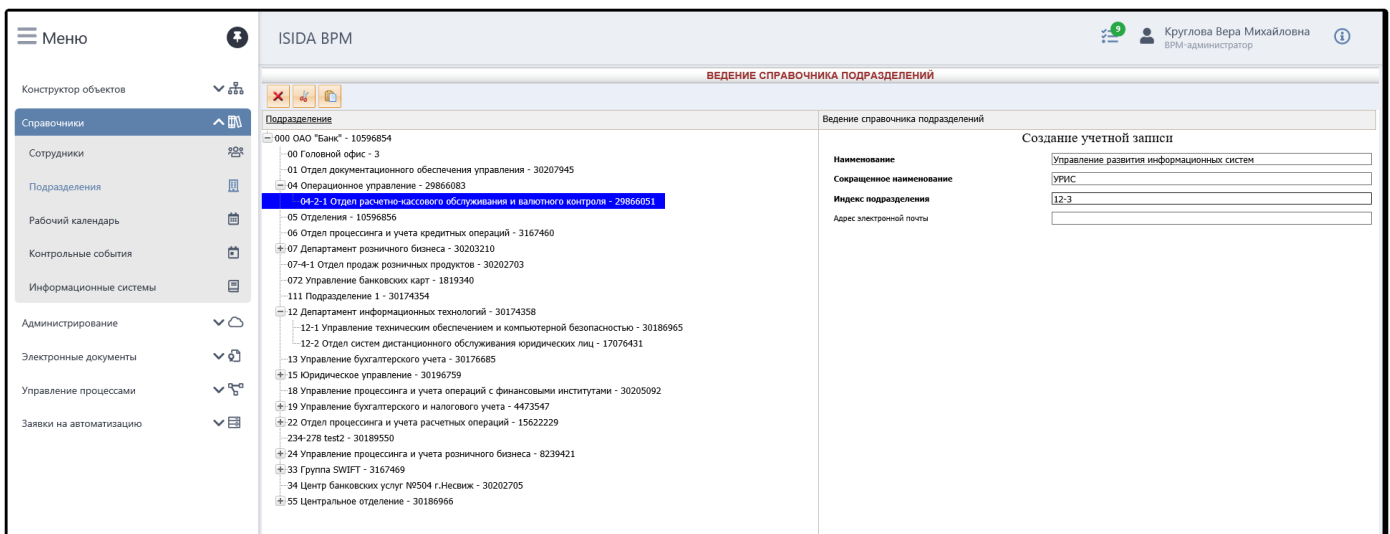


Рисунок 4.1-1 – Создание новой учетной записи подразделения

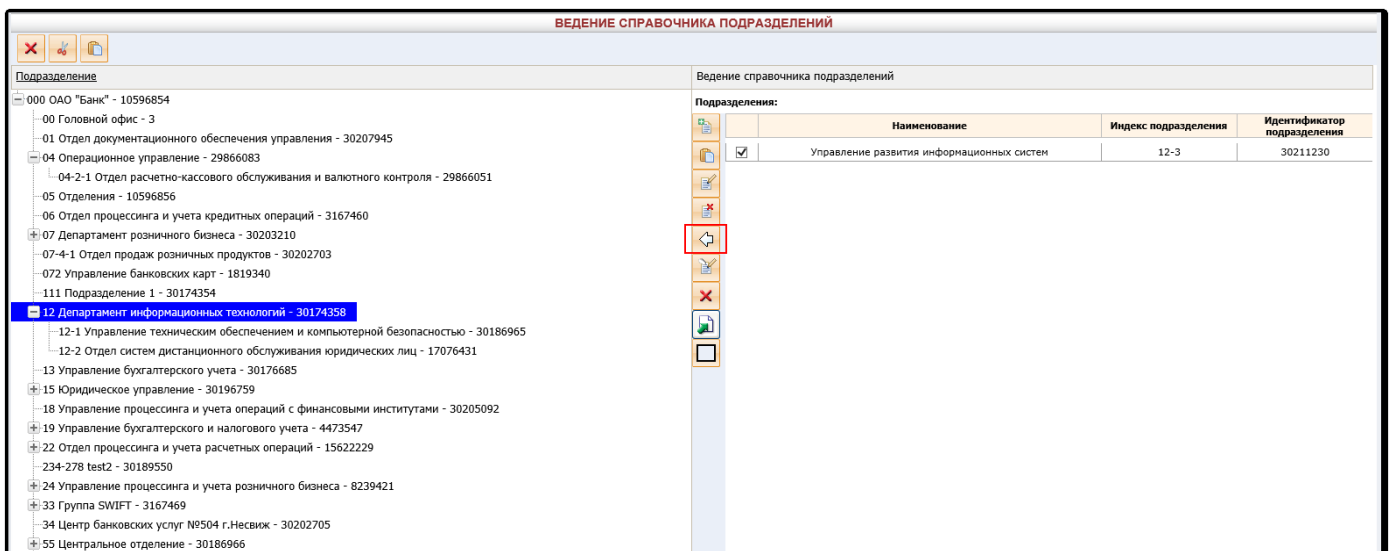


Рисунок 4.1-2 – Помещение подразделения в дерево

Справочник подразделений используется при заполнении полей объекта, на которых указан тип «Подразделение».

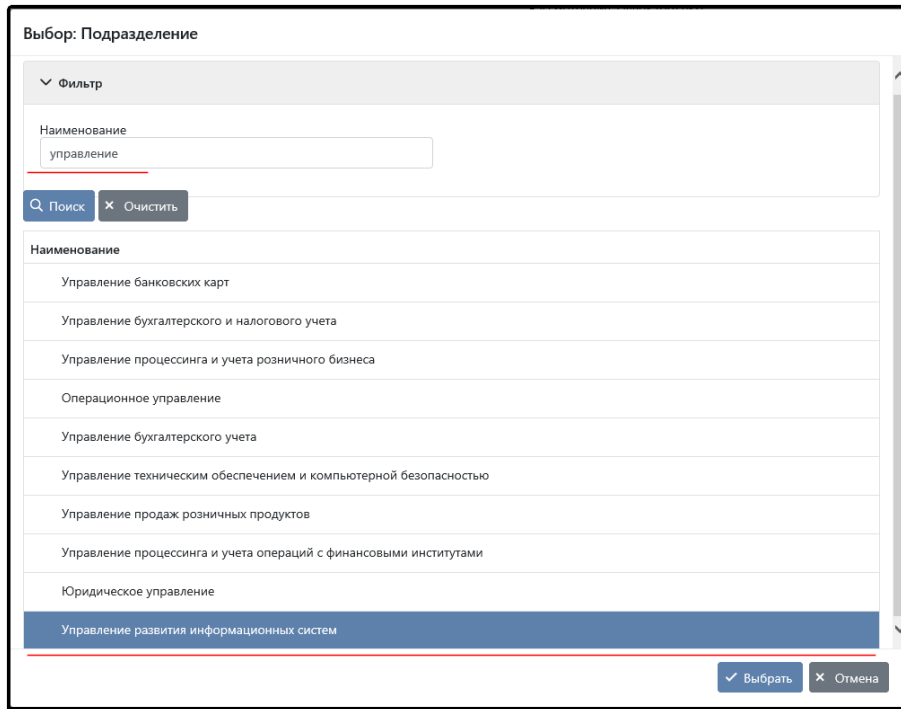


Рисунок 4.1-3 – Использование справочника подразделений при заполнении полей объекта ISIDA BPM

## 4.2. Справочник сотрудников

Справочник сотрудников служит для ведения учётных записей сотрудников организации. В контексте ISIDA BPM понятия «сотрудники» и «пользователи» являются синонимами.

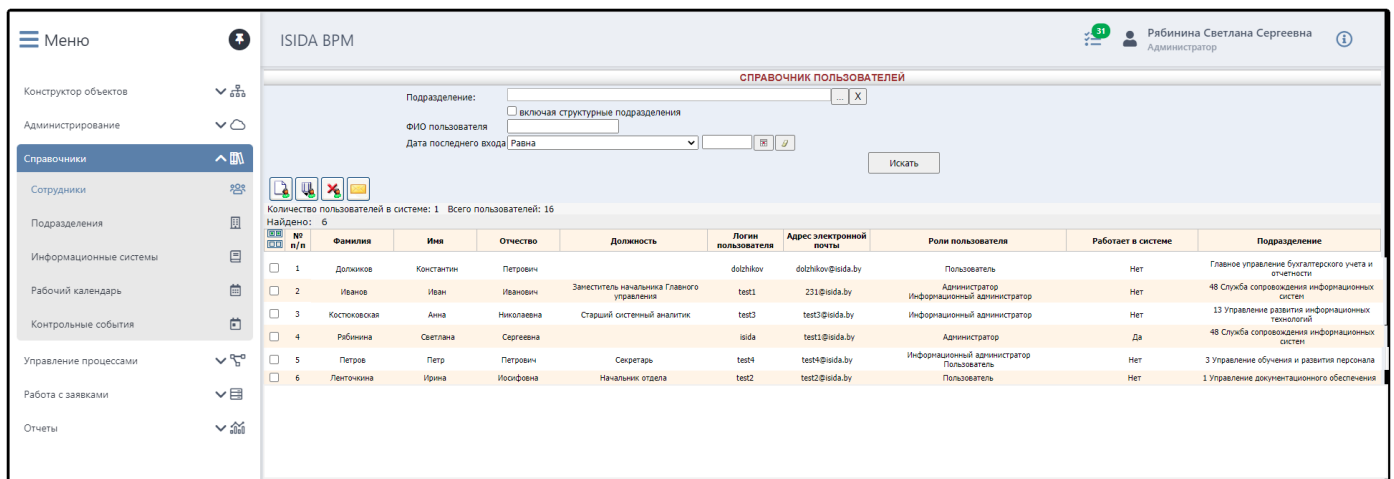


Рисунок 4.2-1 – Справочник пользователей системы

Для создания справочной записи нужно указать ФИО сотрудника, его подразделение, логин, адрес электронной почты, а также установить роль в системе ISIDA BPM.

**пользователь**

Карточка пользователя

Фамилия: Иванов  
 Имя: Иван  
 Отчество: Иванович  
 Подразделение: Центральный аппарат: Главное управление информационных технологий  
 Служба сопровождения информационных систем  
 Должность: Заместитель начальника Главного управления  
 \*Логин пользователя: test1  
 \*Адрес электронной почты: 231@isida.by  
 \*Подразделения пользователя: Служба сопровождения информационных систем  
 Группы типов ЭД: Группа для тестирования (создание документов) [Изменить]  
 Дополнительная информация: [ ]

Роль	Доступна
Администратор	<input checked="" type="checkbox"/>
Информационный администратор	<input checked="" type="checkbox"/>
Пользователь	<input type="checkbox"/>

Добавить Удалить

Подразделение	Учет иерархии
48 Служба сопровождения информационных систем	<input type="checkbox"/>
Центральный аппарат	<input checked="" type="checkbox"/>

Рисунок 4.2-2 – Редактирование справочной записи «Пользователь»

Справочник сотрудников используется при заполнении полей объекта, на которых указан тип «Сотрудник», а также при выборе исполнителей в маршрутных точках, при установке контролёров процесса и т.д.

Выбор: Сотрудник

Фильтр

Фамилия  
васильева

Поиск Очистить

Найдено: 2 << < 1 > >> 15

Фамилия ↑	Имя ↓	Отчество ↓	Должность ↓
Васильева	Наталья	Геннадьевна	Начальник отдела
Васильева	Анна	Фёдоровна	Специалист по тестированию

Выбрать Отмена

Рисунок 4.2-3 – Использование справочника сотрудников при заполнении полей объекта

Рисунок 4.2-4 – Использование справочника сотрудников при заполнении контролёров процесса

### 4.3. Рабочий календарь

Любой бизнес-процесс характеризуется этапами, для исполнителей которых может быть установлен **срок выполнения** задач этого этапа. При установке срока на точке бизнес-процесса имеет значение, каким образом его высчитывать – в **рабочих** или **календарных** днях. Например, на исполнение этапа «Подготовка предложений» исполнителю отведено 15 дней. Если указано «15 рабочих дней», то срок исполнения будет высчитываться с учетом наличия выходных и праздничных дней в этом периоде. Если установлено «15 календарных дней», то выходные и праздники также считаются «рабочими».

Для ведения выходных и праздничных дней в ISIDA BPM предусмотрен специальный справочник «Рабочий календарь» (рис.4.3-1).

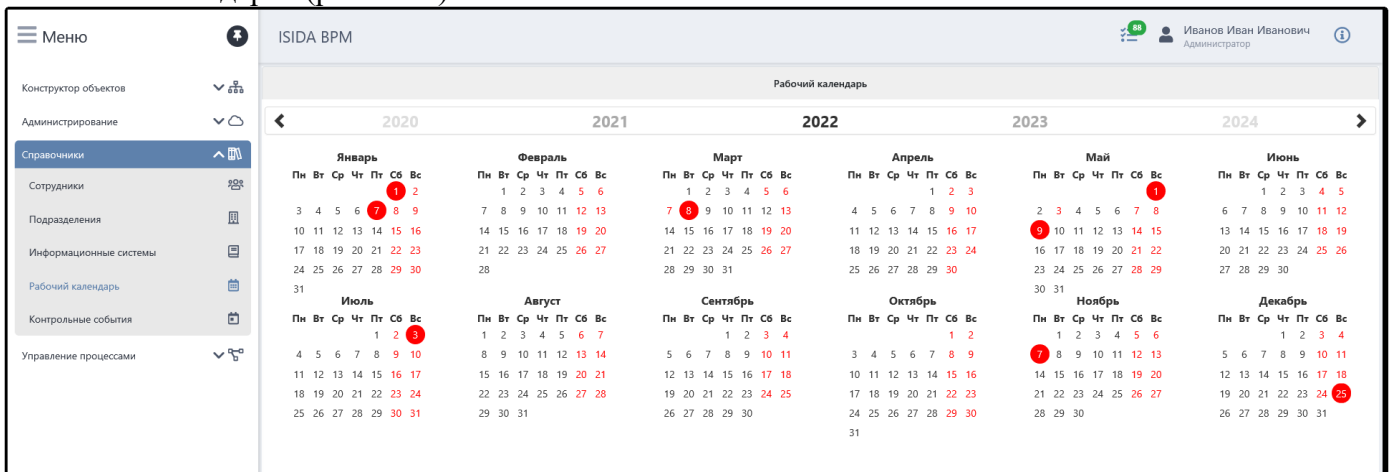


Рисунок 4.3-1 – Сеанс ведения рабочего календаря

Для любого дня года можно указать признак:

- **рабочий** (по умолчанию таковыми считаются все дни года с понедельника по пятницу);
- **выходной** (по умолчанию таковыми считаются все субботные и воскресные дни);
- **праздничный**;
- **ежегодный** выходной/праздник или нет. Если указано, что праздник ежегодный, день автоматически будет установлен таковым и на последующие годы в календаре.



Рисунок 4.3-2 – Настройки выбранного дня в календаре

Для указания переноса рабочего дня на выходной, нужно указать в одном из дней, например, в субботу, признак «Рабочий», а в праздничном дне, допустим в понедельник, «Выходной», и не указывать признак «Ежегодный».

Рисунок 4.3-3 – Настройка переноса рабочего дня на выходной

После того, как справочник наполнен, он за кадром автоматически учитывается системой при расчете сроков исполнения задач бизнес-процесса.

#### 4.4. Справочники приложений

Для заполнения полей объектов могут потребоваться иные справочники, кроме сотрудников и подразделений. Система ISIDA BPM обеспечивает создание и внедрение произвольных справочников, которые требуются приложениям. Предоставляется не только сеанс их ведения, но и компоненты для выбора справочных записей для заполнения полей в объектах.

Справочники – это разновидность «родных» объектов, которые разрабатываются в Конструкторе объектов по общим правилам, при этом объекту устанавливается признак «справочный». Затем для заполнения и просмотра справочных записей разрабатывается специальный комплект форм. В настоящее время формы конструируются командой разработчиков ISIDA BPM по запросу. В результате формы передаются заказчику, вместе с ними предоставляется пункт меню для работы со справочником. В будущем, администраторам ISIDA BPM будет предоставлена возможность не только конструировать структуру справочных объектов, но и разрабатывать простейшие сеансы (визуальные формы) для заполнения и просмотра справочника таких объектов.

В качестве примера можно привести справочник «Информационные системы», который требуется для заполнения объекта «Заявка на изменение информационной системы». У справочной записи

довольно проста структура: «Наименование», «Краткое наименование», «Категория», «Владелец» (ссылка на подразделение) и «Описание». Для администрирования справочника были разработаны формы поиска и просмотра справочных записей (рис. 4.4-1), а также форма редактирования справочной записи (рис. 4.4-2).

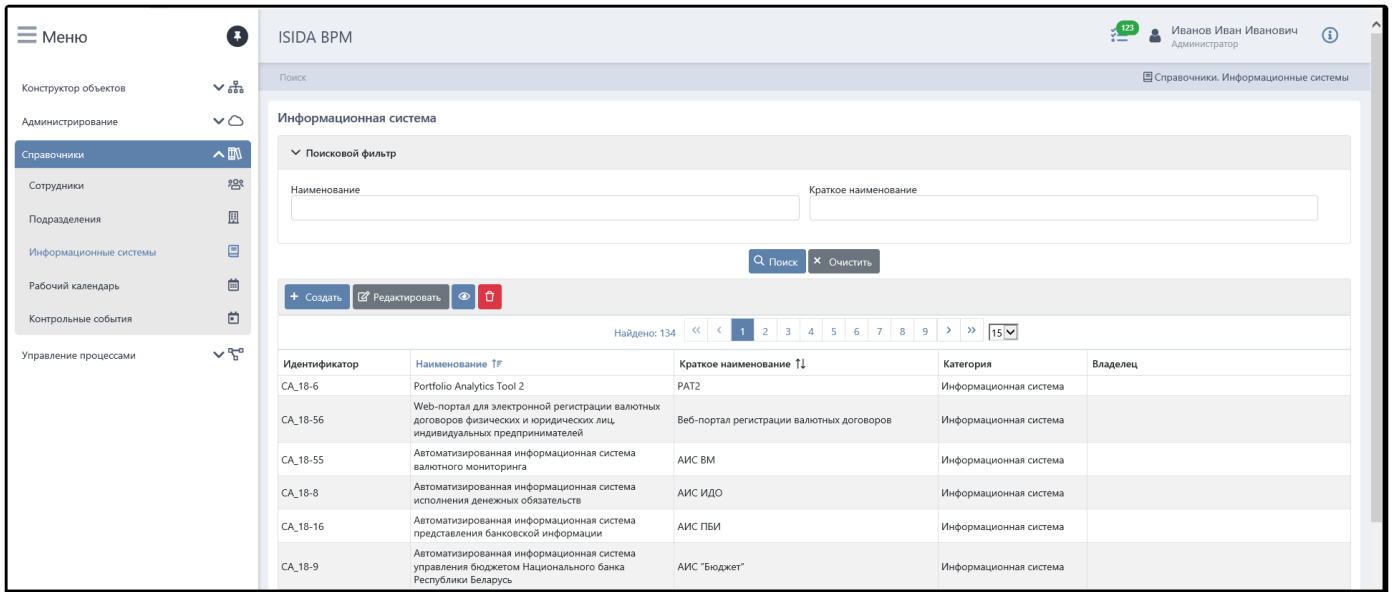


Рисунок 4.4-1 – Справочник «Информационные системы»

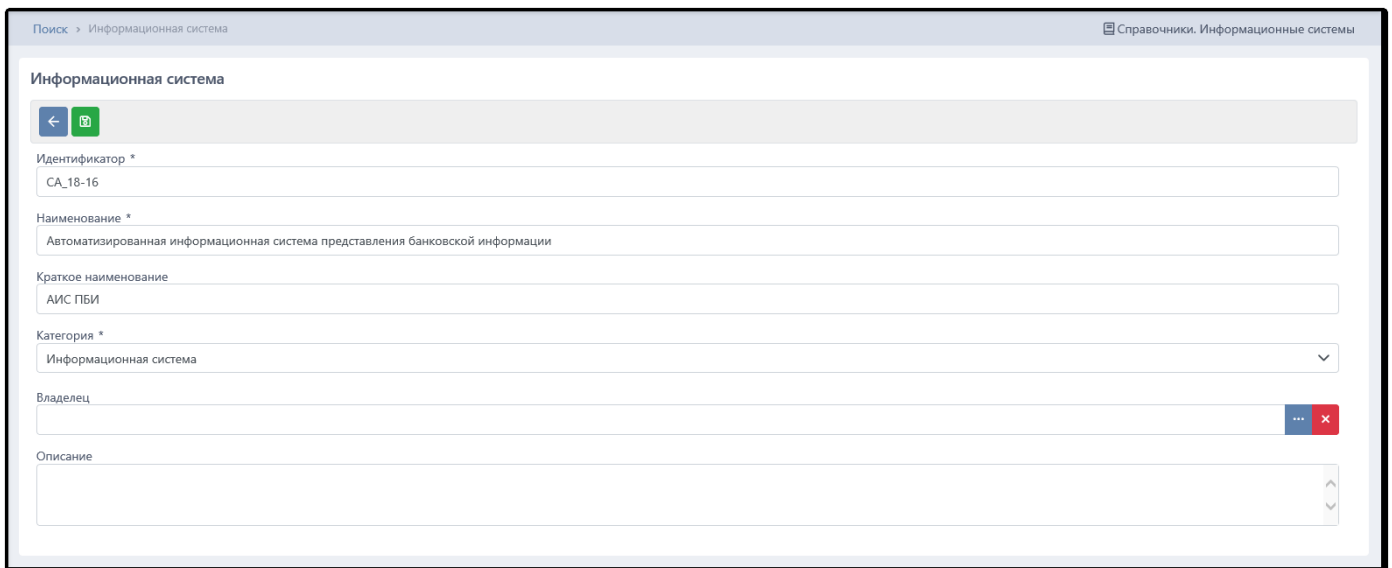


Рисунок 4.4-2 – Редактирование объекта «Информационная система»

Для администрирования справочника предлагается отдельный пункт меню, который может быть включен в любую из ролей, таким образом, права по администрированию справочника могут быть предоставлены любой категории пользователей.

Разработанный справочник может использоваться при описании любого типа объекта. Например, в объекте «Заявка на изменение информационных систем» объявляется поле «Информационная система», тип которого – справочник «Информационные системы». Форма для работы с Заявкой будет содержать соответствующий компонент для выбора значений, который будет осуществлять поиск и выбор одного или нескольких значений (если поле множественное) из указанного справочника (рис. 4.4-3).

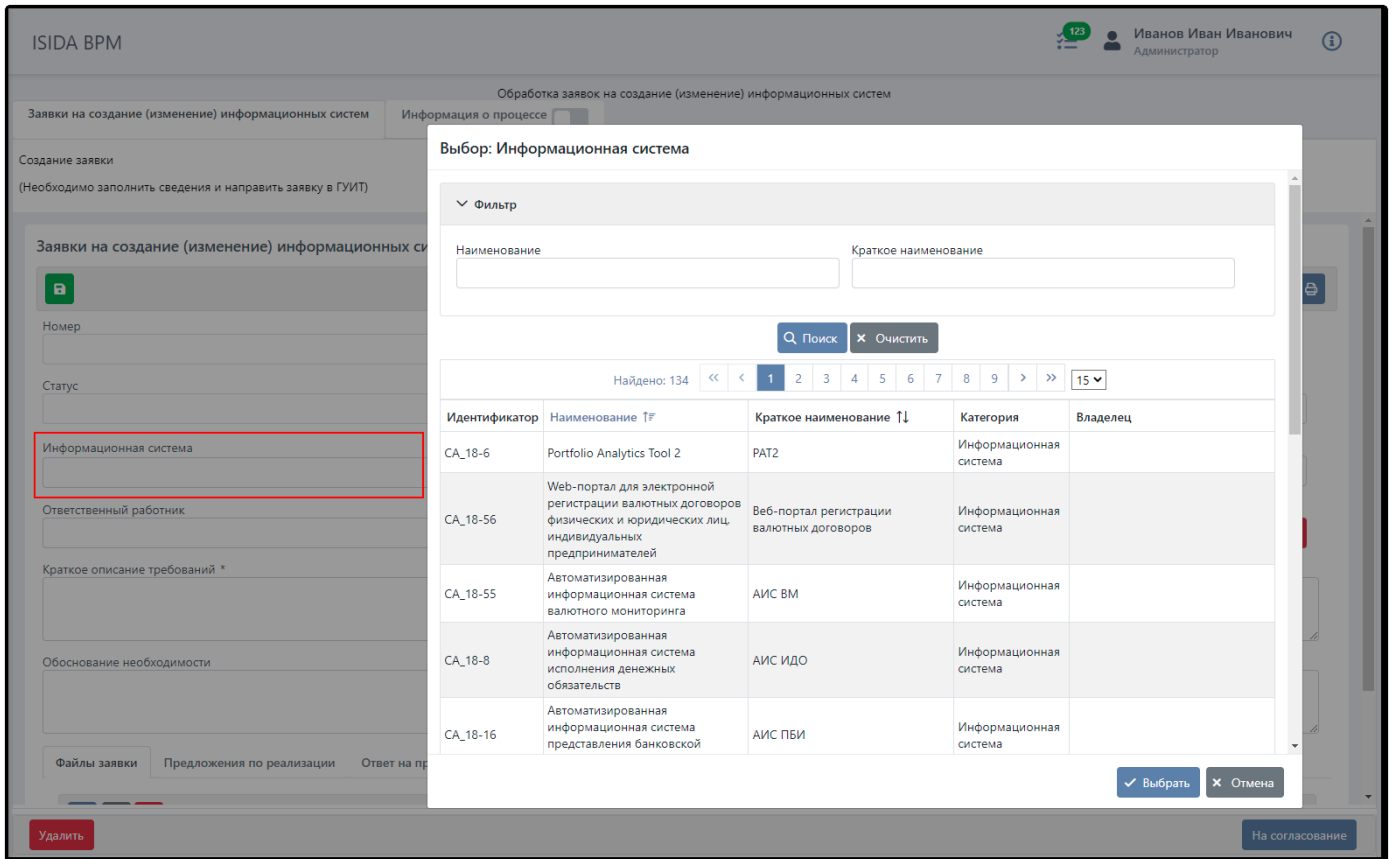


Рисунок 4.4-3 – Использование справочника «Информационные системы»

## 5. Дизайнер процессов

### 5.1. Общие сведения

Объекты и электронные документы в системе ISIDA BPM движутся по заранее описанным бизнес-процессам (маршрутам). Понятия бизнес-процесс (процесс) и маршрут для простоты можно считать синонимами: в одном случае корректнее использовать понятие *процесс*, в другом – *маршрут*. Для моделирования, описания и настройки маршрутов служит специальная визуальная графическая среда разработки *Дизайнер процессов*.

Система ISIDA BPM в своем дизайнере использует *упрощенную нотацию BPMN*. В частности, в ней нет таких элементов, как плавательные дорожки, пулы, отсутствуют некоторые виды потоков и шлюзов. Нотация упрощена с целью облегчения конструирования бизнес-процессов администраторами, а также для максимальной простоты понимания процесса его участниками, ведь процесс, смоделированный и настроенный администратором, будет отображаться в том же виде всем участникам процесса.

Таким образом, в ISIDA BPM управляющие элементы дизайнера представляют минимальный, но достаточный набор, позволяющий смоделировать сложный процесс с развилками, условиями, возвратами, но при этом не загромождают его внешний вид. Графическое представление бизнес-процесса в дизайнере с одной стороны напоминает ориентированный граф с точками (этапами бизнес-процесса) и переходами, а с другой стороны схож с блок-схемой (также с блоками-точками, условиями и переходами между ними). Для моделирования и прочтения таких графов не требуется специально изучать нотацию BPMN, графическое представление максимально доступно для понимания.

Для упрощения графического представления бизнес-процесса могут быть исключены переходы, указывающие на движение объекта в обратном направлении. Вместо этого используются так называемые «прыжки назад», которые описываются в параметрах точки маршрута.

*До запуска* управляемого объекта по маршруту, сам процесс должен быть **смоделирован** в дизайнере, то есть разработано его описание. Администратор, погруженный в контекст задачи, продумывает, каким будет маршрут (сценарий) для обработки для конкретного объекта: какие он будет содержать этапы, будут ли в нем развилки (условия), возвраты, вложенные подпроцессы, сколько исполнителей будет выполнять ту или иную задачу, будут ли задачи выполняться последовательно или параллельно и т.д. Администратор продумывает условия переходов от одного этапа к другому, именуется этапы и кнопки для обработки этих этапов, которые будет нажимать сотрудник и т.д. В момент сохранения процесса сеанс дизайнера верифицирует граф на «правильность», чтобы не было точек без переходов, одинаковых условий, ведущих в разные точки, циклических этапов маршрута, и при найденных ошибках сообщает об этом администратору.

Как и в случае с объектами, описание процесс по умолчанию создается в статусе «**Черновик**», который нужно **опубликовать**. В статусе черновика процесс не может быть использован в эксплуатации, программные компоненты, которые отвечают за запуск процессов, его как бы «не видят». Для черновиков можно выполнять любые корректировки. После публикации процесса он может использоваться для постановки на него управляемых объектов.

Довольно удобно разрабатывать и проверять процессы в тестовой зоне, после чего максимально быстро и без искажений копировать их в зону промышленной эксплуатации. Такую миграцию процесса можно выполнить с помощью функций экспорта/импорта. Для обмена данными используется файл в формате JSON.

## 5.2. Список процессов

Любой процесс, смоделированный в ISIDA BPM, характеризуется статусом:

- **Черновик** – новые процессы, находящийся на стадии конструирования и не опубликованный в системе. К черновику можно возвращаться сколько угодно раз, дополняя его по мере необходимости.
- **Опубликованный** – процесс, смоделированный администратором, который может быть использован в работе.
- **Удаленный** – процесс, который был удален администратором (при этом сохраняется функция их восстановления).

Все процессы, созданные в системе, располагаются в сеансе «Дизайнер процессов» на трех закладках согласно статусу процесса (рис.5.2-1).

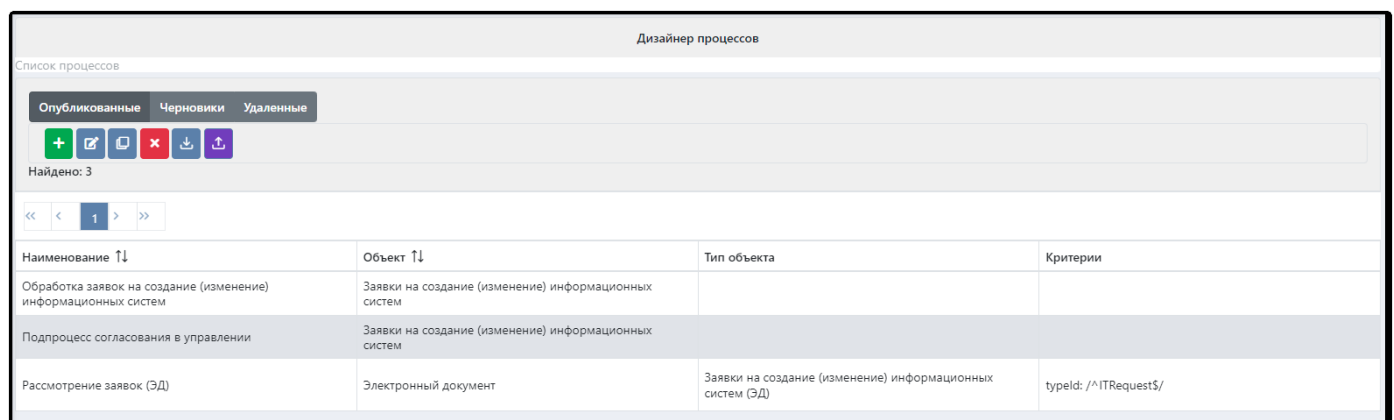


Рисунок 5.2-1. Сеанс «Список процессов» в дизайнере







Основные кнопки для работы с процессом:



- создание нового процесса;



- редактирование ранее созданного процесса;

-  - создание копии процесса (с указанием нового наименования);
-  - удаление процесса (помещается в список «Удалённые»);
-  - экспорт процесса в файл формата JSON;
-  - импорт процесса из файла формата JSON;
-  - публикация процесса-черновика (из списка «Черновики» перемещается в список «Опубликованные»);
-  - восстановление ранее удаленных процессов (из списка «Удалённые» перемещается в список «Черновики»).

### 5.3. Создание процесса

При создании нового процесса система автоматически предлагает «заготовку», состоящую из начальной, конечной точки и одной задачи (рис. 5.3-1).

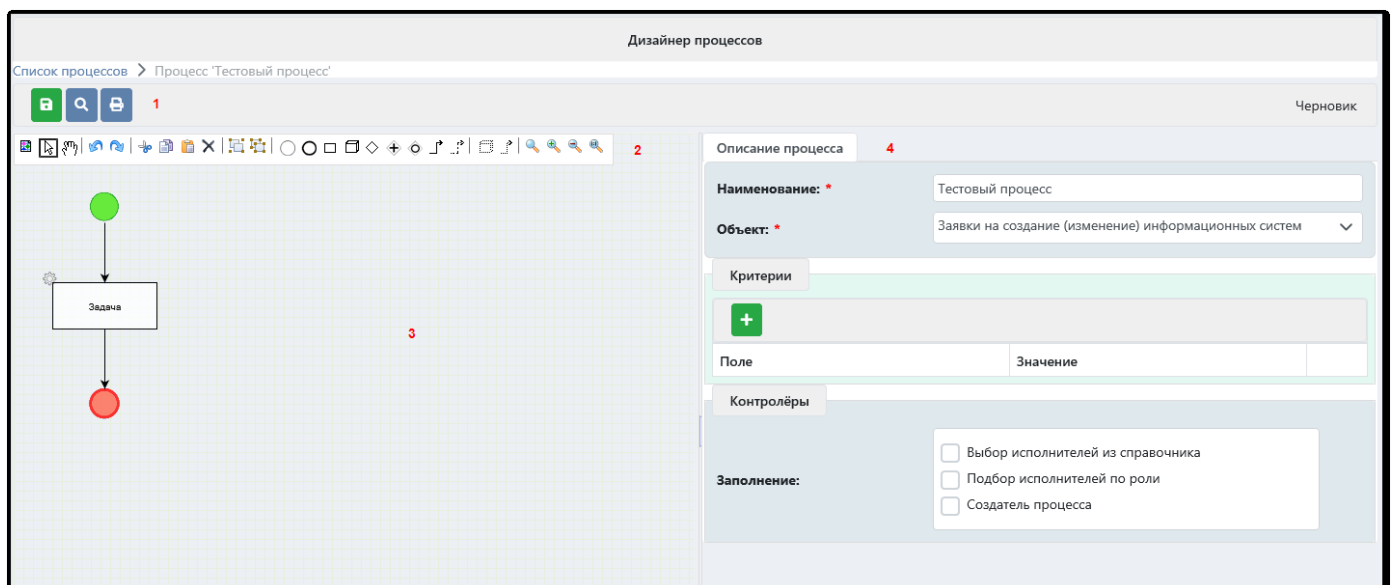





Рисунок 5.3-1. Форма создания/редактирования процесса

Сеанс для редактирования процесса состоит из следующих областей (отмечены на рисунке 5.3-1):

#### 1) **Панель кнопок:**

-  - сохранение процесса;
-  - для просмотра текущего процесса на отдельной закладке браузера;
-  - перенос шаблона процесса на отдельную закладку браузера в формате, пригодном для печати.

#### 2) **Панель инструментов для макетирования процесса:**



- 3) **Рабочее пространство дизайнера** (расположено в левой части формы), в котором макетируется редактируемый процесс.
- 4) **Панель описания процесса** (рис. 5.3-2).

Описание процесса

Наименование: \* Тестовый процесс

Объект: \* Заявки на создание (изменение) информационных систем

Критерии

+

Поле	Значение

Контролёры


Заполнение:

- Выбор исполнителей из справочника
- Подбор исполнителей по роли
- Создатель процесса

Рисунок 5.3-2. Вкладка «Описание процесса»

Для создания нового процесса нужно указать наименование процесса, выбрать **объект** или **электронный объект**, для которого создается процесс (предлагаются объекты, опубликованные в ISIDA BPM с признаком «Основной»).

В левой части сеанса с помощью инструментов администратор системы графически макетирует будущий процесс. Ему доступны следующие возможности:

1.  - просмотр плана процесса. В новом окне отрывается уменьшенный план всего процесса, что облегчает восприятие сложных и объемных маршрутов (рис.5.3-3).

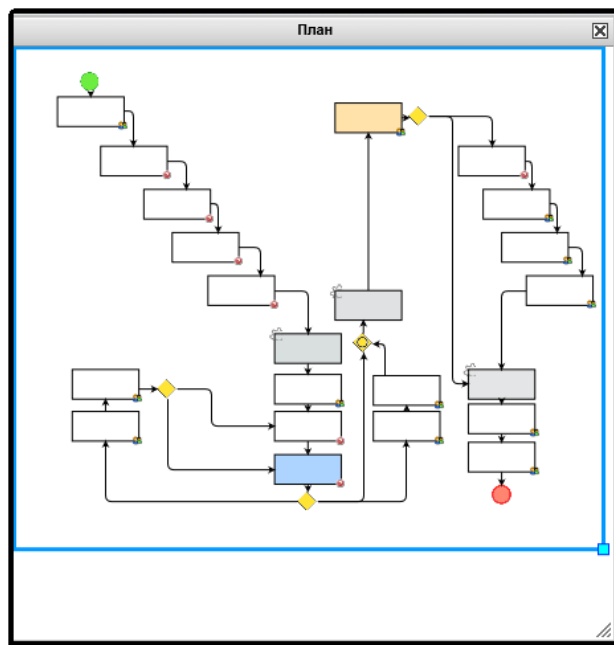











Рисунок 5.3-3. План процесса







2.  - выбор элемента процесса (точки, перехода, условия).
3.  - перемещение границ видимой части изображения процесса без изменения его масштаба с целью просмотра областей процесса, расположенных за границами окна.

4.  - отмена действия и возврат действия сразу после его отмены.
5.  - вырезание, копирование, вставка и удаление элементов процесса.
6.  - изменение размеров изображения процесса относительно рабочего пространства: «По размеру окна» , «Приблизить» , «Отдалить» , «Обычный размер» .

В рабочем пространстве пользователь создает черновик процесса путём перетягивания нужных элементов, расположенных на панели инструментов:



Краткое описание инструментов согласно нотации BPMN:

-  - начальная точка маршрута. При добавлении в маршрут отображается в виде зеленой точки . С этой точки начинается любой процесс.
-  - конечная точка маршрута. Отображается в виде красной точки, должна завершать процесс . Конечных точек может быть несколько, например, в случае, когда процесс разветвлялся на несколько независимых процессов.
-  - задача (этап, процедура, точка маршрута). Имеет множество свойств, которые заполняются в правой части дизайнера. Подробнее о свойствах процедур описано ниже.
-  - подпроцесс. Элемент, который представляет собой отдельный самостоятельный процесс. Для того, чтобы использовать подпроцесс, необходимо сначала создать отдельный процесс, сохранить его, опубликовать, а затем выбрать из выпадающего списка в настройках элемента «Подпроцесс» (рис.5.3-4). Работа с подпроцессами описана в [разделе 5.6](#) данного руководства.

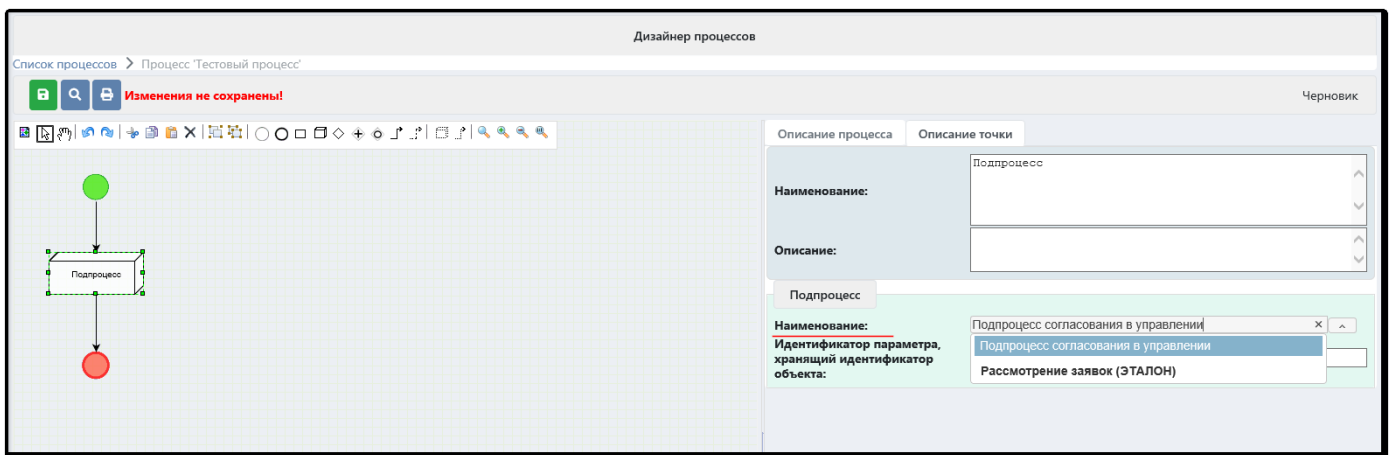




Рисунок 5.3-4. Настройка элемента «Подпроцесс»

 - переход. Используется для соединения отдельных процедур, подпроцессов и условий между собой в единый процесс. Переход можно «нарисовать» протянув стрелку  от центра начальной точки до границы конечной точки (рис.5.3-5,6).

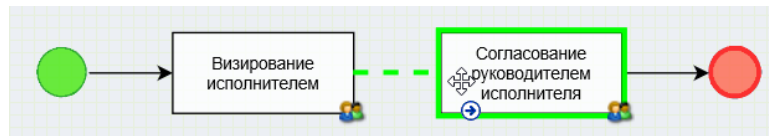
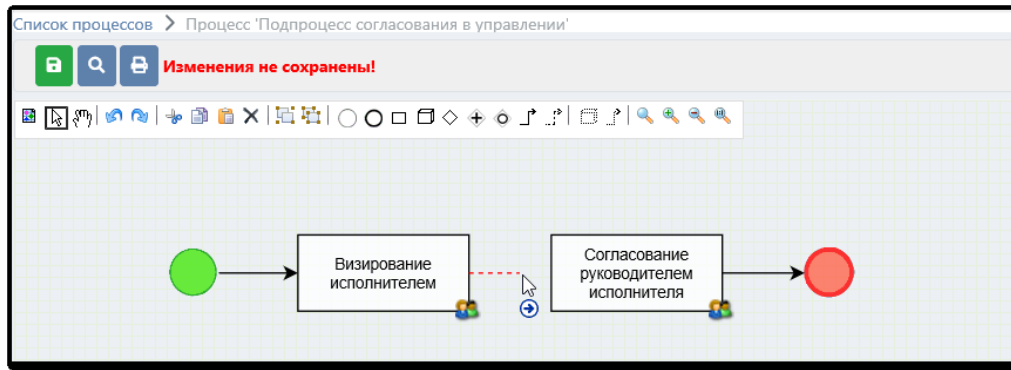


Рисунок 5.3-5,6. Создание перехода между процедурами

◇ + ○ - исключающий, параллельный и включающий шлюзы. Подробно описаны в [разделе 5.5](#) данного руководства.

└ - связь с дочерней точкой. Подробно описана в [разделе 5.10](#) данного руководства.

### Рекомендации по расположению элементов процесса

Желательно располагать все элементы процесса так, чтобы его было максимально удобно «читать». Следует стремиться к тому, чтобы переходы, связующие процедуры, не пересекались. Процедуры можно располагать строго сверху вниз или слева направо (если процесс небольшой) либо каскадом, лестницей, если процесс громоздкий. Нужно учитывать, что тот внешний вид, в котором представлен граф процесса в дизайнера, будет представлен и исполнителям маршрута (обычным пользователям системы). Чем понятнее и проще он будет изображен, тем проще будет в нем разобраться исполнителям.

Громоздкость графа можно уменьшить с помощью использования подпроцессов: самостоятельных процессов, которые на магистральном процессе занимают всего один блок (элемент).

Для наглядности блокам можно устанавливать цвет. Удобно, когда, например, все автоматические процедуры отмечены одним и тем же цветом, или, когда однотипные процедуры также имеют один цвет. Можно отмечать цветом задачи для одного и того же подразделения. Но при этом не следует перегружать громоздкий маршрут слишком большим количеством разных цветов. Цвет заливки устанавливается путём выбора из контекстного меню (рис. 5.3-7).

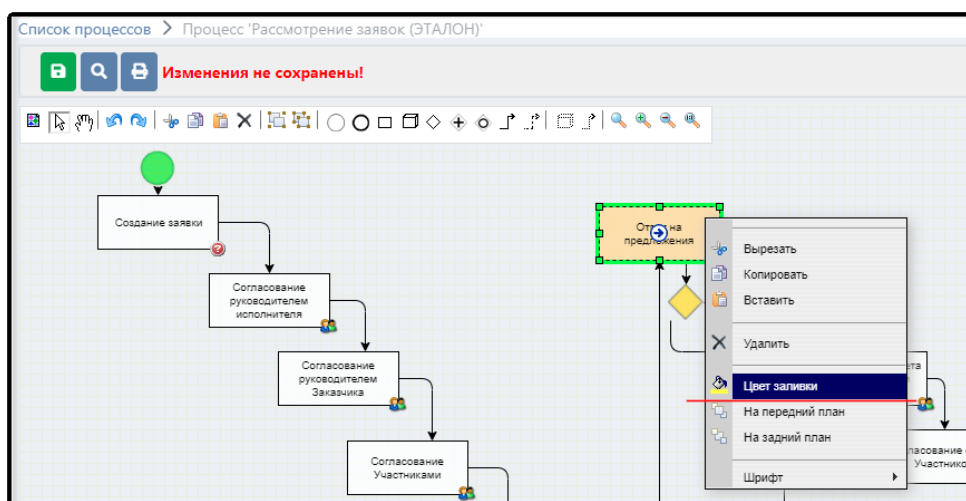


Рисунок 5.3-7. Установка цвета в точке маршрута



### Рекомендации по именованию элементов процесса

Этапам процесса нужно стараться давать уникальные и понятные русскоязычные наименования, например: «Согласование участниками инициативной группы», «Согласование юридическим отделом», «Подписание руководителем подразделения», «Контроль сроков исполнения заявки» и т.д. Можно в наименовании уточнить непосредственно цель, задачу исполнителя данной процедуры с помощью таких слов, как «контроль», «запрос», «выбор», «подготовка» и т.д. Исполнители процедур будут понимать, к какому этапу обработки объекта они относятся, и какая цель перед ними поставлена. Если от исполнителя требуется согласование документа, то и в наименовании нужно указать «Согласование». Если этап предполагает утверждение, подписание, визирование – именно эти слова желательно указать в наименовании задачи.

Автоматическим процедурам также нужно давать понятные наименования, отражающие действия процедуры, например, «Установка статуса заявки «Реализация предложений». Таким образом, смысл назначения процедуры будет понятен и администраторам, и исполнителям.

### 5.4. Настройка точек маршрута

В правой части сеанса расположены настройки и свойства точки (рис.5.4-1).

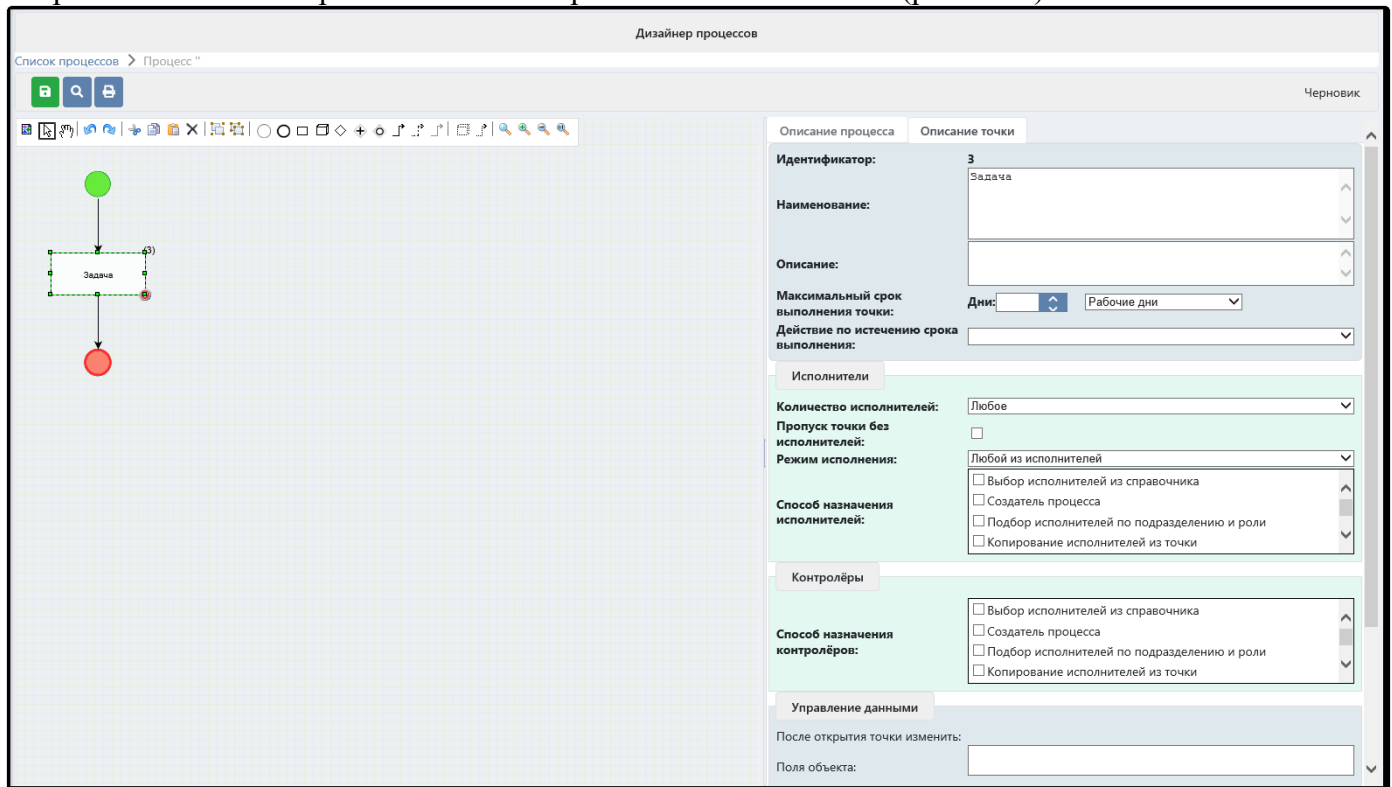


Рисунок 5.4-1. Настройка свойств маршрутной точки

#### Основные свойства маршрутной точки

**Идентификатор** – номер точки по порядку. Присваивается точкам и переходам автоматически при размещении элементов на форме дизайнера.

**Наименование** – русскоязычное название маршрутной процедуры, которое будет отображено в сеансе пользователя. Правила наименования процедур описаны выше.

**Описание** – краткое описание маршрутной процедуры, заполняется при необходимости. Исполнитель точки будет видеть и наименование, и описание процедуры, поэтому в них следует вкладывать цель задачи, исполнение которой требуется от исполнителя.

**Максимальный срок выполнения точки** – количество дней, которые даются пользователю для выполнения процедуры. Может исчисляться в рабочих или в календарных днях. Настройка праздничных дней и переносов ведется в справочнике «Рабочий календарь», работа с которым описана в [разделе 4.1](#). В зависимости от количества дней система автоматически устанавливает срок исполнения на точке маршрута.

**Действие по истечению срока выполнения** - пропустить или приостановить точку, когда срок ее исполнения истечет. Если выбрано действие «пропустить», точка будет пропущена и исполнения перейдет к следующей. Если выбрано «приостановить», точка и процесс будут приостановлены. В каждом из этих случаев будет выброшено соответствующее системное событие.

#### **Блок «Исполнители»:**

**Количество исполнителей** – правило, определяющее количество исполнителей в точке маршрута. Возможные значения:

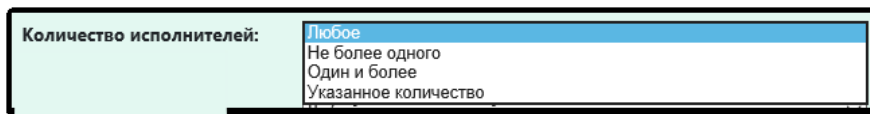


Рисунок 5.4-2. Настройка параметра «Количество исполнителей»

Правило определяет, сколько исполнителей может быть в данной точке. Например, если указано значение «Не более одного», а при запуске процесса пользователь попытается добавить несколько, система проконтролирует соответствие этому правилу и сообщит об ошибке.

**Пропуск точки без исполнителей** – действие системы в случае, если в точке нет исполнителей. При установке галочки «Пропустить», точка будет пропущена и открыта следующая за ней. Если галочка не установлена, точка и процесс будут приостановлены.

**Режим исполнения** – каким образом должна быть исполнена процедура, если исполнителей несколько: всеми исполнителями, указанными в точке, или хотя бы одним из них. Если выбрано значение «Любой из исполнителей», то для исполнения точки достаточно обработать ее одним из исполнителей. Если выбрано значение «Все исполнители (параллельно)», то для завершения процедуру должен обработать каждый из исполнителей, и при этом они работают параллельно. Если выбрано значение «Все исполнители (последовательно)», то для завершения процедуру должен обработать каждый из исполнителей, и при этом они работают последовательно один за другим.

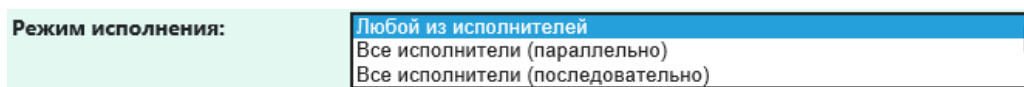


Рисунок 5.4-3. Настройка параметра «Режим исполнения»

**Способ назначения исполнителей** – правило автоматического назначения исполнителей для точки маршрута. Варианты предусмотренных системой правил:

- **Выбор исполнителей из справочника.** Можно указать конкретных пользователей системы путем выбора из справочника. Такой способ используется, когда состав исполнителей фиксированный и известен заранее.
- **Копирование исполнителей из точки.** Исполнители данной точки будут автоматически скопированы с указанной точки, установленной администратором. Такое правило может применяться для однотипного заполнения состава исполнителей (достаточно указать их в одной точке и установить правило копирования во всех остальных, тогда при корректировке исполнителей достаточно внести изменение только в исходную точку, все остальные скопируют этот состав). Также данное правило может быть удобно в следующем случае:

состав исполнителей в точке не известен, но известно, что он такой же как в другой точке маршрута.

- **Подбор исполнителей по функциональной роли.** Необходимо выбрать функциональную роль из доступного перечня (например, «Администратор», «Пользователь»). В качестве исполнителей будут автоматически установлены все сотрудники с указанной функциональной ролью.
  - **Подбор исполнителей по подразделению и роли.** В данном варианте правила подбор исполнителей происходит по двум параметрам: роль и принадлежность определенному подразделению. Например, роль «Ответственный за делопроизводство», подразделение «Управление статистической отчетности». Исполнителями точки станут все сотрудники данного подразделения, обладающие указанной ролью. Роль заранее выбирается из доступного списка, а вот подразделение может принимать различные значения, поэтому правило необходимо уточнить – в каком поле объекта следует взять подразделение. Это означает, что необходимо указать поле объекта (идентификатор поля), который отвечает за подразделение (например, *department*). Допускается любой уровень вложенности полей через точку, например: поле объекта «Информационная система» (*infSystem*), в котором есть поле «Владелец» (*owner*), у которого есть поле «Подразделение», следует заполнить значением *infSystem.owner.department*.
  - **Руководитель подразделения.** Исполнителем данной точки маршрута станет сотрудник подразделения, отмеченный признаком «Руководитель». Если таких сотрудников в подразделении несколько, все они будут назначены исполнителями точки. Для вычисления подразделения необходимо указать поле (идентификатор поля) в котором хранится его значение (например, *department*). Допускается любой уровень вложенности полей через точку, например *infSystem.owner.department*.
- Дополнение:** наполнение точки правилом «руководитель подразделения» происходит непосредственно перед открытием точки, для которой установлено это правило.
- **Создатель процесса.** В качестве исполнителя точки будет назначен создатель (инициатор) экземпляра бизнес-процесса.

Способ назначения исполнителей:	
<input checked="" type="checkbox"/>	Выбор исполнителей из справочника
<input type="checkbox"/>	Подбор исполнителей по подразделению и роли
<input type="checkbox"/>	Копирование исполнителей из точки
<input type="checkbox"/>	Руководитель подразделения

Введите первые буквы фамилии для поиска

ФИО	Логин
Иванов Иван Иванович	test1

Рисунок 5.4-4. Настройка параметра «Заполнение точки исполнителями»

#### **Блок «Контролёры»:**

**Способ назначения контролёров** – правило автоматического заполнения точки контролирующими сотрудниками. Варианты правил схожи правилам для заполнения точки исполнителями.

#### **Блок «Управление данными»:**

Блок предназначается для установки правила, по которому будет меняться поле объекта после открытия точки.

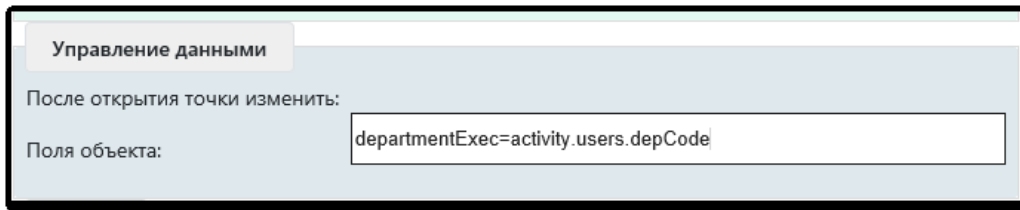


Рисунок 5.4-5. Настройка параметра «Управление данными»

Реализован следующий синтаксис:

левая часть, до знака равно (=): поле объекта которое необходимо изменить. Старые значения при установке новых удаляются.

правая часть, после знака равно (=): константа *activity.users.depCode* - код подразделения исполнителя (исполнителей) точки. Можно указывать несколько полей через запятую (",").

Например: *departmentExec=activity.users.depCode*

#### **Блок «Подписка на события»:**

Отвечает за создание подписок на события, которые выбрасываются конкретной точкой. Подробнее о подписке на события описано [в разделе 8.3](#).

#### **Блок «Задача»:**

Этот блок отвечает за настройку возможностей, которые будут доступны сотруднику при выполнении этого этапа маршрута. Набор таких возможностей очень широк: это и кнопки, которые будут предложены пользователю, и фоновое изменение полей объекта, выброс событий, возможность ввести комментарий при выполнении процедуры или задать текстовое поручение другим исполнителям процесса, установить для них срок выполнения, выбрать исполнителей этапов маршрута и т.д. Для разных организаций, использующих ISIDA BPM, задачи могут быть совершенно разными и зависеть из специфики организации. Описание работы с ними содержится в [разделах 5.7, 5.8](#).

### **5.5. Ветвления и слияния в процессе**

ISIDA BPM даёт возможность строить не только линейные процессы, но и процессы с ветвлениями. Маршрут можно настроить таким образом, что сценарий будет зависеть от заранее заданных условий и процесс будет идти по разным «ветвям».

Для моделирования ветвлений Дизайнер процессов предоставляет следующие инструменты:

◇ + ○ - исключающий, параллельный и включающий шлюзы. Рассмотрим каждый из них.

**◆ Исключающий шлюз (XOR, «Исключающее ИЛИ»)** используется для ветвления процесса на несколько альтернативных потоков, переход к каждому из которых зависит от выполнения некоторого условия. При использовании исключающего шлюза можно продолжить выполнение процесса только по **одному из возможных условных переходов**. Среди переходов, исходящих из исключающего шлюза, допускается использование **перехода по умолчанию** (без указания условия перехода): если ни одно из условий не выполняется, дальнейшее выполнение процесса продолжится по переходу по умолчанию.

После добавления всех переходов из шлюза необходимо настроить условия исходящих переходов (условия перехода становятся доступными в случае наличия у шлюза двух и более исходящих переходов). Условие нужно указывать в поле рядом с наименованием процедуры, на которую осуществится переход (рис. 5.5-1).

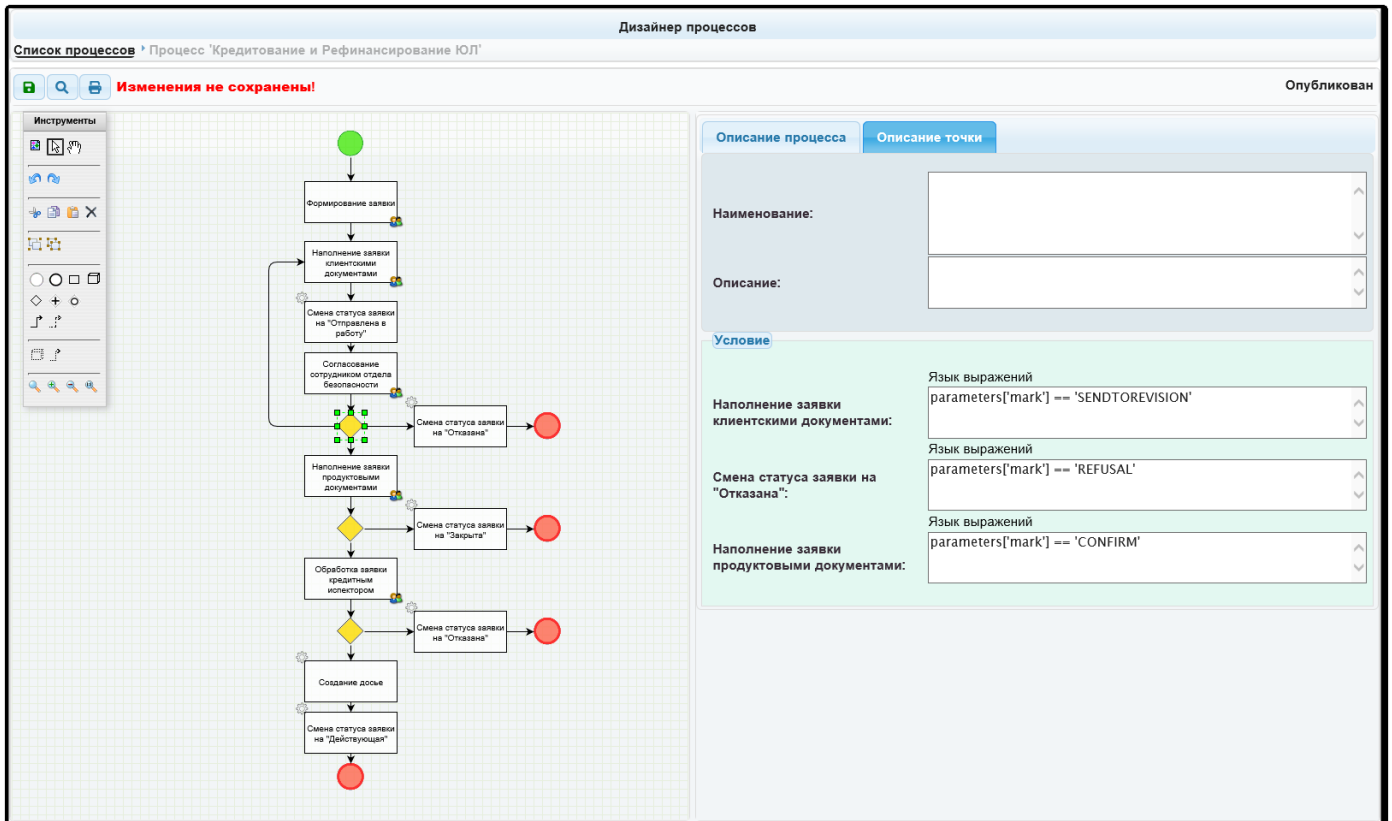


Рисунок 5.5-1. Отображение условий исходящих переходов

Для добавления условия необходимо нажать на текстовое поле, затем выбрать в выпадающем списке тип реализации и указать само условие (рис. 5.5-2). Условие указывается с учётом параметров, которые генерирует предыдущая процедура (параметры ее завершения) либо исходя из реквизитов объекта, для которого создан этот бизнес-процесс. Идентификаторы предлагаются пользователю в сеансе добавления условия.

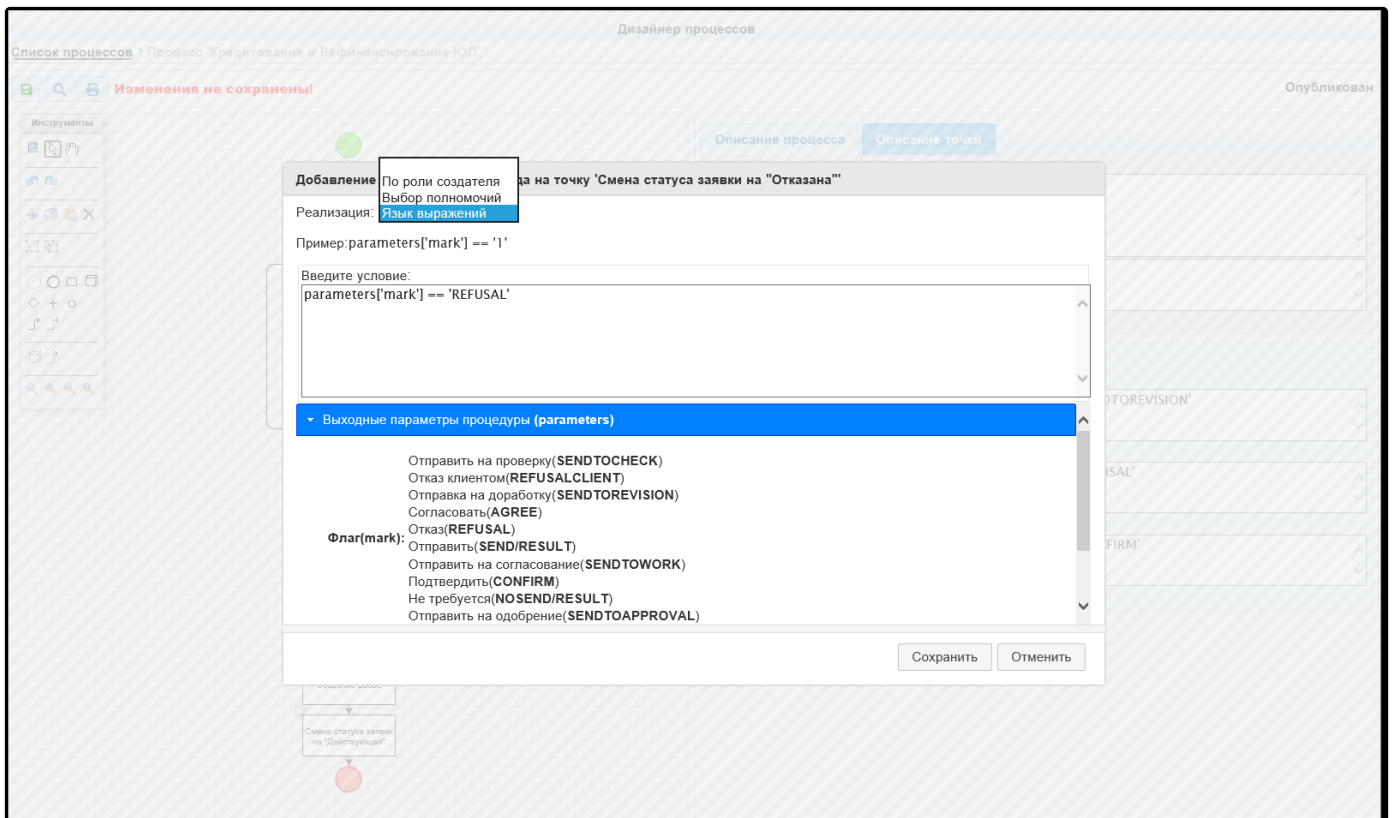


Рисунок 5.5-2. Выбор типа реализации и добавление условий

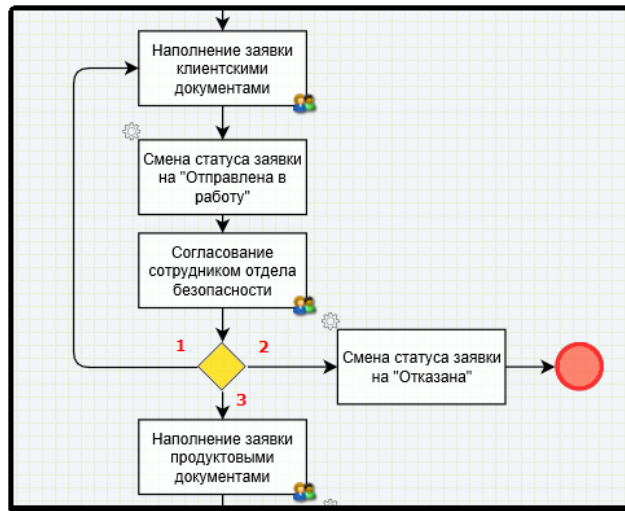


Рисунок 5.5-3. Пример использования исключаящего шлюза

На рисунке 5.5-3 после выполнения процедуры «Согласование сотрудниками отдела безопасности» дальнейшее выполнение процесса может продолжиться **только по одному из трёх потоков**, исходящих из шлюза:

- если условие 1 верно, то исполнение перейдет к процедуре «Наполнение заявки клиентскими документами»;
- если условие 2 верно, то исполнение перейдет к процедуре «Смена статуса заявки на «Отказана»»;
- если условие 3 верно, то исполнение перейдет к процедуре «Наполнение заявки продуктовыми документами».

В этом примере администратор точно знает, что на развилке будет обязательно выполнено хотя бы одно из условий 1, 2 или 3. В маршрутах, в которых на развилке может быть ситуация, когда ни одно из условий не выполнено, нужно предусматривать еще один выход из шлюза (безусловный).


Условия для переходов можно указывать **не только исходя из выходных параметров процедуры** или реквизитов объекта, но и **в зависимости от кнопок** в сеансе процедуры, расположенной в маршруте **до** шлюза.


Например, на рисунках 5.5-4,5 показана часть бизнес-процесса со шлюзом, условия переходов в котором осуществляются **в зависимости от нажатой пользователем кнопки**. Процедура «Ответ на предложения» настроена таким образом, что пользователь может нажать на одну из маршрутных кнопок:

- если будет нажата кнопка «Согласовать ответ в подразделении», то исполнение перейдет к процедуре «Согласование ответа руководителем подразделения Заказчика» (1);
- если будет нажата кнопка «Ответить ГУИТ», то исполнение перейдет к автоматической процедуре «Статус «Реализация предложений»» (2).

Рисунок 5.5-4. Пример настройки переходов с помощью маршрутных кнопок

Рисунок 5.5-5. Выбор кнопки для перехода

Колонка «Переход по умолчанию» (со значком ) служит для установки перехода, по которому система будет продвигать объект в случае пропуска точки, стоящей непосредственно до шлюза.

 **Параллельный шлюз** (AND, «И») используется для ветвления процесса на несколько альтернативных потоков, которые исполняются **параллельно**.

На рисунке 5.5-6 параллельный шлюз используется для создания параллельных веток выполнения процесса: после выполнения процедуры «Согласование заявки инициатором» запустится **одновременное параллельное** выполнение **обеих** процедур «Согласование юридическим отделом» и «Согласование финансовым отделом».

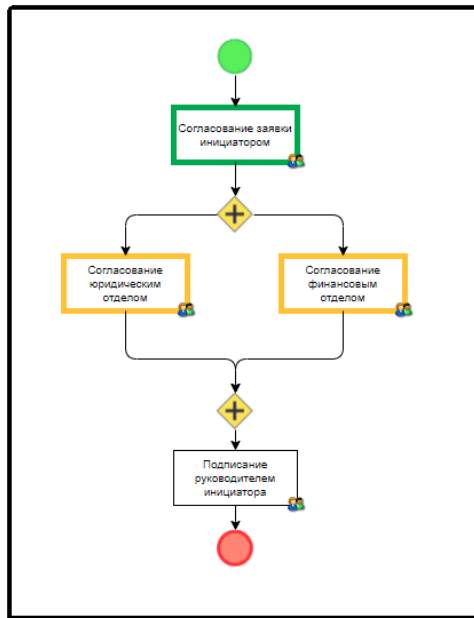


Рисунок 5.5-6. Пример использования параллельного шлюза в процессе

Выполнение процедуры, расположенной после слияния двух потоков, запустится только тогда, когда **выполнятся обе ветки процесса**. Никаких дополнительных условий переходов на параллельном шлюзе указывать не нужно.

Если в месте слияния двух веток процесса указать исключаящий шлюз, то для запуска процедуры, следующей за ним, достаточно выполнения **хотя бы одной** из веток (рис.5.5-7).

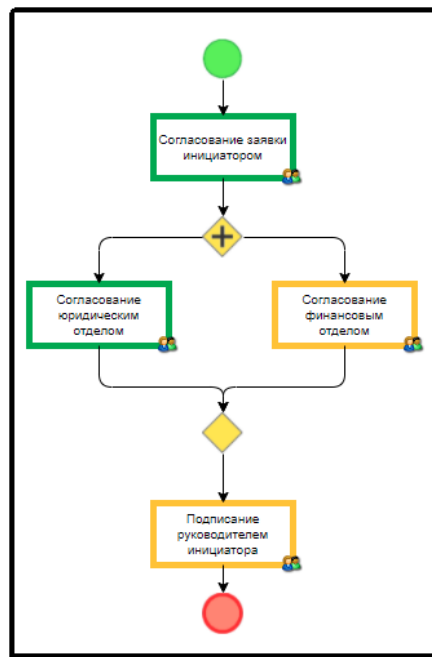



Рисунок 5.5-7. Пример слияния двух веток процесса при выполнении хотя бы одной из них

 **Включающий шлюз** (OR, "ИЛИ") используется для ветвления потока управления на несколько потоков, когда выполнение процесса зависит от выполнения условий. При этом каждое из указанных условий является независимым, и дальнейшее выполнение процесса может продолжиться **сразу по нескольким потокам управления**, если условия будут выполнены. Среди переходов, исходящих из включающего шлюза, допускается использование перехода по умолчанию (без указания условия перехода): если ни одно из условий не выполняется, дальнейшее выполнение процесса продолжится по переходу по умолчанию.



Работа с этим логическим оператором аналогична работе с логическим оператором «Исключающий шлюз».

### Макетирование возвратов

Отдельно стоит описать процесс макетирования процессов, в которых в параллельных точках может быть осуществлен возврат. При моделировании процесса возвратную дугу после шлюза изображать не нужно.

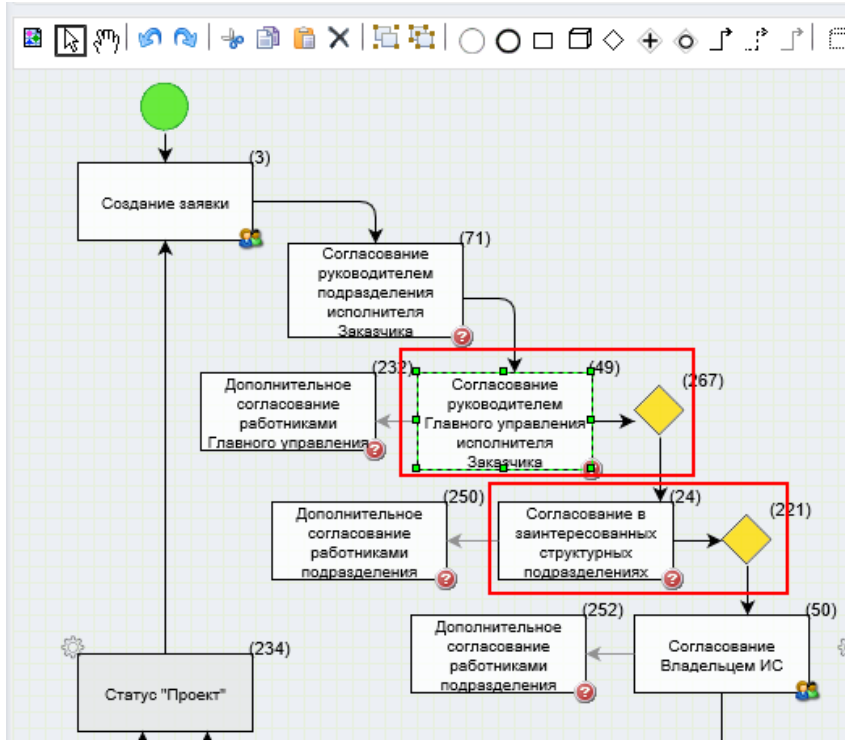


Рисунок 5.5-8. Пример параллельных точек без изображения возвратных дуг

После параллельной точки обязателен шлюз, который объединит все решения, принятые участниками. Переходы для возвратных точек автоматически добавляются в условие без рисования возвратной дуги. Кроме этого, в секции «Переходы» необходимо установить возвратный переход первым в списке с помощью кнопок . В этом случае этот переход будет считаться приоритетным, по нему пойдет процесс, если хотя бы один из участников отклонит объект.

Дизайнер процессов

Список процессов > Процесс "Обработка заявок на создание (изменение) информационных систем"

Изменения не сохранены!

Опубликован

Описание процесса | Описание точки

Идентификатор: 267

Наименование: Для продолжения согласования нужно нажать "Согласовать"

Описание:

Переходы

Точка назначения	Способ	Описание
Согласование в заинтересованных структурных подразделениях (24)	Кнопка	Согласовать
Создание заявки (3)	Кнопка	Вернуть

Рисунок 5.5-9. Возвратный переход

При этом на кнопке, которая реализует возврат, должно быть указано:  
- Движение по маршруту – Назад;

- Точка, в которую может быть выполнен возврат;
- Возвратный комментарий и его обязательность (при необходимости).

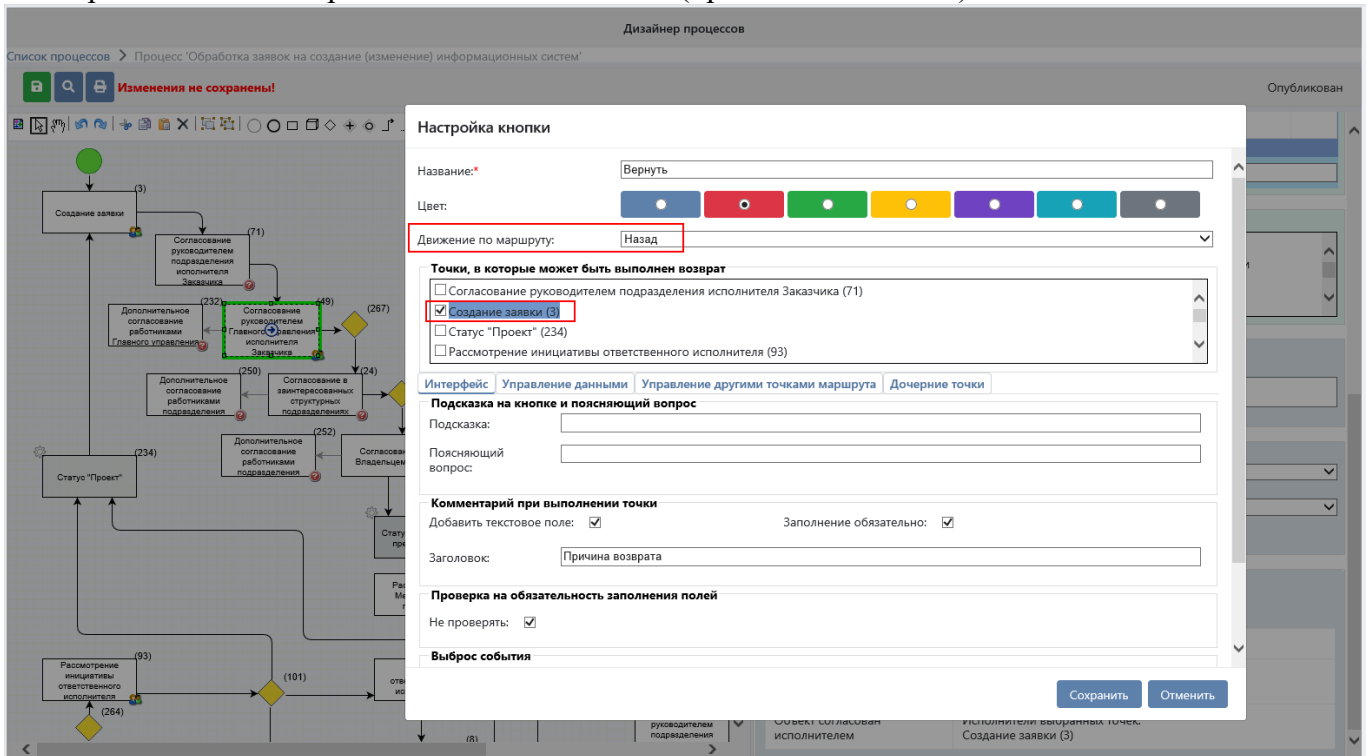


Рисунок 5.5-10. Настройки возвратной кнопки

## 5.6. Использование подпроцессов

ISIDA BPM даёт возможность внутри бизнес-процессов использовать **подпроцессы** – самостоятельные небольшие процессы, имеющие начальную и конечную точку и свою логику обработки объекта. Используются для того, чтобы упростить описание большого, так называемого **магистрального**, процесса, сделать декомпозицию задачи, разбить процесс на отдельные самостоятельные подпроцессы.

Для того, чтобы использовать подпроцесс, необходимо сначала создать отдельный процесс, сохранить его, указать для него объект. Затем опубликовать (рис.5.6-1).

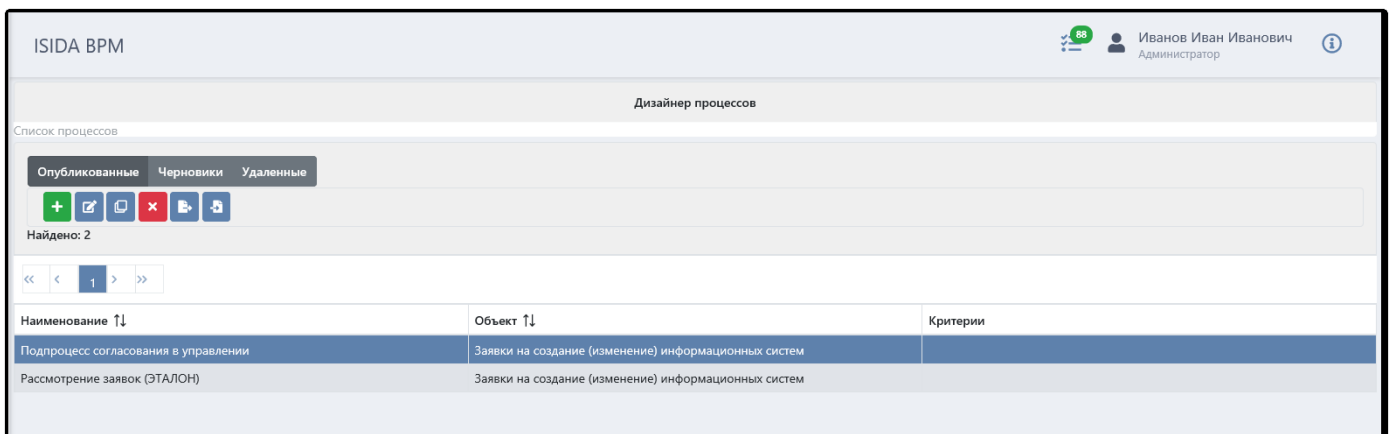



Рисунок 5.6-1. Создание подпроцесса

После этого в магистральном процессе нужно добавить объект «Подпроцесс» (для этого служит специальный элемент ) и настроить для него следующие свойства:

**Наименование** – выбор опубликованного подпроцесса;

**Идентификатор параметра, хранящий идентификатор объекта** – идентификатор, по которому будет установлено однозначное соответствие между объектом, который идет по магистральному процессу и объектом, для которого создается подпроцесс.

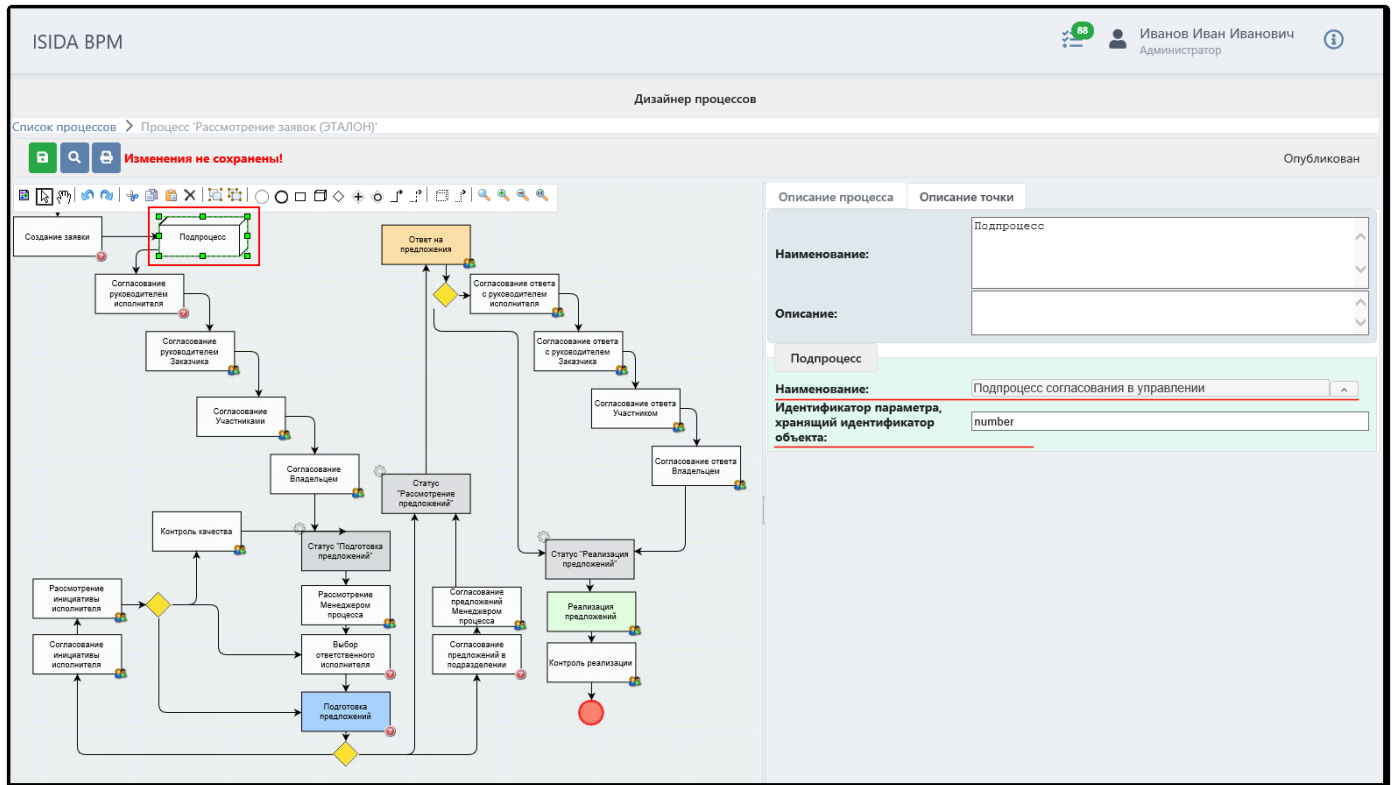


Рисунок 5.6-2. Выбор и настройка свойств подпроцесса

Например, если по магистральному процессу движется объект «Заявка», и этот же объект «Заявка» должен дальше двигаться по подпроцессу. В этом случае в настройках подпроцесса нужно указать идентификатор, однозначно определяющий заявку, например ее номер - *number* (рис.5.6-2). В процедуре, расположенной до начала подпроцесса, в **выходных параметрах** кнопки следует указать этот же идентификатор объекта (рис. 5.6-3), например:

*number=#object['number']*

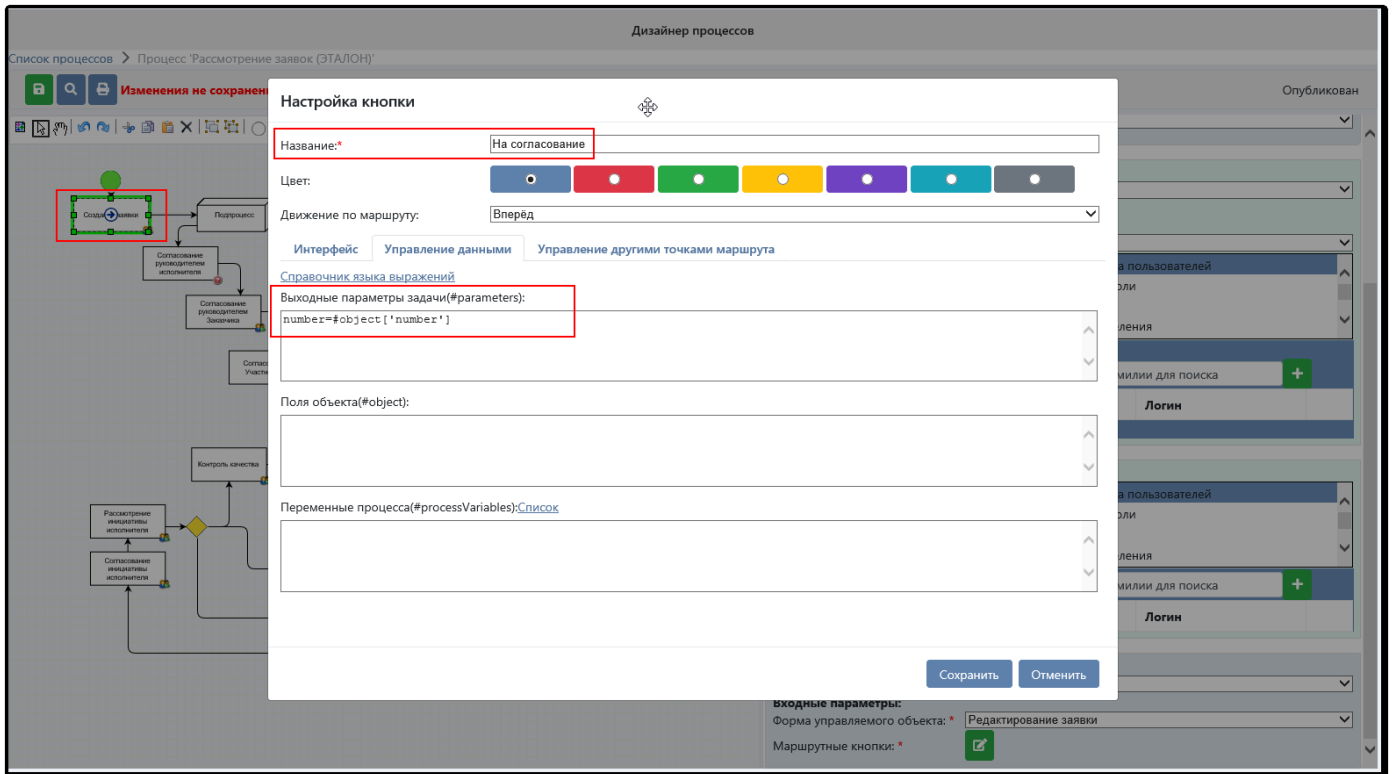


Рисунок 5.6-3. Настройка параметров кнопки до подпроцесса

При описании выходного параметра можно пользоваться [справочником языка выражений](#). Таким образом будет однозначно определено соответствие, означающее, что по магистральному процессу и подпроцессу идет один и тот же объект (в нашем примере – «Заявка»).

## 5.7. Менеджер задач

Важным функциональным блоком в настройках точки маршрута является «Задача» (в более ранних версиях BPM – «Процедура»). В этом блоке администратор настраивает весь набор функциональных возможностей системы, которые будут доступны для работы пользователя при исполнении точки на маршруте.

Перед тем, как задача станет доступной для выбора в настройках точки, она должна пройти специальную процедуру регистрации, для чего служит сеанс «Менеджер задач» (рис.5.7-1). Можно сказать, что Менеджер задач управляет каталогом имеющихся задач и их конфигурацией.

Менеджер процедур

Поиск процедур

Наименование:

Идентификатор:

Тип реализации:

Состояние:

Найдено: 16

Состояние	Наименование	Идентификатор	Тип реализации
✔	Передача пакета документов по электронной почте	transferdocuments	Пользовательская
✔	Визирование документов	documentsight	Пользовательская
✔	Получение пакета документов по электронной почте	receivingdocuments	Пользовательская
✔	Смена статуса	changeStatusTask	Автоматическая
✔	Изменение параметров заявки	edit	Пользовательская
✔	Голосование	voting	Пользовательская
✔	Добавление документов	addDocuments	Пользовательская
✔	Автоматическое создание досье	autoCreateDossier	Автоматическая
✔	Формирование	create	Пользовательская
✔	Согласование	agreement	Пользовательская
✔	Автоматическое обновление досье	autoUpdateDossier	Автоматическая
✔	Принятие решения	decisionMaking	Пользовательская
✔	Изменение полей объекта	bpmChangeObjectFields	Автоматическая
✔	Выполнение анализа	estimate	Пользовательская
✔	Установка полномочий	settingAuthority	Пользовательская
✔	Автоматическое связывание документов	autoLinkingDocuments	Автоматическая

Очистить Поиск

Рисунок 5.7-1. Сеанс «Менеджер процедур»

Каждая задача характеризуется следующими свойствами:

- **Наименование** – русскоязычное название задачи, в котором по возможности отражается заложенная логика;
- **Идентификатор** – англоязычный уникальный идентификатор, заданный разработчиком;
- **Тип реализации** – пользовательская (задача предназначена для обработки пользователем) или автоматическая (задача будет исполнена самой системой путем запуска указанного программного компонента, такая задача будет работать «за кадром» по заложенному в ней сценарию, например, изменять реквизит объекта или производить какую-либо проверку);
- **Состояние** – зарегистрирована ✔, не зарегистрирована ☹, отсутствует программная реализация ✖. Зарегистрированные задачи можно использовать при описании процесса.

Задачи создаются командой разработчиков, а затем публикуются в ISIDA BPM. В конкретной задаче закладывается алгоритм работы с объектом: действия, которые можно выполнить с полями объекта, проверка данных или любые другие операции, необходимые конкретному бизнес-процессу. Логика обработки объекта может быть довольно сложной, все зависит от цели работы данной задачи.

Каждая задача характеризуется входными и выходными параметрами. **Входными параметрами** являются опции, настраиваемые при описании процесса (конфигурация задачи): например, кнопки, которые будут показаны в сеансе пользователю, реквизиты объекта, которые он сможет менять, названия диалоговых окон, значения для выбора из выпадающих списков, параметры, которые в дальнейшем может использовать процедура в алгоритме обработки и т.д. **Выходными параметрами** принято называть значения, которые принимают переменные после работы процедуры. Выходные параметры чаще всего используются для дальнейшего управления движением объекта по маршруту (выбор ветки при развилке) и фиксации результатов выполнения задачи.

Для примера рассмотрим пользовательскую задачу «Изменение параметров заявки» (рис.5.7-2).

Менеджер процедур									
Поиск процедур * Просмотр процедуры									
Наименование:		Изменение параметров заявки							
Идентификатор:		edit							
Тип реализации:		Пользовательская							
Множественность исполнителей:		<input type="checkbox"/>							
Входных параметров: 7									
№ п/п	Наименование	Идентификатор	Компонент	Значение по умолчанию	Обязательный для заполнения	Видимость	Идентификатор реализации	Связан с параметрами	Комментарий
1	Реквизиты заявки	taskRequisites	Список		Нет	Да	taskRequisiteParamet	taskRequiredRequisit	
2	Обязательность документов	taskRequiredDocume	Список		Нет	Нет			
3	Кнопки	buttons	Список	Отправить на согласование; Подтвердить; Отправить на проверку; Отказ; Отправка на доработку	Да	Да			
4	Документы	documents	Дерево		Нет	Да	taskRequisiteParamet	taskRequiredDocume	
5	Реквизиты условий сделки	dealConditionRequisi	Список		Нет	Да	dealConditionRequisi	dealConditionRequire	
6	Обязательность реквизитов условий сделки	dealConditionRequire	Список		Нет	Нет			
7	Обязательность реквизитов заявки	taskRequiredRequisit	Список		Нет	Нет			
Параметров завершения: 1									
№ п/п	Наименование	Идентификатор	Компонент	Значение по умолчанию	Комментарий				
1	Флаг	mark	Список	Отправить на согласование; Подтвердить; Отправить на проверку; Отказ; Отправка на доработку					

Рисунок 5.7-2. Просмотр параметров задачи «Изменение параметров заявки»

Задача в числе прочих имеет такие входные параметры:

- **Кнопки** – запрограммированный список кнопок, которые можно предложить пользователю в конкретной точке маршрута;
- **Реквизиты заявки** – список полей, которые можно предложить пользователю для изменения в конкретной точке маршрута.

В Дизайнере процессов при назначении этой задачи на точку маршруту администратор должен заполнить входные параметры задачи (рис.5.7-3), т.е. определиться с кнопками (выбрать из допустимого списка) и реквизитами заявки.

Дизайнер процессов

Список процессов › Процесс 'Кредитование и Рефинансирование ЮЛ'

Опубликован

**Процедура**

Наименование: Изменение параметров заявки

Входные параметры:

Реквизиты заявки (taskRequisites):

Идентификатор	Наименование	Обязательность реквизитов заявки
interestRate	Процентная ставка	<input type="checkbox"/> <input checked="" type="checkbox"/>
loanTerms	Срок кредитования	<input type="checkbox"/> <input checked="" type="checkbox"/>
sum	Сумма	<input type="checkbox"/> <input checked="" type="checkbox"/>

Кнопки(buttons):

- Отправить на согласование
- Подтвердить
- Отправить на проверку
- Отказ
- Отправка на доработку

Документы(documents):

Идентификатор	Наименование	Обязательность документов

Реквизиты условий сделки (dealConditionRequisites):

Идентификатор	Наименование	Обязательность реквизитов условий сделки

Параметры завершения:

Идентификатор	Наименование
SENDTOWORK	Отправить на согласование
CONFIRM	Подтвердить
SENDTOCHECK	Отправить на проверку
REFUSAL	Отказ
SENDTOREVISION	Отправка на доработку

Флаг(mark):

Идентификатор	Наименование
SENDTOWORK	Отправить на согласование
CONFIRM	Подтвердить
SENDTOCHECK	Отправить на проверку
REFUSAL	Отказ
SENDTOREVISION	Отправка на доработку

Рисунок 5.7-3. Настройка задачи в Дизайнере процессов

В данном примере выходным параметром является переменная *mark*, которая может принимать одно из значений: REFUSAL, CONFIRM и т.д. (рис.5.7-3). Т.к. после процедуры «Обработка заявки кредитным инспектором» в процессе следует разветвление, логично, что ее выходной параметр влияет на дальнейший выбор ветки. На каждой ветке после развилки стоит условие, зависящее от значения *mark* (рис.5.7-4).

Дизайнер процессов

Список процессов › Процесс 'Кредитование и Рефинансирование ЮЛ'

Опубликован

Изменения не сохранены!

**Описание процесса** | **Описание точки**

Наименование:

Описание:

**Условие**

Смена статуса заявки на "Отказана": 1  
Язык выражений parameters["mark"] == 'REFUSAL'

Создание досье: 2  
Язык выражений parameters["mark"] == 'CONFIRM'

Рисунок 5.7-4. Настройка условий для ветвлений в маршруте

Рассмотрим пример одной из автоматических задач – «Смена статуса» (рис. 5.7-5). Входные параметры такой задачи содержат только список возможных текстовых значений статусов.

Менеджер процедур									
Поиск процедур * Просмотр процедуры									
Наименование:		Смена статуса							
Идентификатор:		changeStatusTask							
Тип реализации:		Автоматическая							
Множественность исполнителей:		<input type="checkbox"/>							
Входных параметров: 1									
№ п/п	Наименование	Идентификатор	Компонент	Значение по умолчанию	Обязательный для заполнения	Видимость	Идентификатор реализации	Связан с параметрами	Комментарий
1	Статусы	statuses	Список	Формируется; Отправлена в работу; Действующая; Закрыта; Отказана; Аннулирована; Документы подписаны; Возвращена; Нет доступных предложений; Не удовлетворяет критериям банка	Нет	Да			
Параметров завершения: 0									
№ п/п	Наименование	Идентификатор	Компонент	Значение по умолчанию	Комментарий				

Рисунок 5.7-5. Просмотр параметров задачи «Смена статуса»

При конструировании процесса в дизайнера процедуру можно использовать **сколько угодно раз**, но при этом в **разных процедурах указывать разные текстовые значения статусов**, выбирая из предложенного списка (рис.5.7-6). В алгоритме процедуры заложено, что в объекте (заявке) обновляется ее реквизит (status) соответственно тому значению, которое было указано на точке процесса.

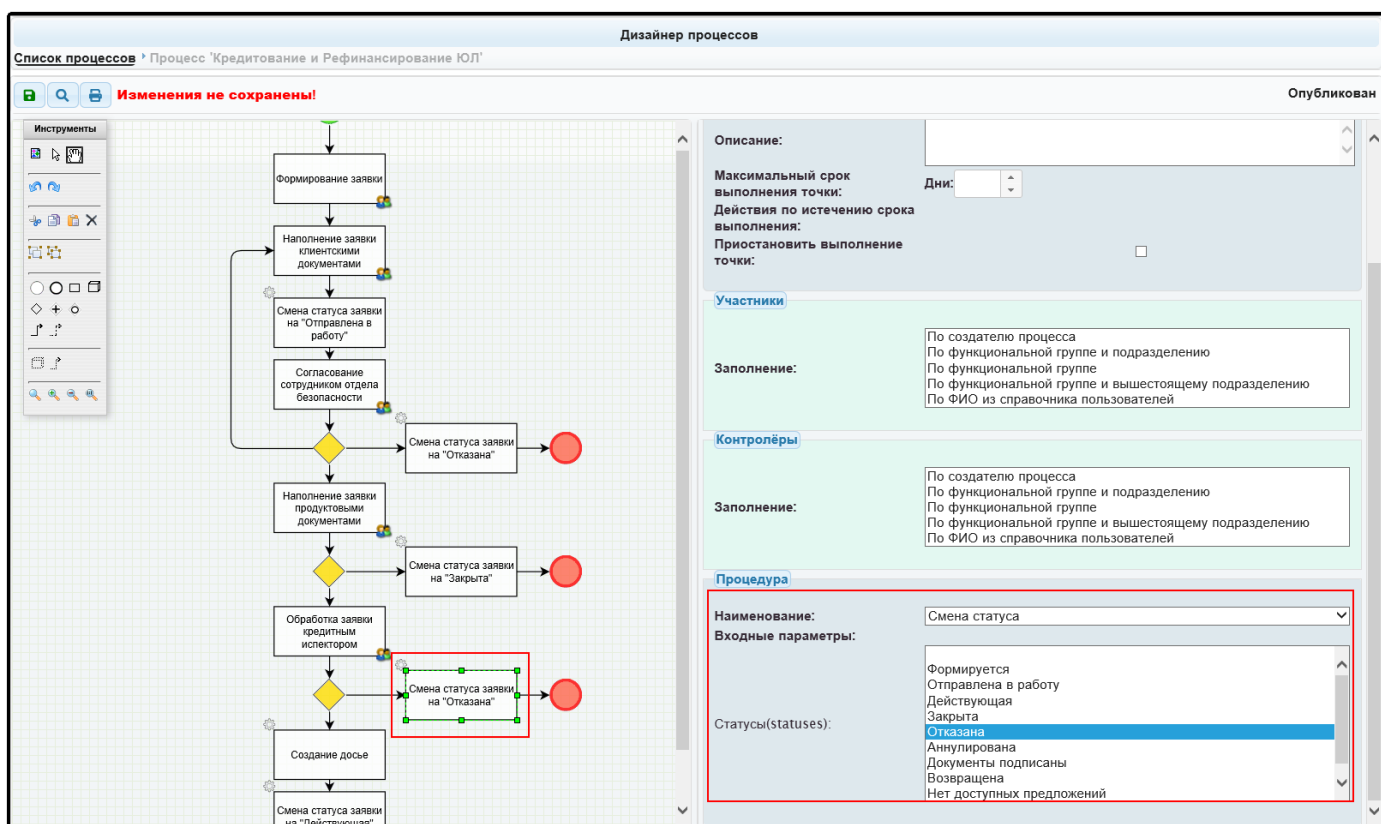


Рисунок 5.7-6. Настройка свойств процедуры

В стандартной комплектации ISIDA BPM предусмотрено две задачи, которые автоматически проходят процедуру регистрации:

- «Универсальная задача» (genericTask),



- задача «Изменение полей объекта» (bpmChangeObjectFields).

Их параметры и функциональность описаны ниже. Остальные задачи, разрабатываются с соблюдением программного интерфейса ISIDA BPM и регистрируются в Менеджере задач вручную.

## 5.8. Универсальная задача genericTask

«Универсальная задача» (*genericTask*) является программным компонентом ISIDA BPM и реализует базовый и достаточно универсальный набор действий, с помощью которого участник бизнес-процесса может обработать управляемый объект в точке маршрута. С помощью различных настроек поведение и внешний вид задачи может быть изменен таким образом, чтобы адаптировать её к контексту бизнес-процесса.

Универсальная задача может работать с любым управляемым объектом. Формы, которые позволяют создавать, просматривать и редактировать объект, заранее создаются командой разработчиков ISIDA BPM и по специальной технологии встраиваются в интерфейс задачи. Формы могут отличаться по внешнему виду и по реализуемым функциям: например, одна форма позволяет редактировать определенные поля объекта, другая позволяет загружать дополнительные файлы, третья позволяет только просматривать объект. Список всех разработанных форм прописывается в конфигурационном файле *application.properties* и после этого автоматически предлагается для выбора в настройках точки в Дизайнере процессов (рис.5.8-1 и 5.8-2).

The screenshot shows a configuration window titled "Задача" (Task). It contains the following fields:

- Наименование:** A dropdown menu with "Универсальная задача" (Generic task) selected.
- Входные параметры:** A section header.
- Форма управляемого объекта: \*** A dropdown menu with "Редактирование заявки" (Request editing) selected.
- Маршрутные кнопки: \*** A green button with a white document icon.

Рисунок 5.8-1. Выбор формы управляемого объекта

The screenshot shows the same configuration window as Figure 5.8-1, but with a dropdown menu open for "Форма управляемого объекта: \*". The menu lists the following options:

- Универсальная задача
- Редактирование заявки
- Подготовка предложений
- Ответ на предложения
- Реализация предложений
- Просмотр заявки

Рисунок 5.8-2. Выбор формы управляемого объекта для универсальной задачи

Кроме выбора формы, в дизайнере можно настроить маршрутные кнопки, которые будут предложены пользователю при обработке им данной задачи. Работа с ними описана в следующих разделах.

### 5.8.1. Общая структура интерфейса универсальной задачи

Рассмотрим общую структуру интерфейса универсальной задачи, предоставляемую пользователю (рис. 5.8-3).

Заявки на создание (изменение) информации о процессе

Рассмотрение заявок

Информация о процессе

Заявки на создание (изменение)

Наименование процесса

Наименование и описание точки маршрута

Вкладка с описанием процесса (таблица и граф)

Создание заявки

Ленточкина Ирина Иосифовна Причина возврата: Прикрепите макет должностной инструкции

(Необходимо заполнить сведения и направить заявку в ГУИТ)

Заявки на создание (изменение) информационных систем

Сопроводительный комментарий

Номер

16

Дата \*

17.05.2021

Статус

Проект

Приоритет \*

Обычный

Информационная система

Программный комплекс "Персонал"

Подразделение

Главное управление персонала

Ответственный работник

Костюковская Анна Николаевна

Краткое описание требований \*

Необходимо разработать шаблон для должностной инструкции работника.

Обоснование необходимости

Согласно Плану цифровой трансформации от 01.02.2020 № 10-10/123 процесс подготовки и согласования инструкции подлежит переводу в электронный вид (пункт 84).

Удалить

Маршрутные кнопки

Направить в ГУИТ

Согласовать в подразделении

Рисунок 5.8-3. Интерфейс универсальной задачи

Описание компонентов:

- 1) **Наименование процесса:** общее наименование бизнес-процесса, задается в «Дизайнере процессов».
- 2) **Наименование и описание точки маршрута:** текстовая характеристика точки, поясняющая ее задачу, также задается в «Дизайнере процессов». Наименование является обязательной характеристикой, в то время как описание опционально. Если описание указано, то оно отображается второй строкой в скобках.
- 3) **Описание процесса (таблица и граф):** хронологическое описание этапов процесса в виде таблицы или визуальное представление процесса в виде графа с отметками о пройденных точках.
- 4) **Сопроводительный комментарий:** поручение, введенное исполнителем предыдущей точки процесса. Отображается ФИО исполнителя и его комментарий. В этом же окне отображается срок исполнения задачи.
- 5) **Форма управляемого объекта:** отображает поля управляемого объекта, а также содержит панель функциональных кнопок по его созданию и редактированию (данные кнопки не влияют на движение объекта по маршруту). Как было сказано ранее, формы создаются разработчиками ISIDA BPM и поставляются вместе с дистрибутивами системы. Каждая точка маршрута может содержать свою форму управляемого объекта, таким образом могут быть разграничены полномочия по отображению сведений и права по редактированию объекта.
- 6) **Маршрутные кнопки:** набор действий, который предоставляется исполнителю задачи для выражения своего решения. Настраиваются в Дизайнере процессов.

## 5.8.2. Настройка универсальной задачи в Дизайнере процессов

Настройка универсальной задачи предполагает выбор формы управляемого объекта (рисунок 5.8-4) и настройку маршрутных кнопок (рисунок 5.8-6).

Рисунок 5.8-4. Выбор формы управляемого объекта для универсальной задачи

Выпадающий список «**Форма управляемого объекта**» содержит перечень визуальных форм, доступных в ISIDA BPM для конкретного типа объекта. Здесь администратору необходимо выбрать, какую именно форму управляемого объекта следует загрузить в данной точке маршрута. Все доступные формы заранее регистрируются в файле настроек *application.properties* в следующем формате:

```
bpm.flowcore.procedure.GenericTask.form.управляемый_объект=перечисление идентификаторов форм через запятую
```

```
bpm.flowcore.procedure.GenericTask.form.id_формы.управляемый_объект.url=url к форме в формате html  
bpm.flowcore.procedure.GenericTask.form.id_формы.управляемый_объект.name=наименование формы
```

и так для каждой формы, которая перечислена в первом параметре.

Например:

```
bpm.flowcore.procedure.GenericTask.form.itrequest=form1,form2,form3,form4,form5,form6  
bpm.flowcore.procedure.GenericTask.form.form1.itrequest.url=/studio-object-form-generator/ObjectEdit.xhtml?formId=form1  
bpm.flowcore.procedure.GenericTask.form.form1.itrequest.name=Редактирование заявки  
bpm.flowcore.procedure.GenericTask.form.form2.itrequest.url=/studio-object-form-generator/ObjectEdit.xhtml?formId=form2  
bpm.flowcore.procedure.GenericTask.form.form2.itrequest.name=Подготовка предложений  
bpm.flowcore.procedure.GenericTask.form.form3.itrequest.url=/studio-object-form-generator/ObjectEdit.xhtml?formId=form3  
bpm.flowcore.procedure.GenericTask.form.form3.itrequest.name=Ответ на предложения  
bpm.flowcore.procedure.GenericTask.form.form4.itrequest.url=/studio-object-form-generator/ObjectEdit.xhtml?formId=form4  
bpm.flowcore.procedure.GenericTask.form.form4.itrequest.name=Реализация предложений  
bpm.flowcore.procedure.GenericTask.form.form5.itrequest.url=/studio-object-form-generator/ObjectEdit.xhtml?formId=form5  
bpm.flowcore.procedure.GenericTask.form.form5.itrequest.name=Просмотр заявки  
bpm.flowcore.procedure.GenericTask.form.form6.itrequest.url=/studio-object-form-generator/ObjectEdit.xhtml?formId=form6  
bpm.flowcore.procedure.GenericTask.form.form6.itrequest.name=Редактирование заявки (видимость исполнителя)
```

Формы для объектов и электронных документов отличаются за счет специфики структуры объекта и ЭД. Работа с формами для ЭД описана в [разделе 5.9](#) данного руководства.

После того, как формы будут перечислены в файле настроек, они станут доступными для выбора в дизайнере процессов (рис. 5.8-5).

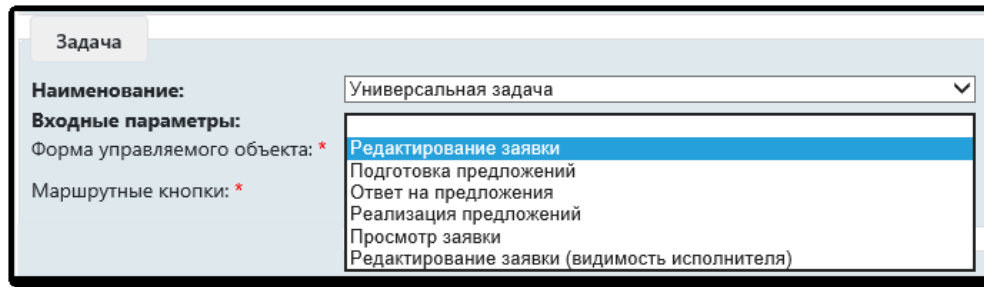


Рисунок 5.8-5. Выбор формы управляемого объекта для универсальной задачи

Раздел «**Маршрутные кнопки**» служит для установки состава, порядка следования и поведения кнопок, которые будут доступны исполнителю в этой точке маршрута (рис.5.8-6).

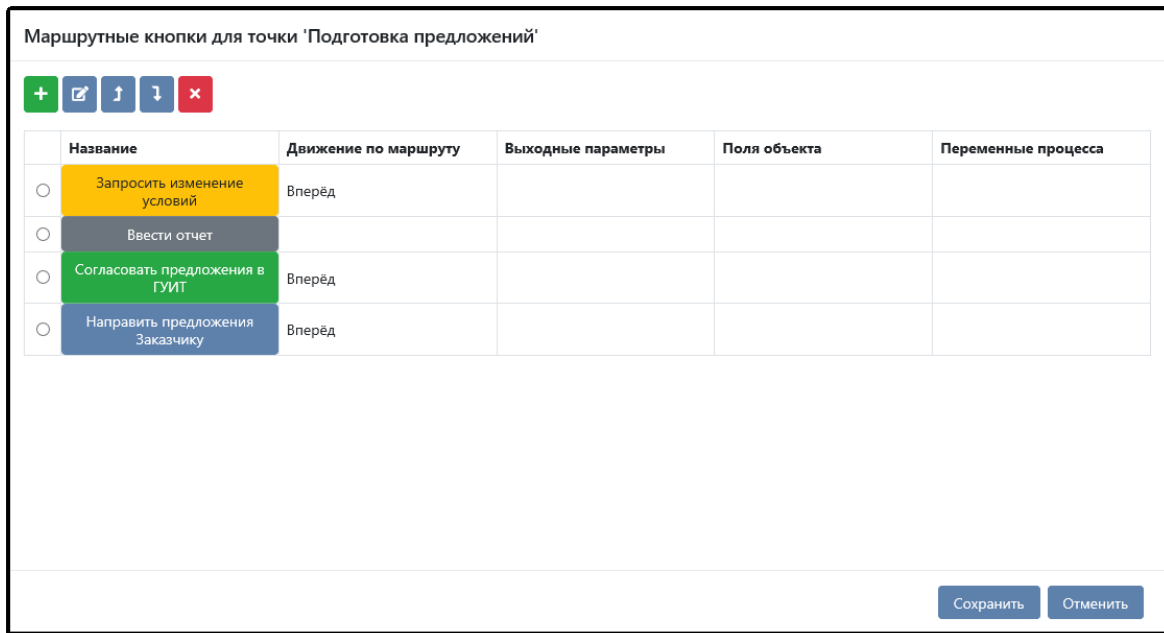


Рисунок 5.8-6. Список маршрутных кнопок

Порядок следования кнопок в данном диалоге совпадает с порядком следования в интерфейсе универсальной задачи. Кнопки, означающие движение назад, будут сгруппированы в левой части формы, остальные кнопки будут сгруппированы в правой части.

### 5.8.3. Настройка маршрутной кнопки для задачи

Диалог настройки отдельной маршрутной кнопки представлен на рисунке 5.8-7.

**Настройка кнопки**

Название:\*

Цвет:

Движение по маршруту:

**Интерфейс** | Управление данными | Управление другими точками маршрута

**Подсказка на кнопке и поясняющий вопрос**

Подсказка:

Поясняющий вопрос:

**Комментарий при выполнении точки**

Добавить текстовое поле:       Заполнение обязательно:

Заголовок:

**Проверка на обязательность заполнения полей**

Не проверять:

**Выброс события**

Событие:

**Изменение срока планового события**

Срок подготовки предложений

Рисунок 5.8-7. Настройка маршрутной кнопки

Для каждой маршрутной кнопки можно задать следующие параметры:

**Название:** надпись, которая будет видна исполнителю задачи.

**Цвет:** выбор цвета для придания кнопке акцента.

**Движение по маршруту:**

- *движение вперед:* означает движение по одному из допустимых переходов, исходящих из точки. Выбор перехода осуществляется в зависимости от заданных условий. Кнопки с движением «Вперед» располагаются в правой части панели маршрутных кнопок. По умолчанию им предлагается синий цвет.
- *движение назад:* означает «прыжок» в точку маршрута, пройденную ранее. Кнопки с движением «Назад» располагаются в левой части панели маршрутных кнопок. По умолчанию им предлагается красный цвет.
- *без движения:* означает, что при нажатии на кнопку объект не движется по маршруту, но могут производиться другие действия: ввод отчета, установка сроков на других точках и т.д. По умолчанию им предлагается синий цвет.

**Раздел «Интерфейс»** позволяет настроить свойства кнопки и внешний вид окна при выполнении задачи.

Секция «Подсказка на кнопке и поясняющий комментарий» позволяют настроить кнопку для понимания ее назначения (рис.5.8-8). Подсказка отобразится при наведении на кнопку, а вопрос будет задан пользователю при нажатии на нее.

**Подсказка на кнопке и поясняющий вопрос**

Подсказка:

Поясняющий вопрос:

Рисунок 5.8-8. Настройки в секции «Подсказка на кнопке и поясняющий комментарий»

Кроме этого, фраза, которая указана в качестве подсказки на кнопке, будет предлагаться в заголовке окна-диалога при обработке этой кнопки пользователем. Если подсказка не указана, окно будет иметь название, совпадающее с наименованием кнопки, например «Согласовать».

### Настройка кнопки

Название:\*

Цвет:

Движение по маршруту:

**Интерфейс** | Управление данными | Управление другими точками маршрута

**Подсказка на кнопке и поясняющий вопрос**

Подсказка:

Поясняющий вопрос:

Рисунок 5.8-9. Настройки в секции «Подсказка на кнопке и поясняющий комментарий»

Заявки на создание (изменение) информационных систем | Информация о процессе

Создание заявки

(Заполните поля Заявки, нажмите кнопку "Сохранить".  
Теперь можете подготавливать и исправлять заявку неограниченное количество времени.  
Когда будет готов окончательный вариант Заявки - запуск)

Номер (заполняется автоматически)  
60705

Статус (заполняется автоматически)  
Проект

Информационная система \*  
Обработка заявок на создание (изменение) информац

Ответственный работник, инициировавший заявку \*  
Ломоносова Ольга Ивановна

Краткое описание требований \*  
Тестовая заявка

Обоснование необходимости \*  
Тестовая заявка

Файлы заявки | Предложения по реализации | Ответ

Имя файла

Удалить

**Необходимо заполнить этапы маршрута и отправить заявку на согласование**

**Согласование руководителем подразделения исполнителя Заказчика**

Исполнители: \*      Количество: один и более      Режим исполнения: Любой из исполнителей

Сопроводительный комментарий:

**Согласование руководителем Главного управления исполнителя Заказчика**

Исполнители:      Количество: любое      Режим исполнения: Все исполнители (последовательно)

Андреев Павел Викторович X

**Согласование в заинтересованных структурных подразделениях**

Исполнители:      Количество: любое      Режим исполнения: Все исполнители (параллельно)

**Согласование Владелец ИС**

Исполнители:      Количество: любое      Режим исполнения: Любой из исполнителей

На согласование    X Отмена

На согласование

Рисунок 5.8-10. Диалог для управления другими точками маршрута

Секция «Комментарий при выполнении точки» позволяет настроить интерфейс окна при нажатии на кнопку: добавить поле для ввода комментария, указать заголовок этого окна и признак обязательности (рис.5.8-11).

Рисунок 5.8-11. Настройки в секции «Комментарий при выполнении точки»

При настройках, указанных на рисунке 5.8-9, внешний вид окна для обработки примет следующий вид (рис. 5.8-12).

Рисунок 5.8-12. Обработка точки с необязательным вводом комментария

Секция «Проверка на обязательность заполнения полей» позволяет снять признак проверки обязательности заполнения полей. В этом случае при обработке процедуры при нажатии этой маршрутной кнопки, система не будет проверять поля, обязательные для заполнения. Например, такая настройка актуальная для кнопок «Вернуть», «На доработку» и т.д.

Рисунок 5.8-13. Установка признака «Не проверять обязательность заполнения полей»

Секция «Выброс события» позволяет указать, какое прикладное событие следует выбросить при нажатии на эту точку. Подробно о выбросе событий описано [в разделе 5.11](#).

Рисунок 5.8-14. Установка выброса события на точке

Секция «Изменение срока планового события» позволяет установить, для какого из запланированных на данном маршруте событий нужно предоставить пользователю секцию смены срока.

Рисунок 5.8-15. Настройка свойства «Изменение срока планового события»

Если галочка установлена, в диалоге обработки точки пользователю будет предложена дополнительная секция для смены планового срока (рис.5.8-16).

Рисунок 5.8-16. Окно для обработки точки с учетом применения свойства «Изменение срока планового события»

Раздел «Управление данными» позволяет изменять:

- выходные параметры задачи;
- поля объекта;
- переменные процесса (рис.5.8-17).

Рисунок 5.8-17. Сеанс «Управление данными»

Например, нажатие кнопки может автоматически переводить управляемый объект в новый статус, т.е. поле *status* заполняется значением «Подготовка предложений», а другая кнопка переводит его в статус «Отменена». Эти выражения можно указать в настройках каждой кнопки и таким образом автоматически управлять статусом объекта. Для записи выражений используется синтаксис *Spring*



*Expression Language (SpEL)*. Для удобства дается подсказка с примерами, доступная по ссылке [«Справочник языка выражений»](#).

**Раздел «Управление другими точками маршрута»** позволяет указать действия, производимые над другими точками маршрута в момент обработки задачи пользователем (рис. 5.8-18).

**Настройка кнопки**

Название:\*

Цвет:

Движение по маршруту:

Интерфейс    Управление данными    Управление другими точками маршрута

Наименование точки	Назначить исполнителей	Установить срок	Сопроводительный комментарий		
			Добавить	Обязательно	Заголовок
Согласование в заинтересованных подразделениях (24)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
Согласование руководителем Главного управления Заказчика (49)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
Согласование Владелец (50)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
Выбор ответственного исполнителя (55)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
Согласование руководителем подразделения Заказчика (71)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>
руководителем Главного					

Рисунок 5.8-18. Сеанс управления другими точками маршрута

Сеанс позволяет изменять следующие настройки:

**Назначить исполнителей:** в некоторых случаях требуется, чтобы кнопка не только фиксировала решение исполнителя и двигала объект по маршруту, но также влияла на иные точки процесса, а именно, задавала в них исполнителей. В приведенном выше примере после нажатия на кнопку «На согласование» исполнителю будет предложен диалог (рисунок 5.8-19). Исполнителю задачи будет предложено выбрать исполнителей тех точек маршрута, которые заранее указал администратор.

Необходимо заполнить этапы маршрута и отправить заявку на согласование

Согласование руководителем подразделения исполнителя Заказчика

Исполнители: \* Количество: один и более Режим исполнения: Любой из исполнителей

Ленточкина Ирина Дмитриевна X

Сопроводительный комментарий:

Согласование руководителем Главного управления исполнителя Заказчика

Исполнители: Количество: любое Режим исполнения: Все исполнители (последовательно)

Должиков Константин Петрович X андреев

Андреев Павел Викторович

Согласование в заинтересованных структурных подразделениях

Исполнители: Количество: любое Режим исполнения: Все исполнители (параллельно)

Согласование Владельцем ИС

На согласование
X Отмена

Рисунок 5.8-19. Диалог для установки исполнителей в других точках

Если точки уже были заполнены исполнителями заранее (указаны администратором в дизайнера процессов), то ФИО исполнителей сразу отображаются в диалоге при обработке задачи.

**Установить срок:** возможность при обработке своей точки установить срок в других точках маршрута. При этом срок можно устанавливать с уточнением *времени* или без, в зависимости от того, указана опция в настройках системы.

Для рассмотрения

Исполнители: Количество: любое Режим исполнения: Все исполнители (параллельно)

Ленточкина Ирина Дмитриевна X

Срок: 31.12.2022 📅

Сопроводительный комментарий:

Прошу рассмотреть до конца года

Для рассмотрения
X Отмена

Рисунок 5.8-20. Диалог для установки срока исполнителя в другой точке

**Запросить сопроводительный комментарий и указать его обязательность:** указать необходимость ввести комментарий при исполнении задачи. В некоторых случаях комментарий должен быть обязательным (опция «Обязательно»). Если сопроводительный комментарий имеет особое наименование (например, «Резолюция»), то его можно задать в поле «Заголовок» (по умолчанию будет наименование «Комментарий»). Тогда диалог маршрутной кнопки будет иметь вид, указанный на рисунке 5.8-21.

**Выбор ответственного исполнителя для исполнения заявки**

Исполнители: \*      Количество: один и более      Режим исполнения: Все исполнители (параллельно)

Ломоносова Ольга Ивановна X

Резолюция:

Прошу рассмотреть документ до конца квартала

На исполнение      X Отмена

Рисунок 5.8-21. Диалог для ввода сопроводительного комментария для другого исполнителя

Если явно не указано наименование сопроводительного комментария, то по умолчанию в диалоге обработки точки будет отображаться фраза «Сопроводительный комментарий».

**Отображать наименование** этапа для следующих точек маршрута в диалоге при обработке своей точки (рис.5.8-22).

Дизайнер процессов

Список процессов > Процесс: «Обработка заявок на создание (изменение) информационных систем»

Изменения не сохранены

Настройка кнопки

Название: \* На согласование

Цвет:

Движение по маршруту: Вперёд

Интерфейс | Управление данными | Управление другими точками маршрута

Наименование точки	Отображать наименование	Назначить исполнителей	Установить срок	Сопроводительный комментарий		
				Добавить	Обязательно	Заголовок
Рассмотрение Менеджером процесса (7)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Подготовка предложений (8)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Ответ на предложения (9)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Реализация предложений (11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Контроль реализации в ГУИТ (12)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Согласование в заинтересованных структурных	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Сохранить      Отменить

Рисунок 5.8-22. Настройка опции «Отображать комментарий»

Если указано «Отображать наименование», то в диалоге для управления другими точками маршрута будут отображены названия этапов. Каждая из секций с точкой особо отделяется от остальных серой рамкой.

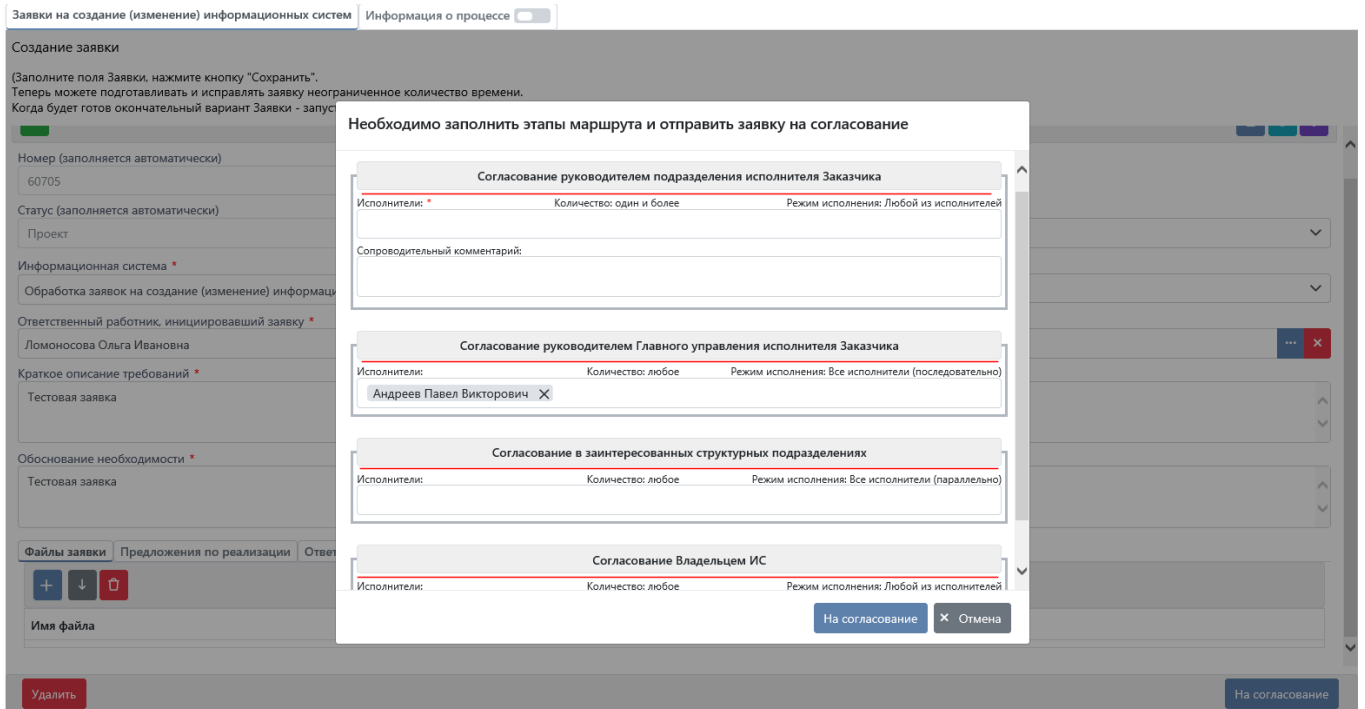


Рисунок 5.8-23. Диалог для управления другими точками маршрута

Если указано «Не отображать наименование», то в диалоге для управления другими точками маршрута названия этапов отображены не будут. Такую настройку лучше применять на точках, в которых есть управление только одной следующей точкой, например в точке «Выбор ответственного исполнителя» нужно выбрать исполнителя следующей точки, но пользователю не так важно наименование этого этапа.

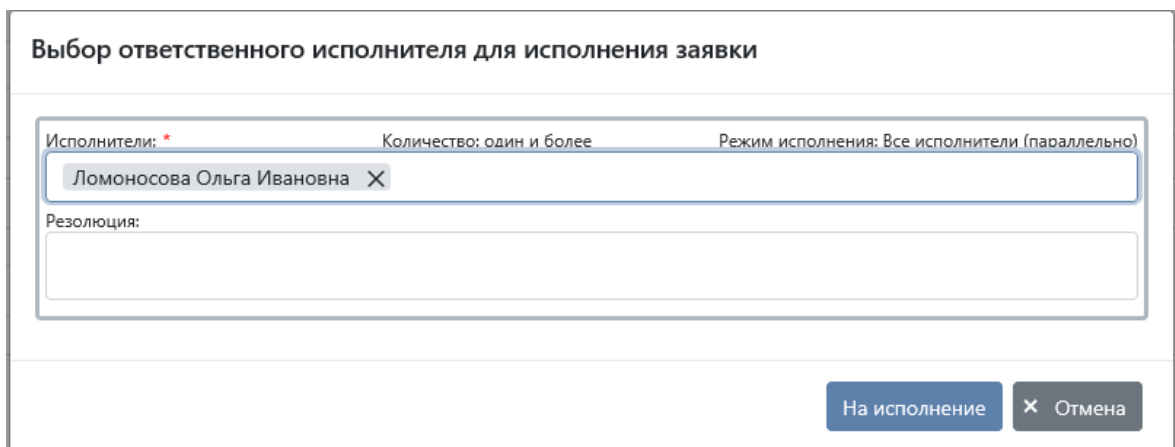


Рисунок 5.8-24. Диалог для управления точкой маршрута без указания названия этапа

**Раздел «Дочерние точки»** доступен для изменения только в случае, если от точки есть переход к дочерней точке (специальная связь между процедурами). Сеанс позволяет указать действия, производимые над дочерними точками маршрута в момент обработки задачи пользователем (рис. 5.8-25). Более подробно о работе с дочерними точками описано [в разделе 5.10](#).

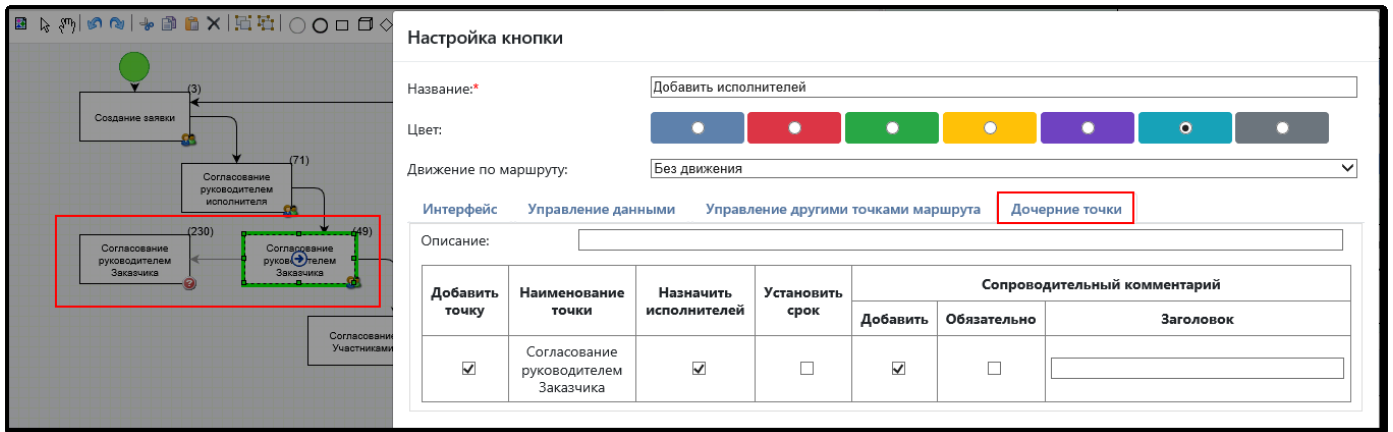


Рисунок 5.8-25. Настройка дочерних точек маршрута

#### 5.8.4. Настройка возвратных кнопок

Для **возвратных** кнопок (прыжок управляемого объекта в обратном направлении) дополнительно задаются точки, в которые может быть осуществлен возврат (рисунок 5.8-26). При этом также может быть запрошен сопроводительный текст, в том числе с особым заголовком (например, «Причина возврата»). В этом случае диалог маршрутной кнопки будет иметь вид, представленный на рисунке 5.8-26.

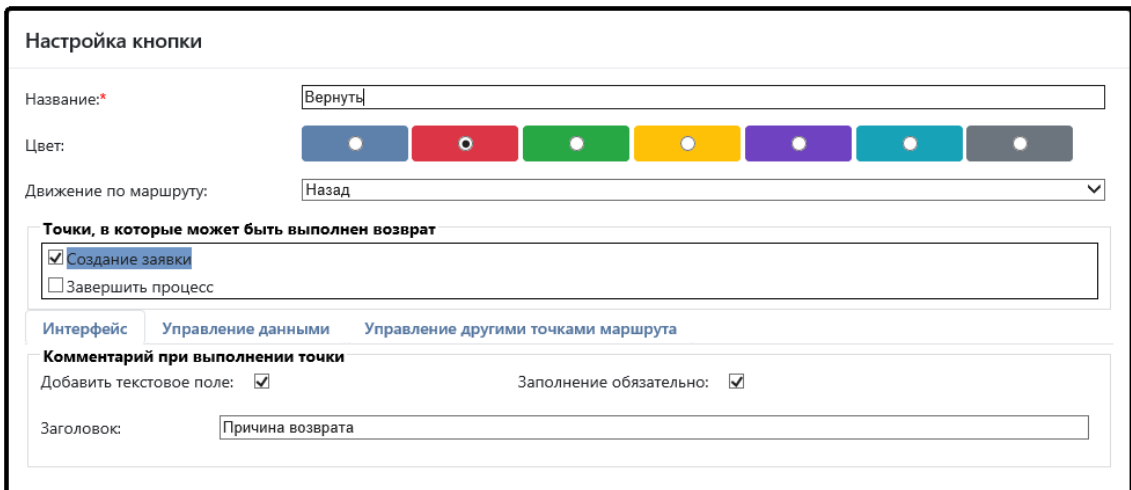


Рисунок 5.8-26. Настройка возвратной маршрутной кнопки



Рисунок 5.8-27. Возврат управляемого объекта с сопроводительным текстом (причиной возврата)

#### 5.8.5. Настройка кнопок без движения

Кнопки «Без движения» могут служить фиксации промежуточных действий, которые не требуют движения объекта. Например, на исполнение задачи «Реализация предложений» отводится 20

рабочих дней, но за это время от исполнителя ждут промежуточные результаты его работы. Тогда настройка такой кнопки может выглядеть так:

Рисунок 5.8-28. Настройка маршрутной кнопки «Без движения»

Диалог для пользователя на этой кнопке будет предлагать обязательное поле для ввода отчета и кнопку «Отчет». Все отчеты фиксируются как комментарий точки маршрута.

Рисунок 5.8-29. Диалог для ввода отчета на точке

### 5.8.6. Задача «Изменение полей объекта»

Задача «Изменение полей объекта» относится к категории автоматических и предназначена для установки простым полям управляемого объекта определенных значений, а также для работы с событиями. При настройке задачи Дизайне процессов предложит список допустимых полей (рис.5.8-30).

Рисунок 5.8-30. Выбор поля для задачи «Изменение полей объекта»

Задача


Наименование:

Входные параметры:

Поля объекта: \*

Идентификатор	Наименование	Значение
status	Статус	Подготовка предложений

Рисунок 5.8-31. Настройка задачи «Изменение полей объекта»

Следует заметить, что значение полей объекта можно устанавливать и с помощью «Универсальной задачи», настроив раздел «Поля объекта». В этом случае поле будет автоматически изменено при успешном завершении точки маршрута. Но если нужно особым образом выделить такие действия в визуальном представлении графа, можно использовать задачу «Изменение полей объекта». Так как это автоматическая задача, то в графе она отмечается особым знаком .

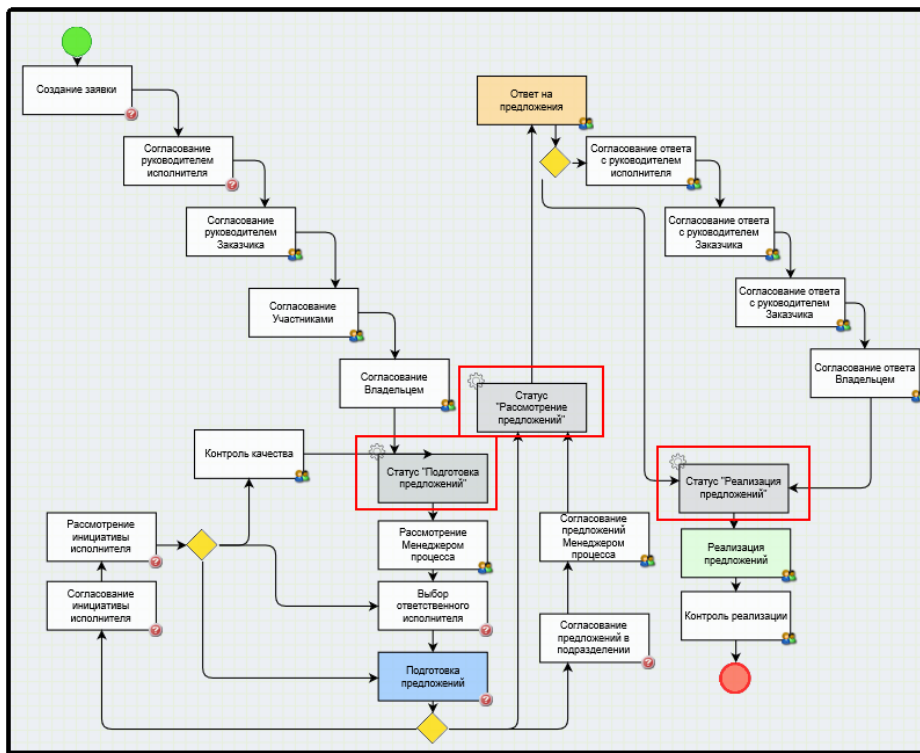


Рисунок 5.8-32. Просмотр задач «Изменение полей объекта» в графе процесса

Кроме изменения полей объекта, автоматическая задача может выбрасывать прикладные события, генерировать плановые события, аннулировать плановые события и хранить подписку на события.

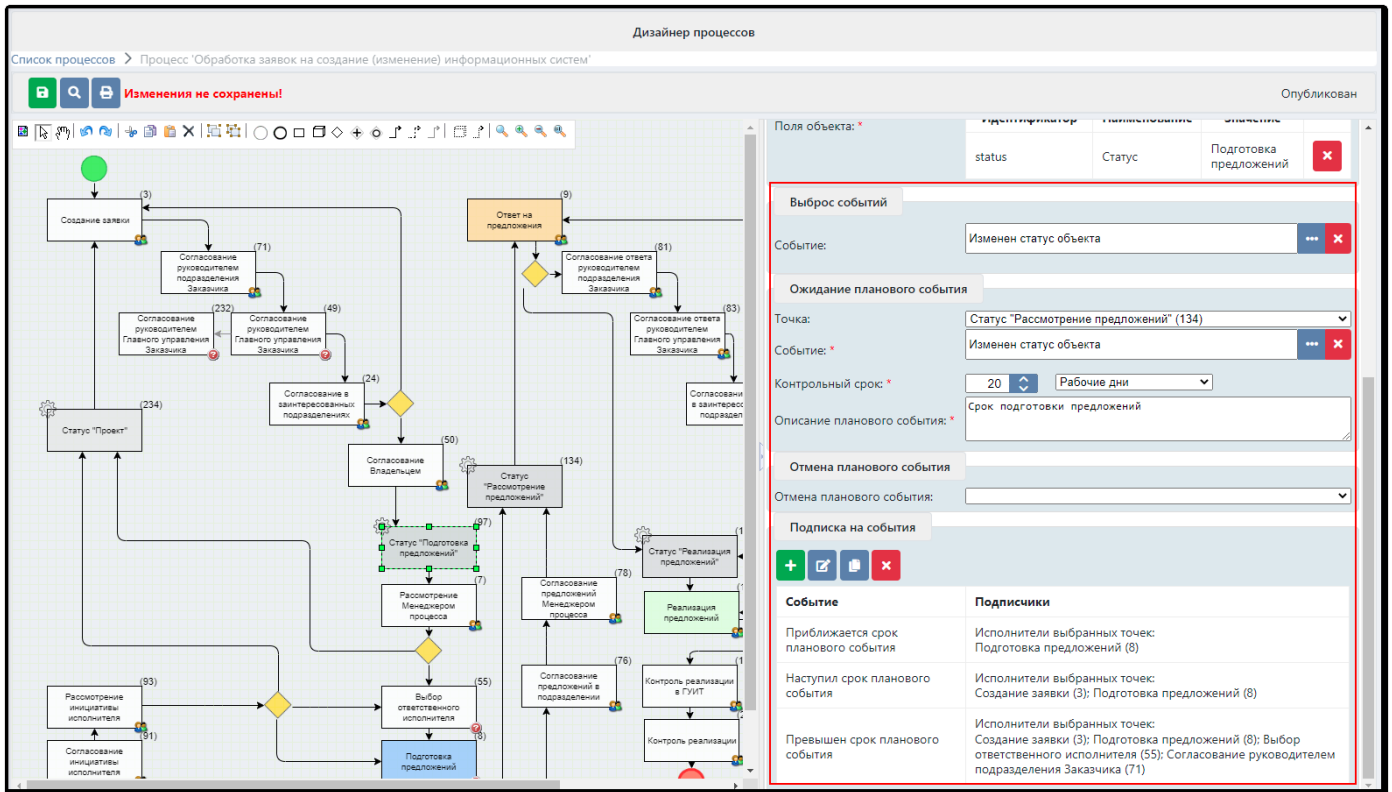


Рисунок 5.8-33. Описание секций задачи «Изменение полей объекта»

Подробно об этом описано [в разделе 5.11](#) и [разделе 5.12](#) данного руководства.

### 5.9. Настройка задач для электронных документов

Настройка процесса для электронного документа схожа с настройкой для объекта. В Дизайнере процессов необходимо сначала в качестве объекта выбрать «Электронный документ», а затем выбрать тип электронных документов, для которых разрабатывается маршрут.

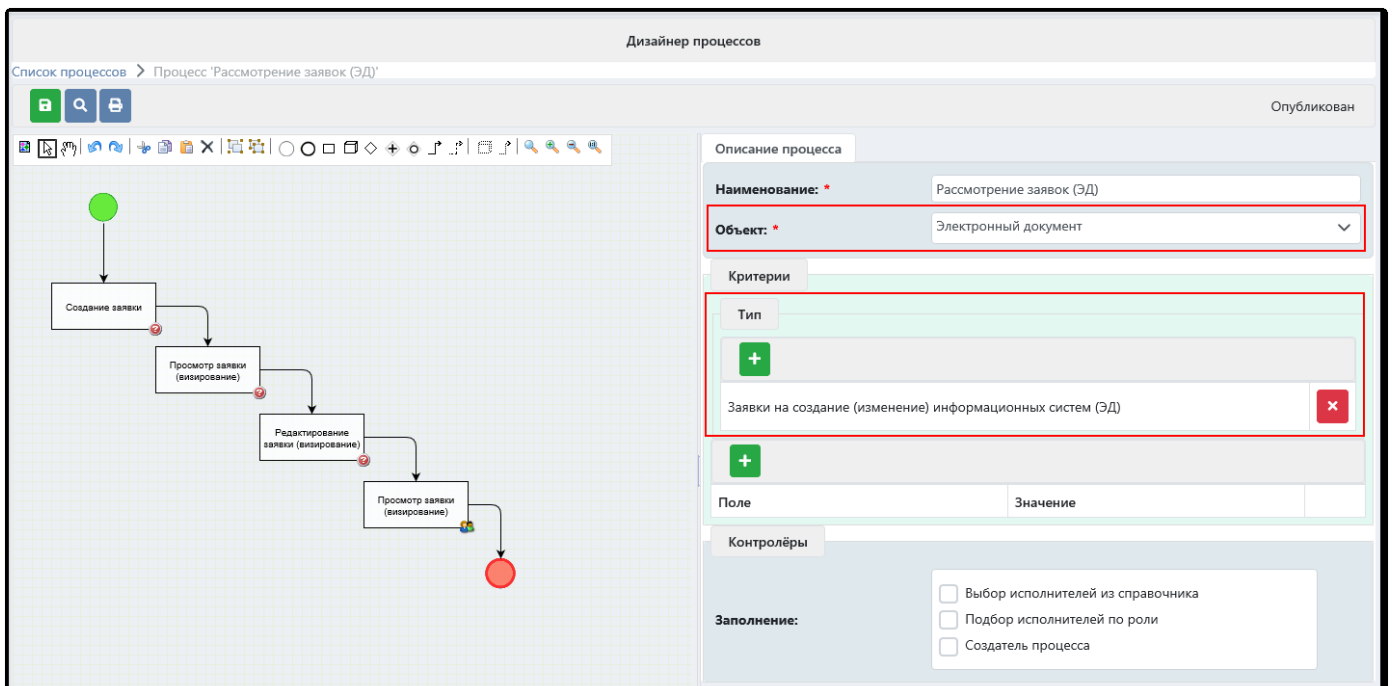


Рисунок 5.9-1. Настройка процесса для электронного документа

Как для создания/просмотра объектов, так и для электронных документов необходимы визуальные формы. Для электронных документов на данный момент имеется три вида форм:



- **Создание документа:** на форме предлагаются поля для заполнения реквизитов ЭД, которые отмечены флагом «Доступности для создания» в сеансе настройки типа, а также секция для добавления файлов ЭД;
- **Просмотр документа:** на форме отображаются значения реквизитов ЭД, которые отмечены флагом «Включения реквизита в полный просмотр ЭД», а также секция для просмотра файлов ЭД;
- **Редактирование документа:** форма аналогична форме просмотра документа, но с возможностью редактирования реквизитов ЭД, которые отмечены флагом «Доступности для изменения».

Как правило, в первой точке процесса указывается задача «Создание документа», в остальных – «Просмотр» или «Редактирование», в зависимости от сценария процесса и контекста задачи.

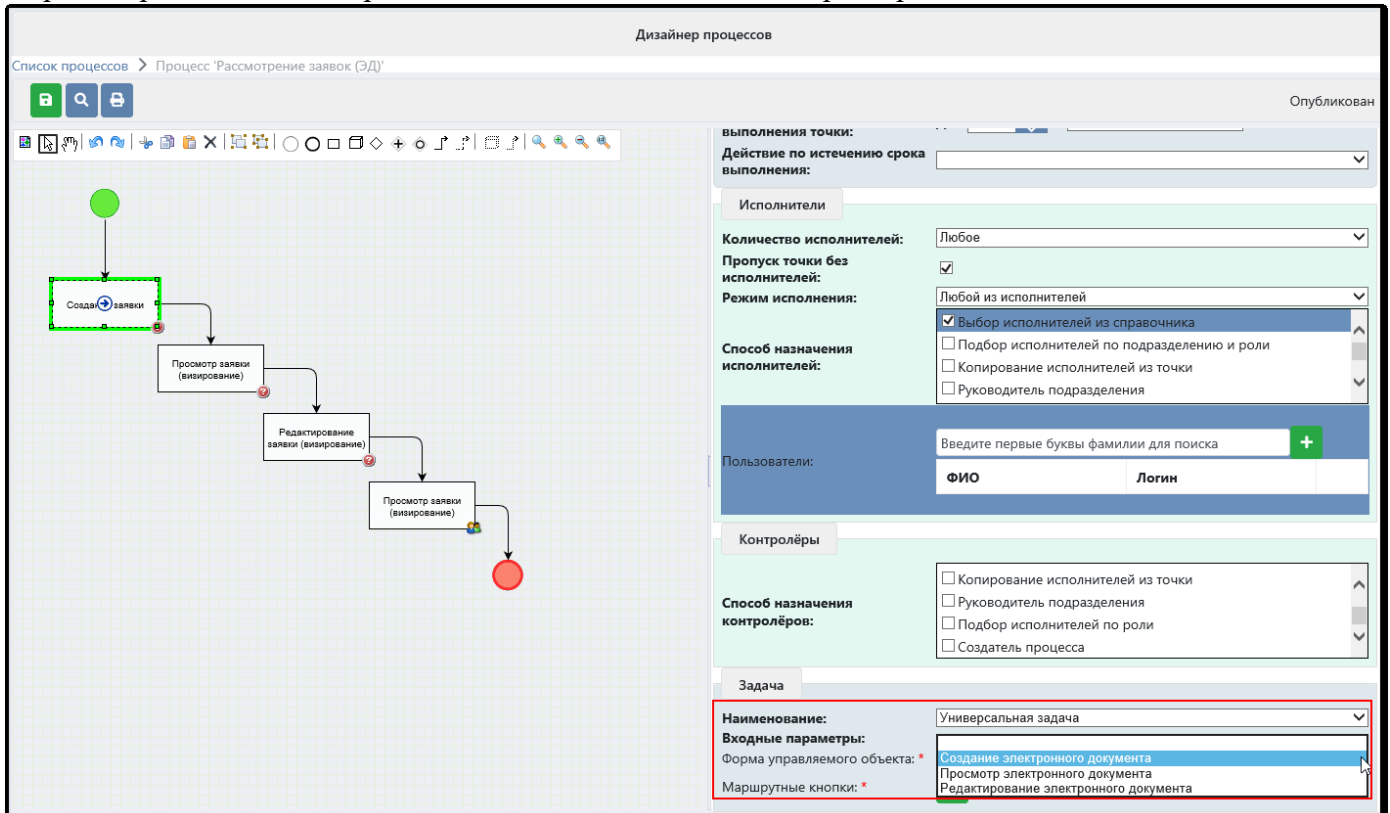


Рисунок 5.9-2. Выбор формы для универсальной задачи для ЭД

Все доступные формы заранее регистрируются в файле настроек *application.properties* в следующем формате:

```
bpm.flowcore.procedure.GenericTask.form.$edocument$=перечисление идентификаторов форм через запятую
```

```
bpm.flowcore.procedure.GenericTask.form.id_формы.$edocument$.url=url к форме в формате xhtml
```

```
bpm.flowcore.procedure.GenericTask.form.id_формы.$edocument$.name=наименование формы
```

и так для каждой формы, которая перечислена в первом параметре.

Например:

```
bpm.flowcore.procedure.GenericTask.form.$edocument$=form1,form2,form3
```

```
bpm.flowcore.procedure.GenericTask.form.form1.$edocument$.url=/studio-object-form-generator/ObjectEdit.xhtml?formId=form1
```

```
bpm.flowcore.procedure.GenericTask.form.form1.$edocument$.name=Создание электронного документа
```

```
bpm.flowcore.procedure.GenericTask.form.form2.$edocument$.url=/studio-object-form-generator/ObjectEdit.xhtml?formId=form2
```

```
bpm.flowcore.procedure.GenericTask.form.form2.$edocument$.name=Просмотр электронного документа
```

```
bpm.flowcore.procedure.GenericTask.form.form3.$edocument$.url=/studio-object-form-generator/ObjectEdit.xhtml?formId=form3
```

Если процесс описывает сценарий для обработки ЭД, на форме настройки кнопок есть возможность указать, должна ли в задаче осуществляться подписание документа. На данный момент ISIDA BPM поддерживает подписание простой электронной подписью (виза).

Рисунок 5.9-3. Настройка маршрутной кнопки для визирования документа

Любая кнопка (особенно та, которая двигает документ вперед) может стать кнопкой двойного назначения: она не только фиксирует решение, принятое пользователем (например, согласие с документом, передача управления следующему человеку по маршруту), но и вырабатывает подпись в электронном документе, чтобы зафиксировать это согласие юридически значимым способом. Для документальных процессов также важно уточнить статус подписи (завизировано / согласовано / подписано / утверждено).

Рисунок 5.9-4. Настройка статуса подписи при выполнении точки

Статусы подписей можно увидеть на закладке «Подписи» при просмотре электронного документа в поступлениях или мониторинге.

Монитор процессов

Статистика > Список процессов > Описание процесса

Электронный документ    Информация о процессе

Номер: 10    Дата \*: 05.04.2022

Краткое описание требований \*: Тестовая заявка на доработку СЭД    Тип заявки \*: Создание

Приоритет \*: Низкий    Статус: Подготовка предложений

Информационная система \*: Банк данных входной отчетности    Подразделение-заказчик: Главное управление персонала

Ответственный работник: Круглова Вера Михайловна

Файлы: **Подписи**

№	ФИО	Дата и время подписания	Должность	Статус	Результат проверки подписи
1	Круглова Вера Михайловна	05.04.2022 11:16	специалист	Завизировано	Подпись действительна
2	Круглова Вера Михайловна	05.04.2022 11:17	специалист	Согласовано	Подпись действительна
3	Урбанович Евгений Павлович	05.04.2022 11:23	Начальник отдела	Утверждено	Подпись действительна

Рисунок 5.9-5. Закладка «Подписи» электронного документа

В результате прохождения маршрута документ перемещается в оперативный архив, что обеспечивается заполнением полей объекта *depCode* (код архива). Система оперативного архива допускает функции поиска и просмотра ЭД, а в будущем и выгрузки в архив организации.

Дизайнер процессов

Список процессов > Процесс 'Рассмотрение заявок (ЭД)'

Опубликован

Маршрутные кнопки для точки 'Создание заявки'

	Название	Движение по маршруту	Выходные параметры	Поля объекта	Переменные процесса
<input type="radio"/>	Завизировать	Вперёд		depCode=5162	
<input type="radio"/>	Удалить	Назад		lid = 'recycle'	

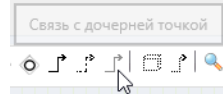
Сохранить    Отменить


Рисунок 5.9-6. Настройка кодов архивов для ЭД

Следует отметить, что в будущем в ISIDA BPM планируется поддержать более сложную комбинацию описанных функций: по маршруту будет двигаться структурированный объект, а на разных этапах на его основе будут порождаться различные электронные документы. Объект в этом случае будет использоваться для контроля всего процесса и объединения общих сведений, в некотором смысле выполнять функции пакетирования данных, а электронные документы будут фиксировать принятые решения юридически значимым образом, а также собирать материал для передачи в архив.

### 5.10. Дочерние точки

Этап бизнес-процесса может быть настроен таким образом, что исполнитель точки помимо обработки своей задачи вынужден добавить сотрудников для получения их мнения. Такие сотрудники («дочерние точки») никак не влияют на движение процесса, но оставляют свой комментарий, по которому исполнитель основной точки («родительской») может принять решение о судьбе объекта.



Для добавления дочерних точек в маршрут служит специальная связь - . Дочерняя точка оформляется в процессе как отдельный блок-задача и связывается с родительской упомянутой выше связью (рис.5.10-1).

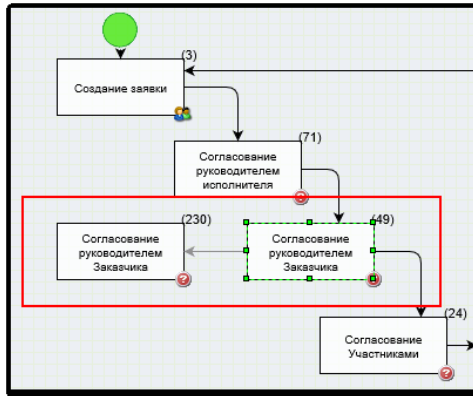


Рисунок 5.10-1. Добавление дочерних точек в дизайнера процессов

Традиционно считается, что **дочерние** исполнители начинают работу над поступившим объектом **одновременно**, поэтому в этой точке нужно указать в качестве режима исполнения – «Все исполнители (параллельно)». В **родительской** точке нужно добавить кнопку, по которой будет происходить добавление дочерних точек, и заполнить для этой кнопки раздел «Дочерние точки» (рис.5.10-2). В кнопке следует указать «Без движения по маршруту».

**Настройка кнопки**

Название:\*

Цвет:

Движение по маршруту:

Интерфейс    Управление данными    Управление другими точками маршрута    **Дочерние точки**

Описание:

Добавить точку	Наименование точки	Назначить исполнителей	Установить срок	Сопроводительный комментарий		
				Добавить	Обязательно	Заголовок
<input checked="" type="checkbox"/>	Согласование руководителем Главного управления Заказчика (232)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>

Рисунок 5.10-2. Настройка связи с дочерними точками

При исполнении задачи, в которой настроена связь с дочерними точками, пользователю будет предложен следующий диалог:

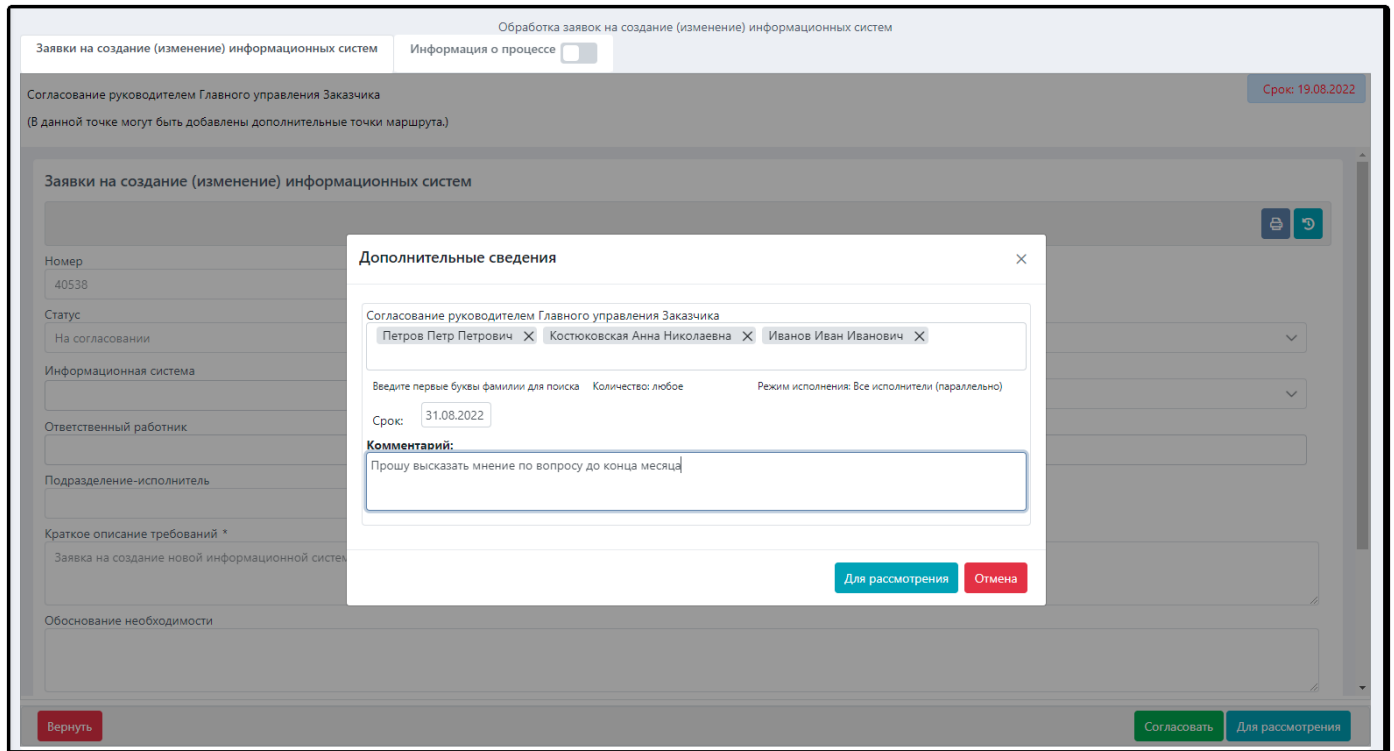


Рисунок 5.10-3. Добавление дочерних точек при выполнении задачи

Сотрудники, которые были добавлены в «родительской» точке отражены в табличном представлении процесса.

The screenshot shows the 'Информация о процессе' (Process information) view. A table displays the task list. The task 'Согласование руководителем Главного управления Заказчика' (Approval by the head of the Main Directorate of the Client) is expanded, showing three child nodes. The first child node is highlighted with a red border, and a tooltip is visible over it.

Задача	Исполнитель	Должность	Статус выполнения	Дата начала	Дата выполнения	Срок	Сопроводительный комментарий	Комментарий	Журнал
Создание заявки	Рябина Светлана Сергеевна		Выполнена	18.08.2022 17:50	18.08.2022 17:52				
Согласование руководителем Главного управления Заказчика	Рябина Светлана Сергеевна		Ожидание	18.08.2022 17:52		19.08.2022			
Рябина Светлана Сергеевна									
Согласование руководителем Главного управления Заказчика	Костюковская Анна Николаевна	Старший системный аналитик	Ожидание	18.08.2022 17:55		31.08.2022			
	Петров Петр Петрович	Секретарь	Ожидание	18.08.2022 17:55		31.08.2022			
	Иванов Иван Иванович	Заместитель начальника Главного управления	Ожидание	18.08.2022 17:55		31.08.2022			
Согласование в заинтересованных подразделениях	Ленточкина Ирина Иосифовна	Начальник отдела							

Рисунок 5.10-4. Отображение дочерних точек в таблице «Информация о процессе»

В графовом представлении блок с дочерними точками не отображается с целью упростить восприятие процесса простым пользователем. При нажатии на «родительскую» точку информация о ее «дочерних» отображается в правой части сеанса.

Поиск задач > Просмотр задачи

Обработка заявок на создание (изменение) информационных систем

Заявки на создание (изменение) информационных систем | Информация о процессе

Выполненные точки | Пропущенные точки | Открытые точки | Приостановленные точки | Просроченные точки | Возвращенные точки | Активный

Идентификатор: 49  
 Наименование: Согласование руководителем Главного управления Заказчика  
 Описание: В данной точке могут быть добавлены дополнительные точки маршрута.  
 Максимальный срок выполнения: Дни: 1 | Рабочие дни: 19.08.2022  
 Действие по истечению срока выполнения:  
 Дата и время открытия точки: 18.08.2022 17:52

Исполнители  
 Количество исполнителей: Любое  
 Пропуск точки без исполнителей: Да  
 Режим исполнения: Все исполнители (последовательно)  
 Руководитель подразделения  
 Копирование исполнителей из точки  
 Подбор исполнителей по подразделению и роли  
 Выбор исполнителей из справочника

Способ назначения исполнителей:  
 Пользователи:

ФИО	Логин
Рябина Светлана Сергеевна	isida

Дочерние точки

Автор	Исполнители	Статус выполнения
Рябина Светлана Сергеевна	Костюковская Анна Николаевна Петров Петр Петрович Иванов Иван Иванович	Открыта

Рисунок 5.10-5. Отображение дочерних точек в графе процесса

### 5.11. Выброс прикладных событий

Под выбросом события понимается создание экземпляра события каким-либо программным компонентом и передача события специальному слушателю, который будет управлять процессом его обработки. В роли программного компонента, выбрасывающего событие, в ISIDA BPM может выступать маршрутная кнопка или автоматическая задача.

#### Настройка выброса события для маршрутной кнопки

В любом этапе маршрута на любой кнопке при необходимости может быть настроен выброс *прикладного* события (рис.5.11-1,2).

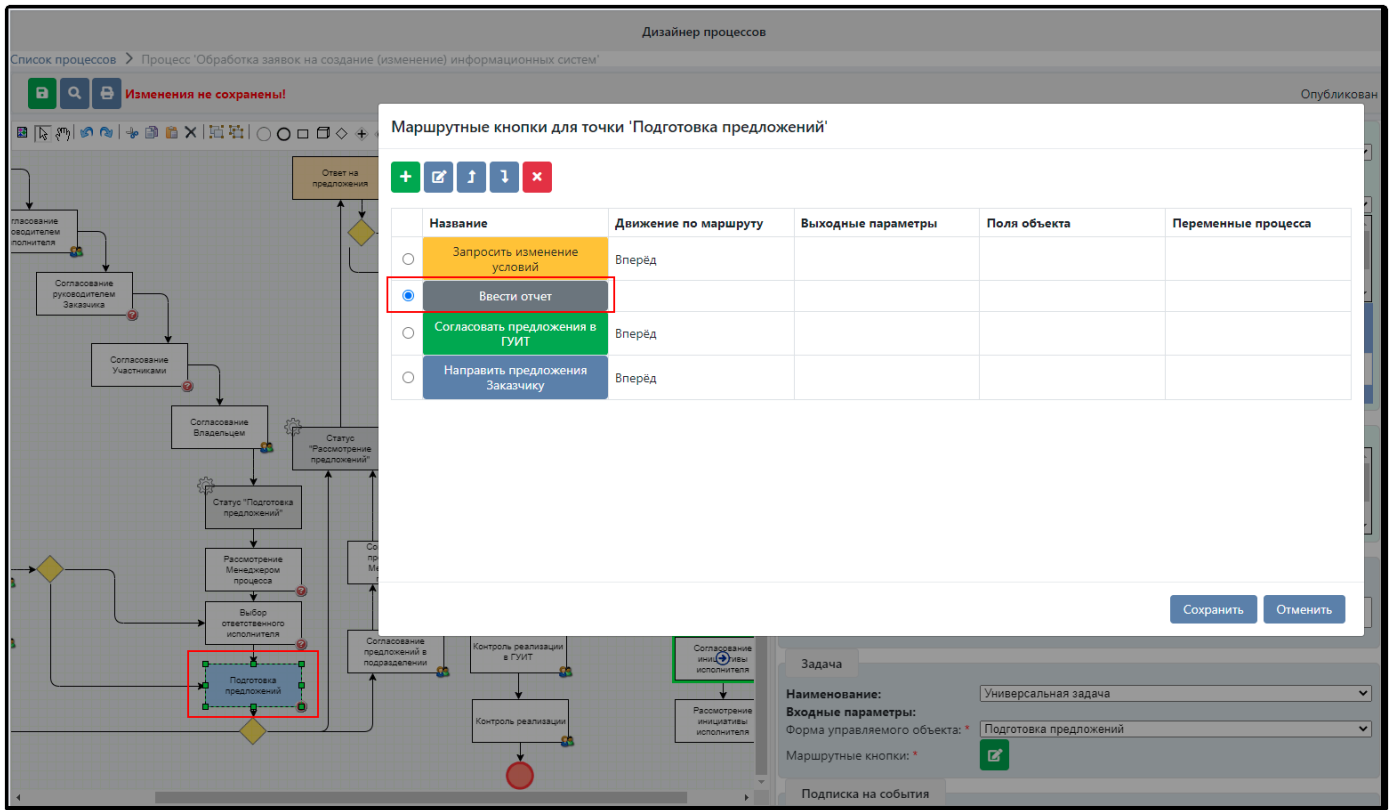


Рисунок 5.11-1. Настройка кнопки «Вести отчет»

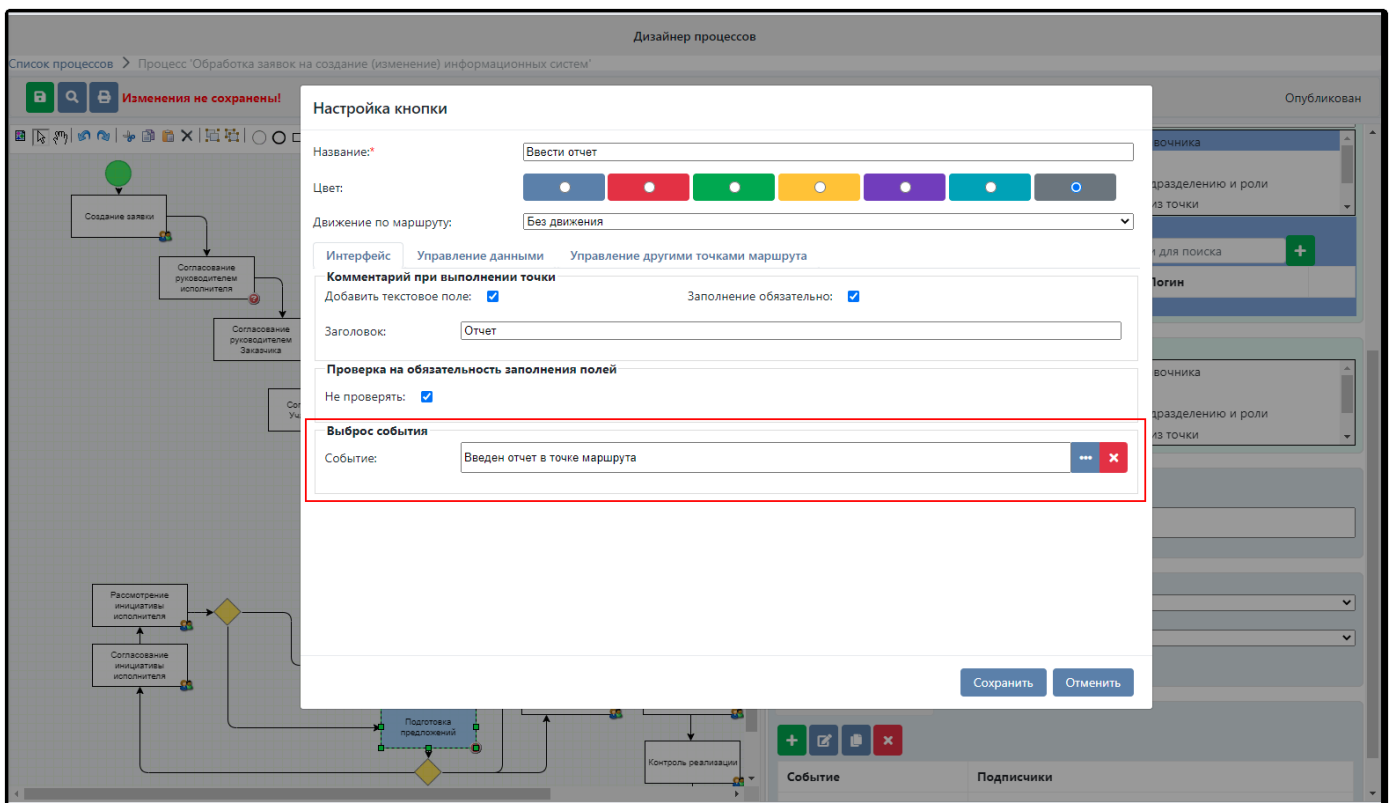


Рисунок 5.11-2. Настройка выброса события на кнопке

В качестве событий, которые может выбросить кнопка, предлагаются только **прикладные** события, т.е. те, которые создаются не системой, а администратором (рис.5.11-3).

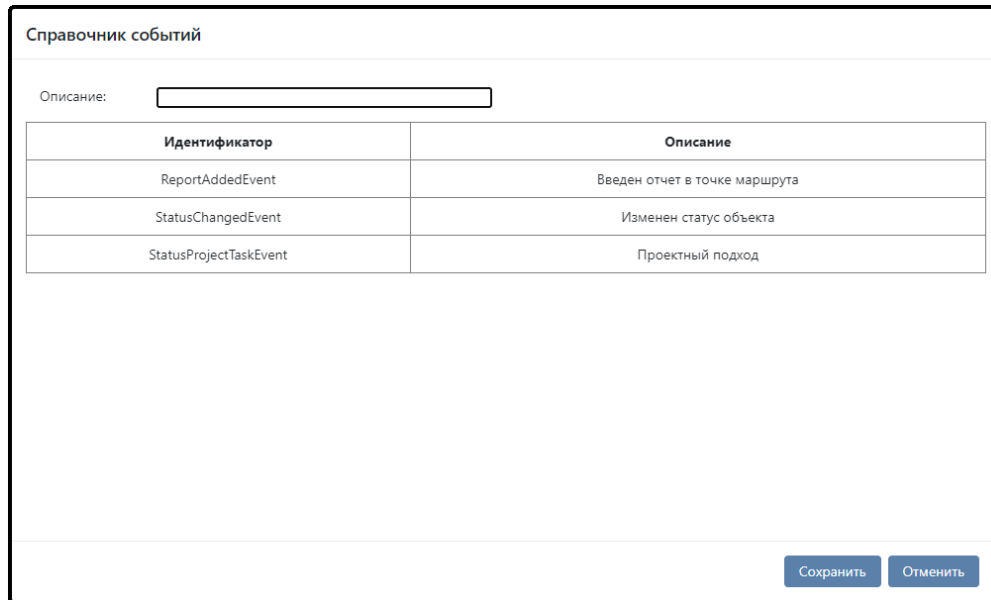


Рисунок 5.11-3. Выбор события из перечня прикладных

### Настройка выброса события для автоматической задачи

На автоматических задачах «Изменение полей объекта» также может быть настроен выброс прикладного события (рис.5.11-4).

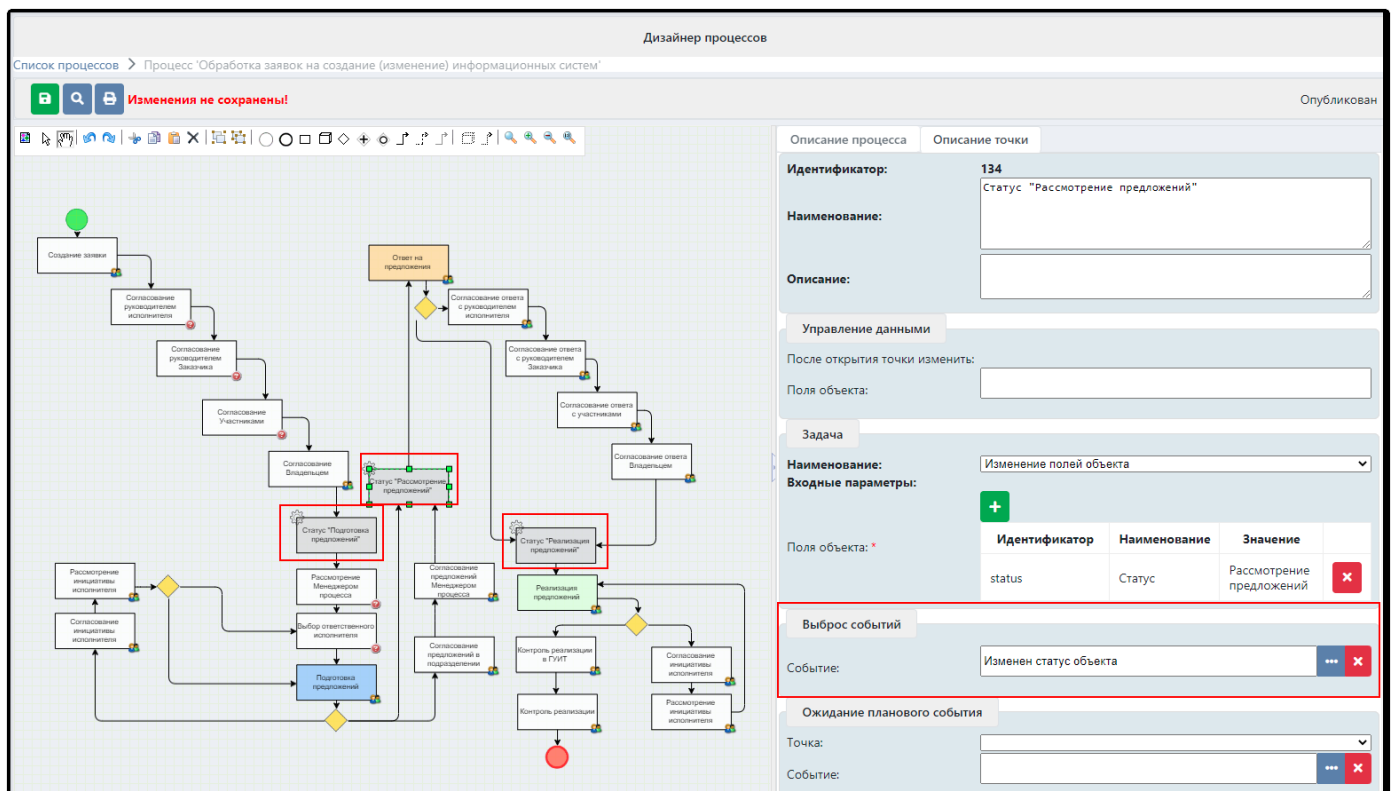


Рисунок 5.11-4. Настройка выброса события «Изменен статус объекта»

В данном примере, после успешного прохождения этой точки маршрута, во-первых, будет изменен статус объекта, а во-вторых, будет выброшено событие об изменении статуса объекта.



## 5.12. Плановые события

### Настройка ожидания планового события

На точке маршрута можно настроить не только выброс прикладного события, но и запланировать наступление события в будущем. Например, в некотором процессе в точке «Статус Подготовка предложений» начинается отсчет 20 рабочих дней, которые предусмотрены бизнес-процессом на подготовку предложений. Плановый срок в 20 рабочих дней – это не срок на конкретной точке, а срок, который отводится на обработку целого этапа маршрута (на рис.5.12-1 он отмечен синей рамкой).

Событие	Подписчики
Приближается срок планового события	Исполнители выбранных точек: Подготовка предложений (8)
Наступил срок планового события	Исполнители выбранных точек: Создание заявки (3); Подготовка предложений (8)
Превышен срок планового события	Исполнители выбранных точек: Создание заявки (3); Подготовка предложений (8); Выбор ответственного исполнителя (55); Согласование руководителем исполнителя (71)

Рисунок 5.12-1. Настройка ожидания наступления планового события

Для того, чтобы «запланировать» наступление события на точке, нужно сначала выбрать точку, событие которой ожидается. Для выбора точки система предлагает выпадающий список (рис.5.12-2).

Рисунок 5.12-2. Выбор точки, событие которой ожидается

Затем необходимо выбрать событие из предложенного перечня (система предлагает только те **прикладные** события, которые выбрасываются именно выбранной точкой).

Идентификатор	Описание
StatusChangedEvent	Изменен статус объекта

Рисунок 5.12-3. Выбор ожидаемого события

Затем следует установить срок, тип дней и внести описание планового события (текст).

Рисунок 5.12-4. Установка контрольного срока и описания события

Описание планового события и целевой срок (планируемая дата) могут использоваться в уведомлениях о приближении, наступлении и истечении планового срока. Для это применяются следующие шаблоны:

- plan.event.description* – описание планового события;
- plan.event.targetTime* – плановый срок события.

Подробно о подписке на события и создании макетов уведомлений рассказано [в разделе 8.4.](#)

### **Начало отсчета времени ожидания планового события**

Как только точка, на которой установлено ожидание планового события пройдена, начинается отсчет рабочих или календарных дней. Плановые события можно увидеть на соответствующей закладке в сеансе просмотра (мониторинга) процесса:

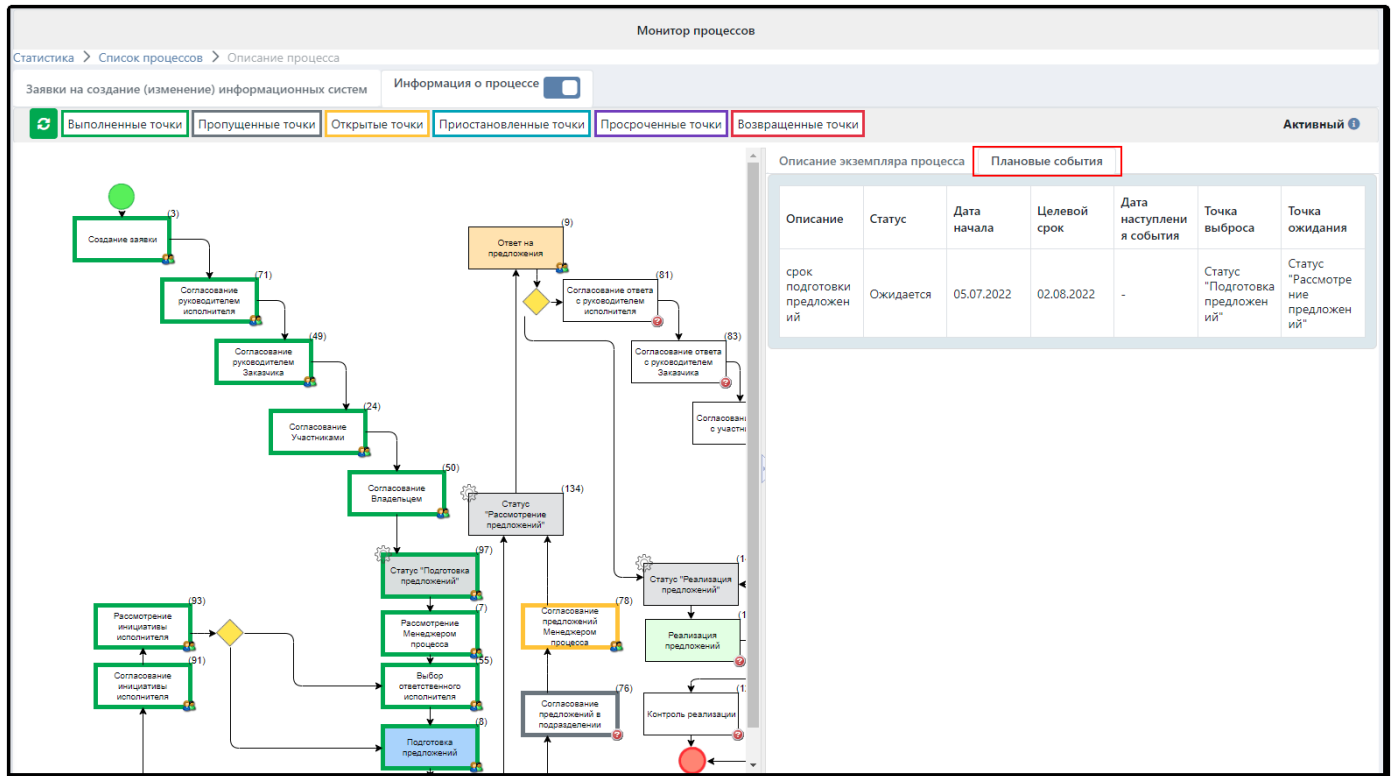


Рисунок 5.12-5. Просмотр плановых событий

Как только плановое событие наступит, то есть в точке «Статус «Рассмотрение предложений» будет выброшено событие «Изменен статус объекта», специальным регламентом «Контроль плановых событий» в таблице «Плановые события» будет заполнена графа «Дата наступления события». Пока событие не наступило, регламенты контроля сроков наступления плановых событий отслеживают дату его приближения. По мере приближения к запланированному целевому сроку, регламенты контроля фиксируют события:

- «Приближается срок планового события» (работает регламент «Событие о приближении срока планового события»)
- «Наступил срок планового события» (регламент «Событие о наступлении срока планового события»);
- «Превышен срок планового события» (регламент «Событие о превышении срока планового события»).

Как только плановое событие наступило, целевой срок регламентом больше не контролируется. Более подробно о регламентах системы описано [в разделе 8.5](#).

### Перенос плановых событий

В системе предусмотрен функционал переноса плановых событий. Для этого в маршрутной задаче должна быть предусмотрена кнопка со специальной настройкой - свойством «Изменение срока планового события» (рис.5.12-6).

Рисунок 5.12-6. Настройка свойства для переноса планового события

В этом случае при исполнении задачи пользователю будет предложен диалог, в котором он увидит плановый срок и поле для его изменения. Также можно указать сопроводительный комментарий (причину переноса срока).

Рисунок 5.12-7. Перенос планового события

Новый срок, автор переноса срока и его комментарий будут зафиксированы в журнале операций на закладке «Плановые события».

Поиск задач > Просмотр задачи

Обработка заявок на создание (изменение) информационных систем

Заявки на создание (изменение) информационных систем | Информация о процессе

Выполненные точки | Пропущенные точки | Открытые точки | Приостановленные точки | Просроченные точки | Возвращенные точки | Активный

Описание экземпляра процесса | Плановые события

Описание	Статус	Дата начала	Целевой срок	Дата наступления события	Точка выброса	Точка ожидания	Журнал
Срок подготовк и предложений	Ожидается	16.08.2022	27.09.2022	-	Статус "Подготовка предложения"	Статус "Рассмотрение предложения"	

Журнал операций

Дата операции	Событие	Работник	Содержание	Предыдущий срок	Новый срок
17.08.2022	Изменение срока планового события	Рябина Светлана Сергеевна	Срок продлен на неделю	13.09.2022	20.09.2022
18.08.2022	Изменение срока планового события	Рябина Светлана Сергеевна	Срок перенесен на неделю	20.09.2022	27.09.2022

Рисунок 5.12-8. Журнал операций для просмотра действий с плановыми событиями

В момент переноса срока планового события фиксируется соответствующее событие о переносе. Позже система по общим правилам рассылает уведомления всем сотрудникам, подписанным на него.

Изменён "Срок подготовки предложений"  
**Новый плановый срок: 27.09.2022**

**Плановый срок изменил:** Рябина Светлана Сергеевна  
**Комментарий:** Срок перенесен на неделю

**Реквизиты:**  
**Номер:** 40515  
**Дата:** 16.08.2022  
**Содержание:**  
**Тип заявки:** Создание  
**Приоритет:** Обычный  
**Статус:** Подготовка предложений  
**Подразделение:** Главное управление операций на финансовых рынках  
**Информационная система:** -

Рисунок 5.12-9. Уведомление об изменении срока планового события

## 6. Диспетчер процессов

**Диспетчер процессов** – это корневой компонент, «движок», в котором «живут» выполняющиеся экземпляры процессов. Диспетчер запускает в точках процесса требуемые задачи и осуществляет переходы между точками процесса в соответствии с установленной логикой.

### 6.1. Основные термины и определения

Для того, чтобы описать правила движения объекта по процессу, реализуемые в ISIDA BPM, требуется дать несколько дополнительных определений.

**Начальная точка** – точка, запускающая процесс обработки объекта. При открытии такой точки процесс считается запущенным (**активным**).

**Задача (процедура, точка)** – исполняемый сценарий, подразумевающий ручное или автоматическое выполнение. В задаче может быть назначен срок исполнения, один или несколько исполнителей и контролёров. Пользовательские задачи ожидают решения пользователя, поэтому могут долгое время находиться в активном (открытом / запущенном) состоянии. Автоматические задачи выполняются немедленно, поэтому им активное состояние не характерно.

**Переход** – однонаправленная связь между двумя точками, этап, который начинается в момент закрытия точки-начала и заканчивается в момент открытия точки-конца. Переход от точки к точке осуществляется мгновенно.

**Условие перехода** – логическое выражение, которое оценивается системой для определения последовательности выполнения процедур в рамках процесса. Переход может быть безусловным, в этом случае по закрытии точки-начала процесс автоматически переходит к следующей точке. На переходе может быть установлено условие, тогда он будет осуществлен только при его выполнении.

**Конечная точка** – точка, завершающая процесс обработки объекта. Процесс в системе ISIDA BPM может иметь только одну начальную точку и несколько конечных. При первом достижении одной из конечных точек процесс считается пройденным (**завершённым**).

**Приостановленный процесс** – процесс, в котором произошла ошибка в продвижении от точки к точке: точка не была корректно завершена, не нашлось переходов в следующую точку, или подходящих переходов нашлось несколько.

## 6.2. Запуск экземпляра процесса

Для каждого управляемого объекта может быть предусмотрено несколько процессов, поэтому первоначально Диспетчер выполняет подбор процесса, соответствующий конкретному экземпляру объекта. Основным критерием отбора является тип объекта (для электронных документов – тип документа), затем проверяются дополнительные критерии, указанные в описании процесса, если дополнительных критериев нет, то достаточным считается соответствие типа. Объект будет направлен по первому подходящему процессу.

При запуске экземпляра процесса система осуществляет поиск начальной точки, с которой начинается движение, следующая за ней точка открывается автоматически, начинается процедура ее исполнения: статус точки устанавливается в значение «Открыта и выполняется» (*open*), фиксируется дата и время запуска, если точка является автоматической, выполняется ее процедура. Статус экземпляра процесса также устанавливается в значение «Активный» (*open*), дата и время запуска сохраняются в соответствующих переменных экземпляра процесса.

## 6.3. Выполнение экземпляра процесса

Движение начинается из начальной точки, открывающейся при запуске экземпляра процесса. После выполнения процедуры (участником точки или системой автоматически), осуществляется закрытие точки и поиск выхода для дальнейшего движения. Под закрытием понимается изменение статуса точки на «Завершена» (*closed.completed*), фиксация даты и времени закрытия (обработки) точки.

После закрытия точки, осуществляется проверка, является ли данная точка конечной. Если да, то экземпляр процесса считается завершённым, выполняется его остановка. Если у точки нет исходящих переходов, движение по этой ветке прекращается. Если исходящие переходы существуют, выбираются переходы для дальнейшего движения.

- 1) Если у точки есть безусловные переходы, все они считаются подходящими.
- 2) Если у точки есть условные переходы, то подходящими считаются те из них, для которых в настоящий момент выполняются указанные условия.

Важная особенность: поиск подходящего перехода для обычной точки выполняется **до первого** совпадения, таким образом, безусловные переходы всегда имеют наивысший приоритет, затем проверяются условные переходы – как только найден переход с выполняющимся условием, движение выполняется по нему.

После выбора перехода для дальнейшего движения выполняется вход в следующую точку. Статус устанавливается в значение «Открыта и выполняется» (*open*), фиксируется дата и время запуска. Если открытая точка является автоматической, то запускается выполнение ее задачи. Если задача выполнена успешно, происходит дальнейшее движение по процессу (точка закрывается, осуществляется переход, открывается следующая по процессу и т.д.). Если автоматическая задача выполнена некорректно, меняется статус процесса - «Приостановленный».

#### 6.4. Остановка экземпляра процесса

Остановка экземпляра процесса выполняется при достижении первой конечной точки маршрута. При этом фиксируется дата и время остановки, а статус процесса меняется на «Завершен» (*close*). При этом не выполняется закрытие всех остальных точек. Они так и остаются в запущенном состоянии. Исключение таких экземпляров объектов из поступлений сотрудников обеспечивает сам алгоритм (он отбирает открытые точки в открытых экземплярах процессов).

Наличие открытых точек в остановленных экземплярах процессов позволяет осуществлять учет процессов, выполненных в определенном периоде, формировать для пользователя «картину дня».

#### 6.5. Исполнение точек

Ход процесса обеспечивается специальным комплексом системы «BPM-Ядро» (*bpm.flowcore*), который представлен двумя услугами «Взаимодействие с СУБД» (*bpm.flowcore.dbinteractor*) и «Диспетчер» (*bpm.flowcore.dispatcher*).

Алгоритм исполнения для **автоматических** точек следующий:

1. С заданной периодичностью диспетчер ищет точки в статусе «open» запущенных процессов. Порция (количество) точек для поиска задается параметром:  
*bpm.flowcore.dispatcher.SystemProcedureExecuterService.maxPoolSize*  
и по умолчанию равен 10 точек.  
Периодичность работы диспетчера задается параметром:  
*bpm.flowcore.dispatcher.SystemProcedureExecuterService.fixedRate*  
и по умолчанию равен 5 секунд.

Кроме этого у диспетчера есть возможность «засыпать», если нет активности по процессам;

2. Диспетчер исполняет точку, осуществляя переход до следующей точки.

Алгоритм исполнения для **пользовательских** точек следующий:

1. Диспетчер получает идентификатор точки от BPM в момент обработки этой точки исполнителем (пользователем в системе);
2. Диспетчер исполняет точку, осуществляя переход до следующей точки. Пользователь в сеансе ожидает результата перехода на следующей точке.

## 7. Монитор процессов

**Монитор процессов** позволяет администраторам получать актуальную комплексную информацию о ходе процессов, исполняющихся в ISIDA BPM. Данная услуга также позволяет при необходимости досрочно завершить процессы, возобновить приостановленный процесс, откатить процесс на ранее

заданную точку. В сеансе монитора процессов администраторам предоставлена возможность смены исполнителей и контролёров в точках процесса, остановка/запуск точки, изменение условий и изменение входных параметров процедур.

## 7.1. Поиск и просмотр процессов

Сеанс поиска процессов отображает количество активных, приостановленных и завершённых экземпляров процесса по каждому из объектов и по электронным документам.

№ п/п	Объект	Активные	Приостановленные	Завершённые
1	Электронный документ	3	1	16
2	Заявки на создание (изменение) информационных систем	26	4	6

Рисунок 7.1-1. Сеанс «Монитор процессов»

**Активный** – означает, что процесс был запущен и на данный момент не завершился (исполняется). Обработка объекта/документа продолжается.

**Приостановленный** – означает, что при выполнении какой-либо процедуры произошел сбой, и точка завершилась некорректно или при завершении одной из точек процесса не был найден подходящий переход в другую точку, в результате чего выполнение процесса приостановилось.

**Завершённый** – процесс обработки объекта подошел к своему логическому завершению, успешно достигнув конечной точки.

Сеанс для просмотра списка процессов разделен на две части (рис.7.1-2). В левой части расположены критерии для поиска, в правой – список найденных процессов.



ISIDA BPM 89 Иванов Иван Иванович  
Администратор

Монитор процессов

Статистика > Список процессов 'Заявки на создание (изменение) информационных систем'

Статистика

Уникальный номер объекта:

Статус экземпляра процесса:

Инициатор:

Дата запуска: с: 01.01.2022 по:

Поля объекта

Статус:

Тип заявки:

Приоритет:

Краткое описание требований:

Очистить Искать

Найдено: 21

№ п/п	Уникальный номер объекта ↑↓	Статус экземпляра процесса ↑↓	Наименование процесса ↑↓	Подпроцесс	Открытые точки экземпляра процесса	Дата запуска ↑↓	Дата завершения ↑↓	Инициатор ↑↓	
1	10311	Активный	Подпроцесс согласования в управлении	Да	2	19.01.2022 12:52		Иванов Иван Иванович	🔍
2	10311	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	1	19.01.2022 12:50		Иванов Иван Иванович	🔍
3	10310	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	1	19.01.2022 11:44		Иванов Иван Иванович	🔍
4	10309	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	1	17.01.2022 15:45		Иванов Иван Иванович	🔍
5	10307	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	0	14.01.2022 10:24		Иванов Иван Иванович	🔍
6	10300	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	1	11.01.2022 09:52		Иванов Иван Иванович	🔍
7	10299	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	1	06.01.2022 15:27		Иванов Иван Иванович	🔍
8	10298	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	1	06.01.2022 15:12		Иванов Иван Иванович	🔍
9	10297	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	1	06.01.2022 15:07		Иванов Иван Иванович	🔍
10	10296	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	0	06.01.2022 09:54		Иванов Иван Иванович	🔍
11	10295	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	1	05.01.2022 17:10		Иванов Иван Иванович	🔍
12	10294	Активный	Рассмотрение заявок (ЭТАЛОН)	Нет	1	04.01.2022 15:18		Иванов Иван Иванович	🔍

Рисунок 7.1-2. Список процессов в сеансе монитора

Критерии в разделе «Поля объекта» настраиваются администратором в файле настроек *application.properties* в виде:

*as.integration.bpm.searchFields.mun* объекта=идентификаторы полей объекта через запятую, например:

*as.integration.bpm.searchFields.itrequest=status,requestType,priority,contents*

Поля объекта

Статус:

Тип заявки:

Приоритет:

Краткое описание требований:

Очистить Искать

Рисунок 7.1-3. Поля объекта для поиска бизнес-процессов

Состав колонок в таблице с найденными процессами настраивается в файле *application.properties*:

# Колонки результата поиска в мониторинге по полям объекта *itrequest*

# Параметр (#sortable) - для установки сортировки колонки

```
# В датах сначала пишется форматер
as.integration.bpm.monitor.viewFields.itrequest=number,process$objectPid,process$state,process$name,pr
ocess$subprocess,process$numberThreads,process$startTime|dd.MM.yyyy
HH:mm#sortable,process$endTime|dd.MM.yyyy HH:mm#sortable,process$creatorId
```

Бизнес-процесс открывается в сеансе, схожем с сеансом дизайнера процессов, но с меньшим объемом функций (рис.7.1-4).

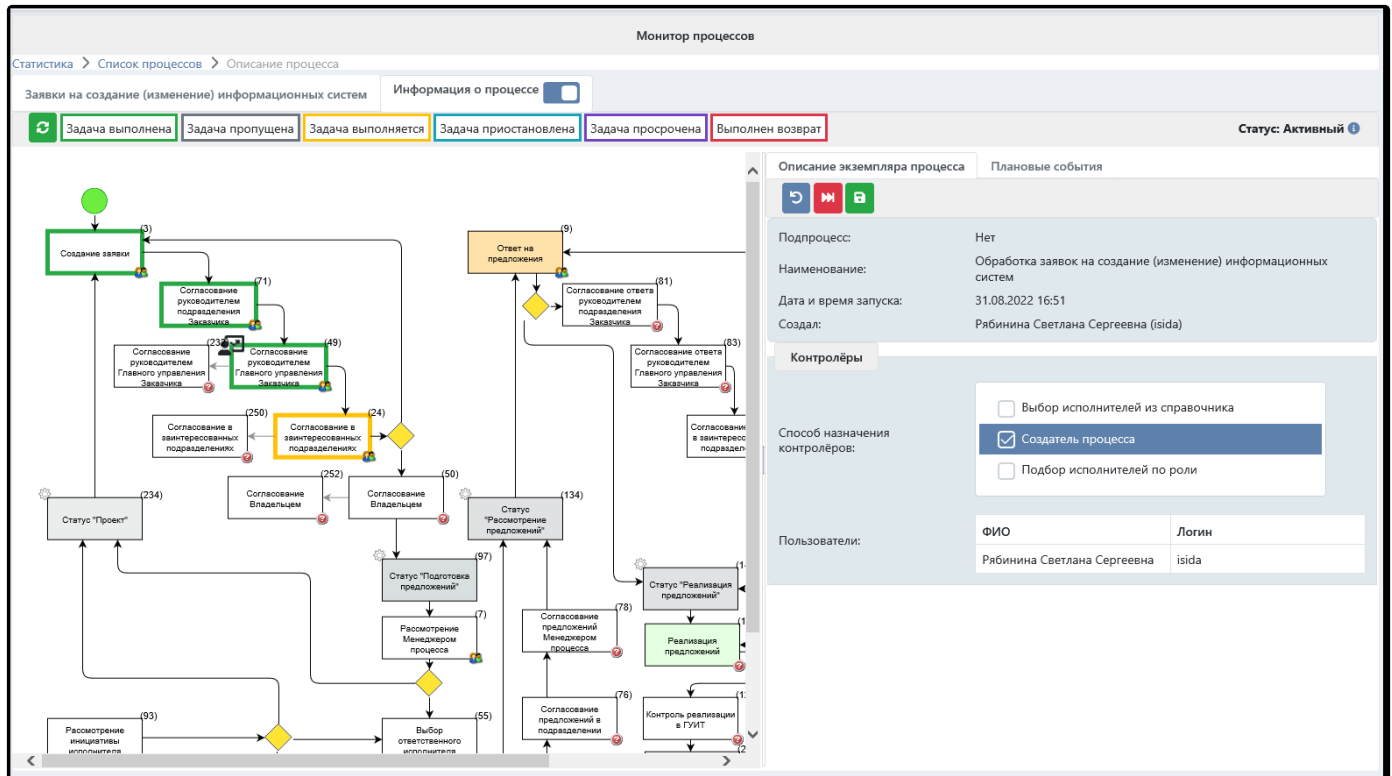




Рисунок 7.1-4. Сеанс просмотра бизнес-процесса

Основные рабочие области сеанса:

### Верхняя панель



Кнопка «Обновить изображение процесса»  позволяет вручную обновить данные экземпляра процесса. Кнопка «Сведения для разработчиков»  предназначена для сохранения файла JSON с данными о процессе. В правой части панели указан статус бизнес-процесса, например, «Активный». В цветных прямоугольниках расположены статусы точек в зависимости от цвета элементов в графе.

**Область для отображения графа**, который отражает текущее состояние экземпляра процесса в режиме реального времени (рис.7.1-5).

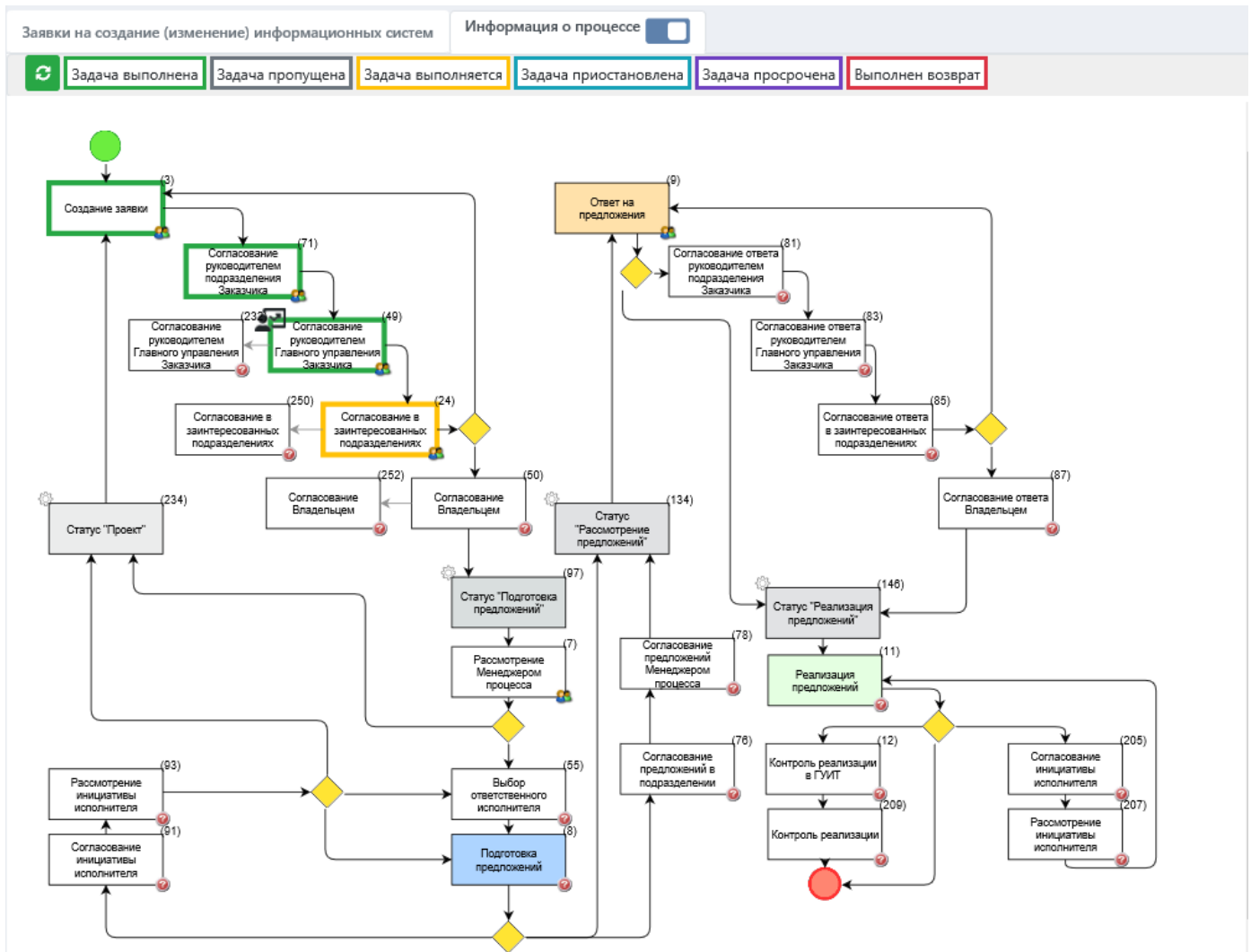


Рисунок 7.1-5. Граф для отображения экземпляра процесса

Для наглядности точки с разными статусами отмечены разными цветами. Выполненные задачи помещены в **зеленую** рамку, открытые и выполняющиеся в настоящий момент задачи – в **желтую** рамку, приостановленные задачи – в **синюю** рамку, просроченные задачи – в **фиолетовую**, пропущенные – в серую, возвращенные – в **красную**, следующие по хронологии задачи – в **черную**.

Информацию о процессе можно просмотреть как в виде графа, так и в виде таблицы, с помощью переключателя режима просмотра  **Граф/Таблица** (рис.7.1-6).

Задача	Исполнитель	Должность	Статус выполнения	Дата начала	Дата выполнения	Срок	Сопроводительный комментарий	Комментарий	Журнал
Создание заявки	Рябинина Светлана Сергеевна		Выполнена	31.08.2022 16:51	31.08.2022 16:55				
Согласование руководителем подразделения Заказчика	Петров Петр Петрович	Секретарь	Выполнена	31.08.2022 16:55	31.08.2022 17:00				
Согласование руководителем подразделения Заказчика	Ленточкина Ирина Иосифовна	Начальник отдела	Выполнена	31.08.2022 17:00	31.08.2022 17:00				
Согласование руководителем Главного управления Заказчика	Петров Петр Петрович	Секретарь	Выполнена	31.08.2022 17:00	31.08.2022 17:04	01.09.2022		Согласовано	
Согласование руководителем Главного управления Заказчика	Крупова Вера Михайловна		Ожидание	31.08.2022 17:03					
Согласование руководителем Главного управления Заказчика	Степанов Виктор Дмитриевич		Ожидание	31.08.2022 17:03					
Согласование в заинтересованных подразделениях	Ленточкина Ирина Иосифовна	Начальник отдела	Выполнена	31.08.2022 17:04	31.08.2022 17:05				
Согласование в заинтересованных подразделениях	Петров Петр Петрович	Секретарь	Ожидание	31.08.2022 17:04					
Создание заявки	Рябинина Светлана Сергеевна								
Согласование Владелец									

Рисунок 7.1-6. Таблица для отображения информации о процессе

В таблице этапы маршрута также отмечены иконками:

- успешно пройденные задачи;
- задачи, в которых был произведен возврат;
- приостановленные администратором или системой задачи;
- открытые, но не исполненные задачи, ожидающие обработки;
- задачи, ожидающие обработки (еще не открытые).
- задачи, в которых были добавлены дополнительные участники согласования.

Кнопка «Печать» формирует файл PDF с таблицей «Информация о процессе». На выбор предлагается печать отчета в обычной и расширенной форме.

**Панель описания реквизитов** (расположена в правой части формы), разбитая на две вкладки: «Описание экземпляра процесса» и «Описание точки» (рис. 7.1-7).

Описание экземпляра процесса | Описание точки

Подпроцесс: Нет

Наименование: Рассмотрение заявок (ЭТАЛОН)

Дата и время запуска: 30.11.2021 14:30

Создал: Иванов Иван Иванович (test1)

**Контролёры**

Заполнение

По ФИО из справочника пользователей

По функциональной группе

По создателю процесса

Пользователи:

ФИО

Логин

Рисунок 7.1-7. Вкладка «Описание экземпляра процесса»

Панель кнопок служит для управления процессом:



Кнопка «Возврат на начальную точку» маршрута позволяет остановить выполнение процесса на текущей точке, добавить комментарий при возврате, снова открыть первую точку этого же маршрута и заново запустить процесс (рис.7.1-8 и 7.1-9). При этом выбрасывается событие «Возврат в точке маршрута».

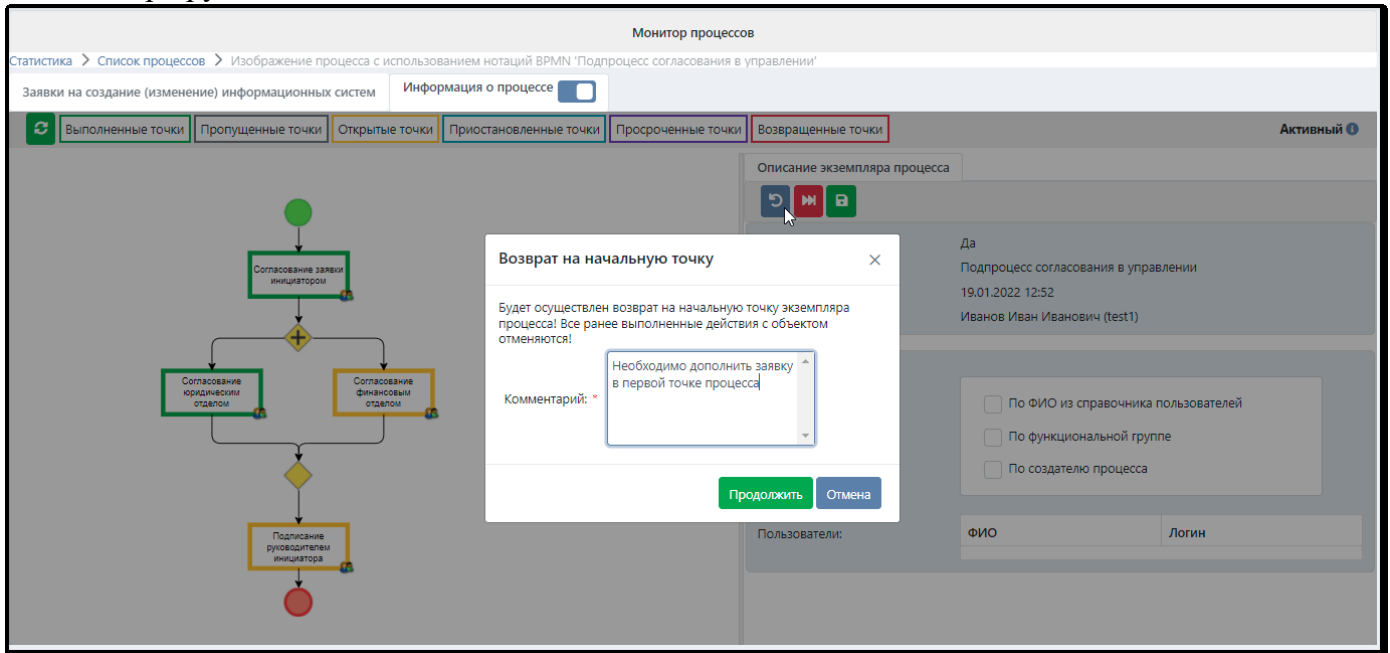


Рисунок 7.1-8. Форма для ввода комментария при возврате

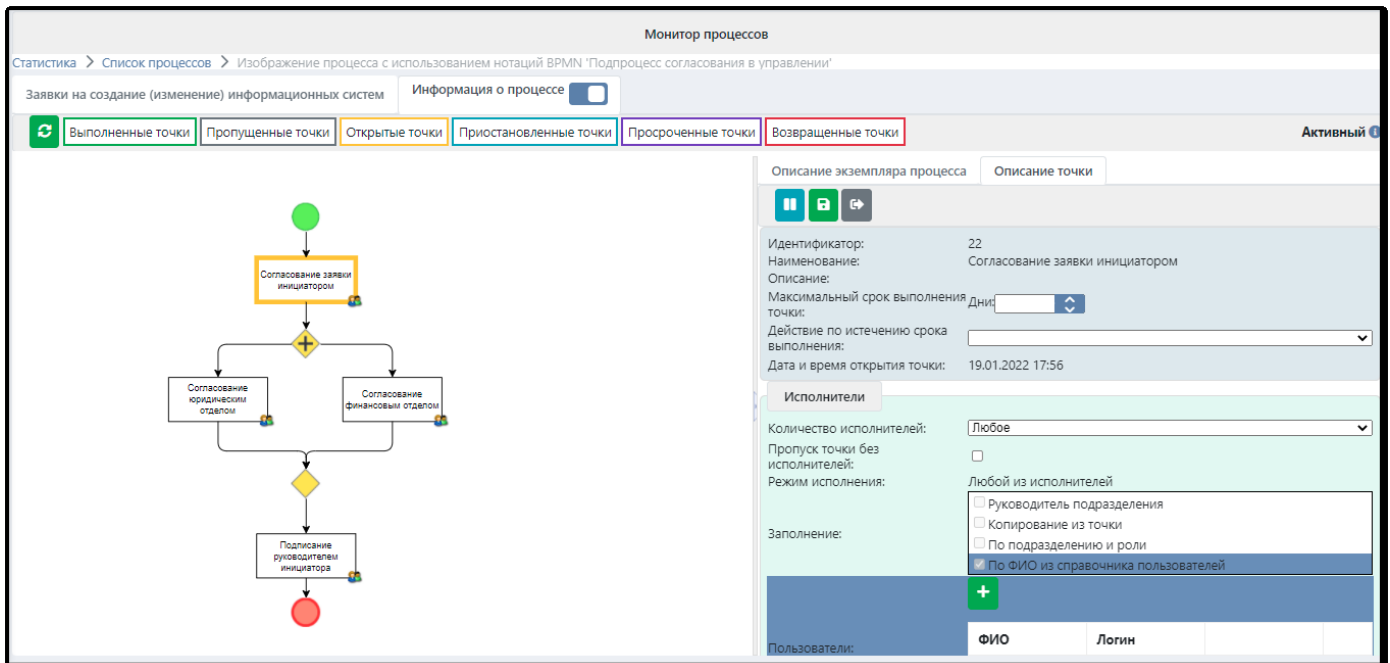



Рисунок 7.1-9. Процесс с открытой первой точкой



Кнопка «Завершить экземпляр процесса» позволяет досрочно завершить процесс (рис.7.1-10), после чего статус экземпляра процесса изменится с «Активный» на «Завершенный». Комментарий, введенный при завершении, будет отображен в свойствах процесса. Затем при необходимости процесс можно удалить по кнопке  (рис.7.1-10). При завершении процесса объект, находящийся на этом маршруте, автоматически уходит из поступлений исполнителей.

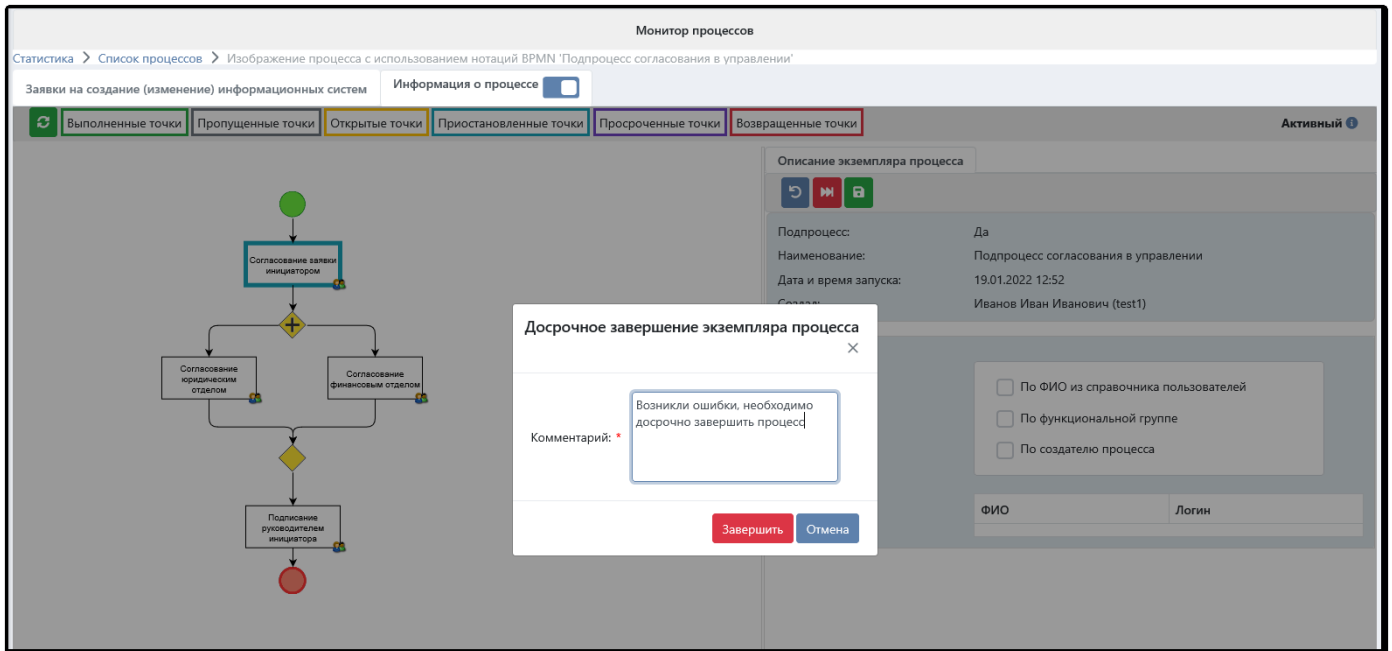


Рисунок 7.1-10. Досрочное завершение экземпляра процесса

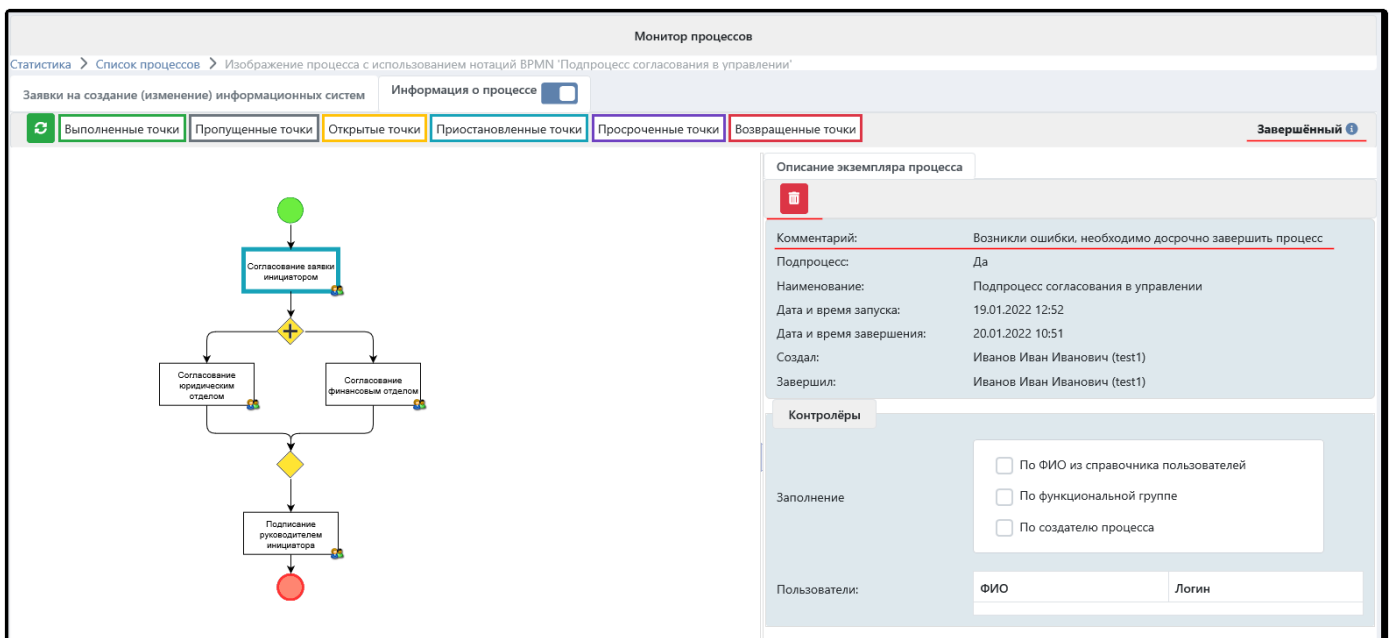




Рисунок 7.1-11. Просмотр завершённого процесса

 **Кнопка «Запустить экземпляр процесса»** позволяет запустить приостановленный процесс, возобновить его исполнение. Статус экземпляра процесса изменится с «Приостановленный» на «Активный».

 **Кнопка «Сохранить экземпляр процесса»** позволяет сохранить изменения, внесенные администратором в конкретный исполняющийся экземпляр процесса. Например, администратор может изменить состав контролёров процесса, и сохранить эти изменения. Они будут применены к исполняющемуся процессу.

## 7.2. Управление ходом исполнения процесса

Рассмотрим закладку «Описание точки» и возможности, доступные на ней администратору. Они отличаются в зависимости от статуса точки.

### *Точка завершена успешно*

В случае, если текущее состояние точки «Выполнена», то на вкладке «Описание точки» отображаются ее свойства в режиме просмотра (без возможности внесения изменений).

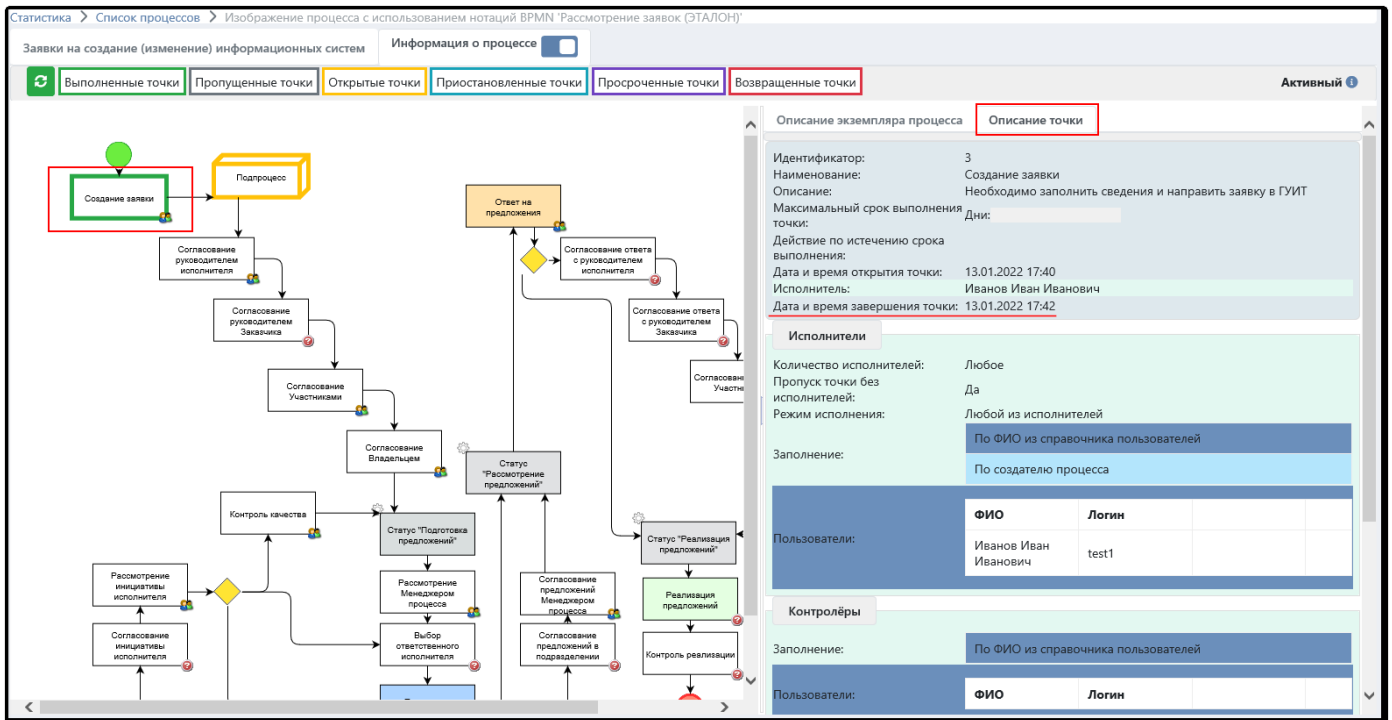


Рисунок 7.2-1. Вкладка «Описание точки» для выполненного этапа маршрута

### Точка открыта (исполнение начато)

В случае, если текущее состояние точки «Открыта» (ожидание обработки) или «Создана, не выполняется/не стартовала» (рис. 7.2-2) администратор может не только просмотреть свойства этой точки, но и изменить участников, условия перехода, состав или наименования маршрутных кнопок.

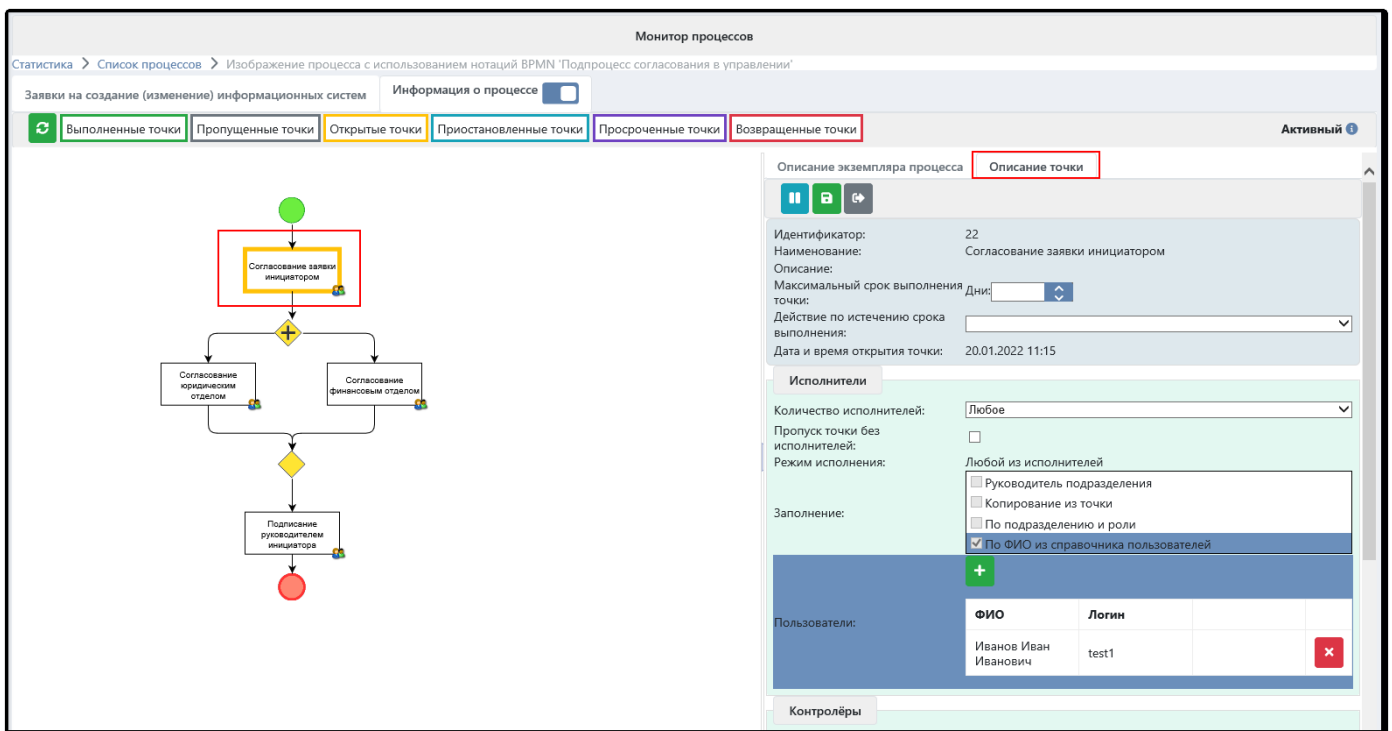




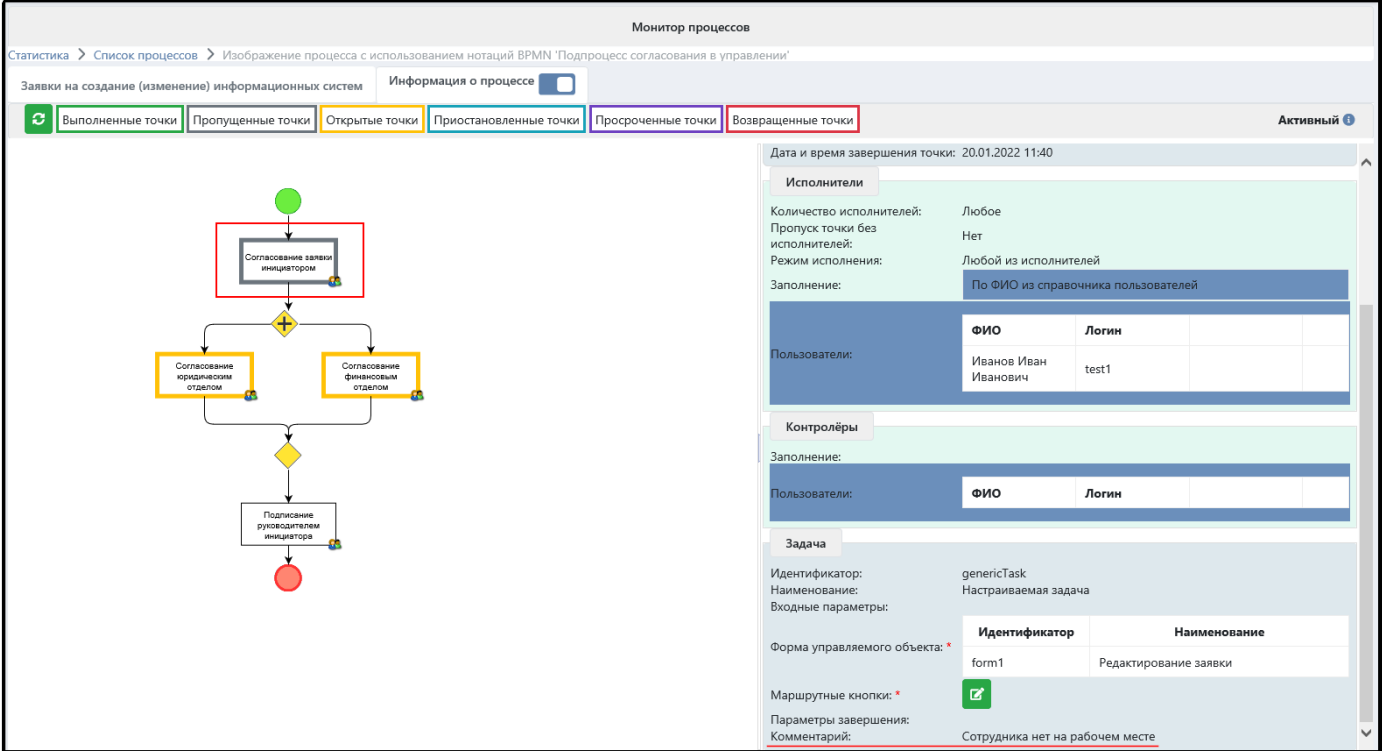


Рисунок 7.2-2. Вкладка «Описание точки» для открытого этапа маршрута

Доступные для управления кнопки:

-  - остановить точку; статус точки переводится в «Приостановлена», объект уходит из поступлений исполнителей этой точки;
-  - запустить остановленную точку; статус точки переводится в «Открыта», объект появляется в поступлениях исполнителей этой точки;
-  - сохранить изменения, внесенные администратором в свойства точки;
-  - пропустить точку: передать исполнение следующей за ней точке. При пропуске точки нужно ввести комментарий; статус точки будет переведен в «Пропущена», исполнение перейдет к следующей точке (рис.7.2-3). Если после пропускаемой точки в маршруте предусмотрена развилка, то при ее пропуске необходимо выбрать, по какой из ветвей должен дальше идти объект.



The screenshot shows the 'Монитор процессов' (Process Monitor) interface. The main area displays a BPMN diagram with a task 'Согласование заявки инициатором' (Request approval by initiator) highlighted in red. Below it are two parallel tasks: 'Согласование юридическим отделом' (Approval by legal department) and 'Согласование финансовым отделом' (Approval by finance department). The right panel shows the configuration for the selected task, including completion date, executors, controllers, and task details.

**Исполнители**

Количество исполнителей:	Любое				
Пропуск точки без исполнителей:	Нет				
Режим исполнения:	Любой из исполнителей				
Заполнение:	По ФИО из справочника пользователей				
<b>Пользователи:</b>	<table border="1"> <thead> <tr> <th>ФИО</th> <th>Логин</th> </tr> </thead> <tbody> <tr> <td>Иванов Иван Иванович</td> <td>test1</td> </tr> </tbody> </table>	ФИО	Логин	Иванов Иван Иванович	test1
ФИО	Логин				
Иванов Иван Иванович	test1				

**Контролёры**

<b>Пользователи:</b>	<table border="1"> <thead> <tr> <th>ФИО</th> <th>Логин</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>	ФИО	Логин		
ФИО	Логин				



**Задача**

Идентификатор: genericTask  
Наименование: Настраиваемая задача  
Входные параметры:

Идентификатор	Наименование
form1	Редактирование заявки

Форма управляемого объекта: \*  
Маршрутные кнопки: \*  
Параметры завершения:  
Комментарий: Сотрудника нет на рабочем месте.

Рисунок 7.2-3. Вкладка «Описание точки» для пропущенного этапа маршрута

Добавление исполнителя или контролёра в исполняющейся точке происходит аналогично тому, как это реализовано в дизайнера процессов (рис.7.2-4). По кнопке  следует выбрать исполнителя, добавить его в таблицу «Исполнители» или «Контролёры» и нажать «Сохранить» . Сохраненные изменения будут автоматически применены к исполняющемуся процессу.



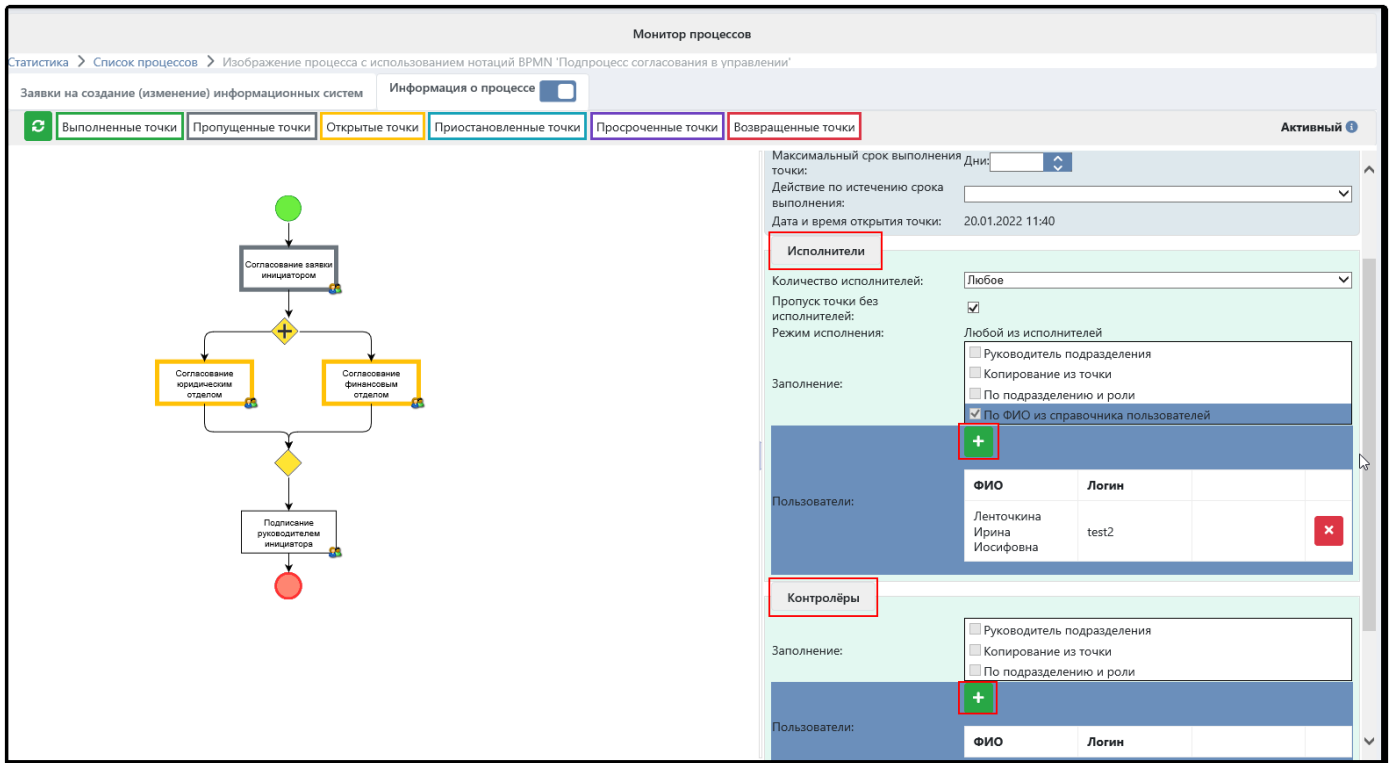



Рисунок 7.2-4. Добавление исполнителей и контролёров в исполняющуюся точку

Для удаления участника на незавершенной точке служит кнопка «Удалить» .

Для изменения условий переходов нужно выбрать элемент «шлюз» и в его настройках воспользоваться кнопкой «Редактировать»  (рис.7.2-5).

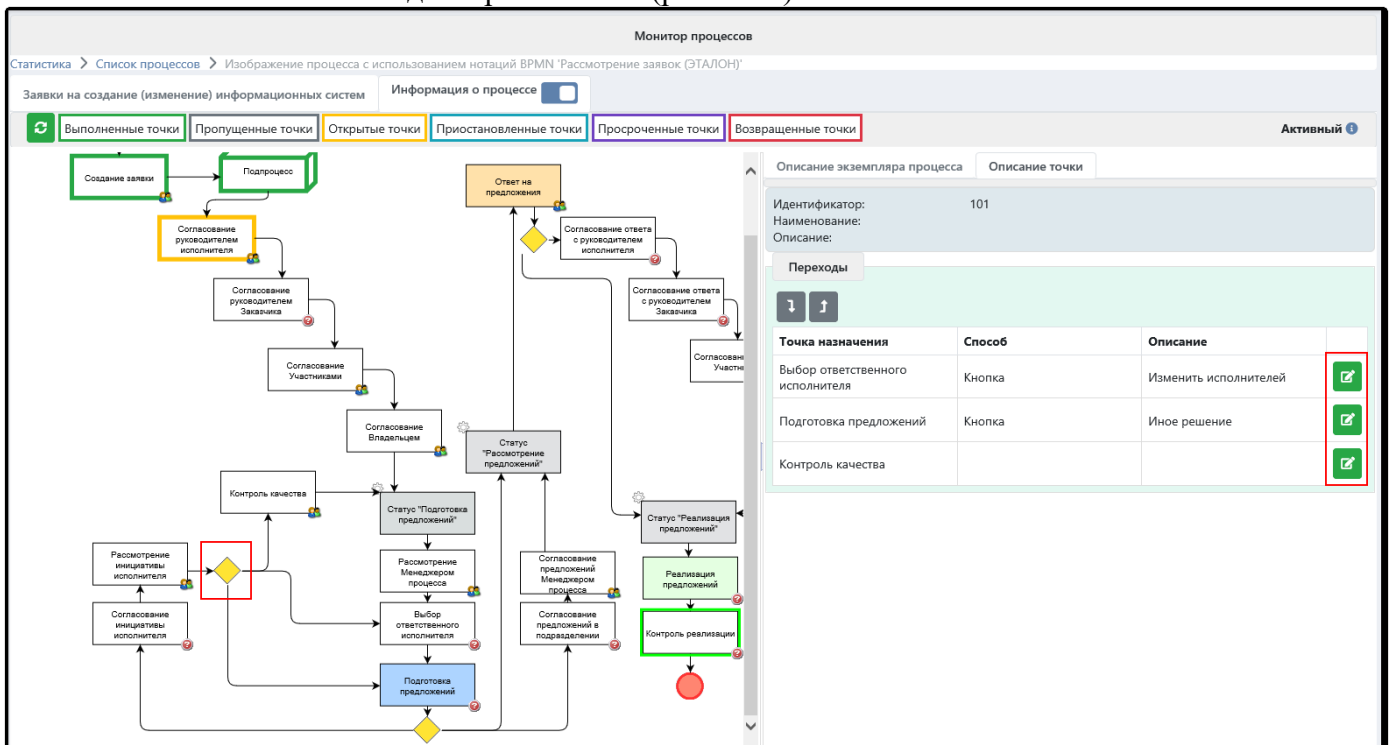






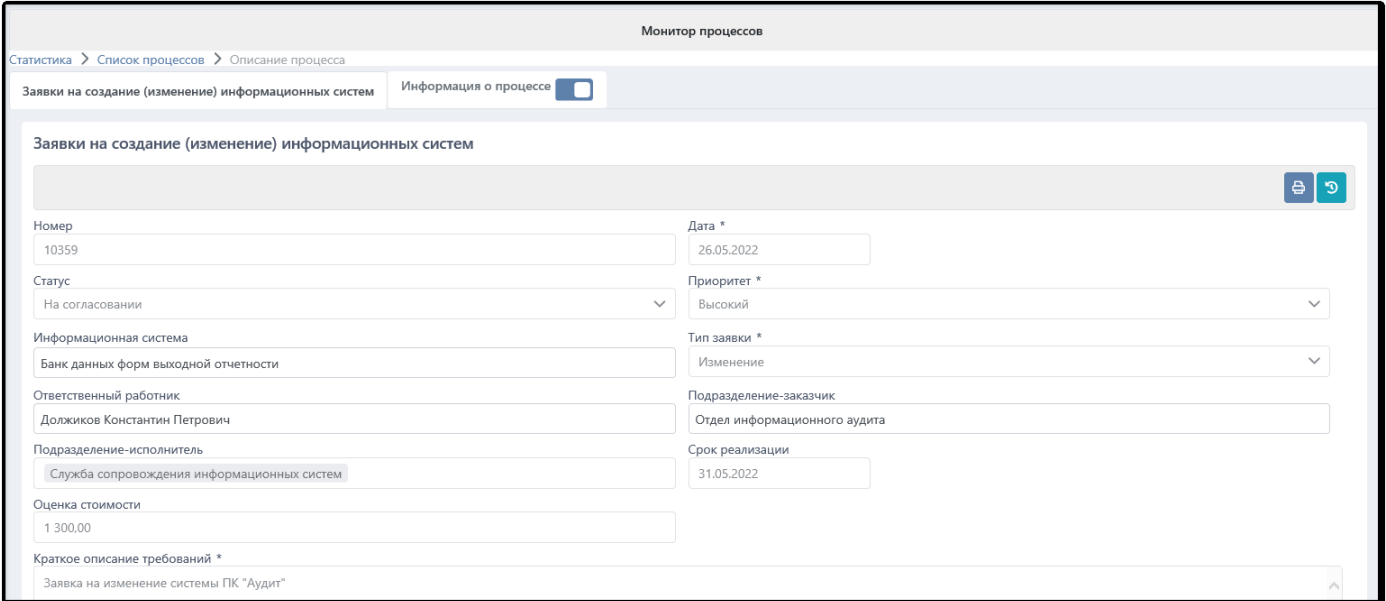
Рисунок 7.2-5. Изменение условий переходов в разветвлении маршрута

Аналогичным образом можно менять маршрутные кнопки в исполняющемся процессе, их состав и наименования.

**Важно:** перед редактированием какого-либо этапа маршрута его рекомендуется приостановить , затем сохранить внесённые изменения  и после этого запустить снова .


### 7.3. Сравнение редакций объектов

Сеанс монитора процессов позволяет не только администрировать процесс, но и просмотреть объект, относящийся к этому процессу. Если для объекта установлена опция «Хранить историю», то будет доступен сеанс просмотра редакций по кнопке .



Монитор процессов

Статистика > Список процессов > Описание процесса

Заявки на создание (изменение) информационных систем | Информация о процессе 

Заявки на создание (изменение) информационных систем

Номер: 10359 | Дата \*: 26.05.2022

Статус: На согласовании | Приоритет \*: Высокий

Информационная система: Банк данных форм выходной отчетности | Тип заявки \*: Изменение

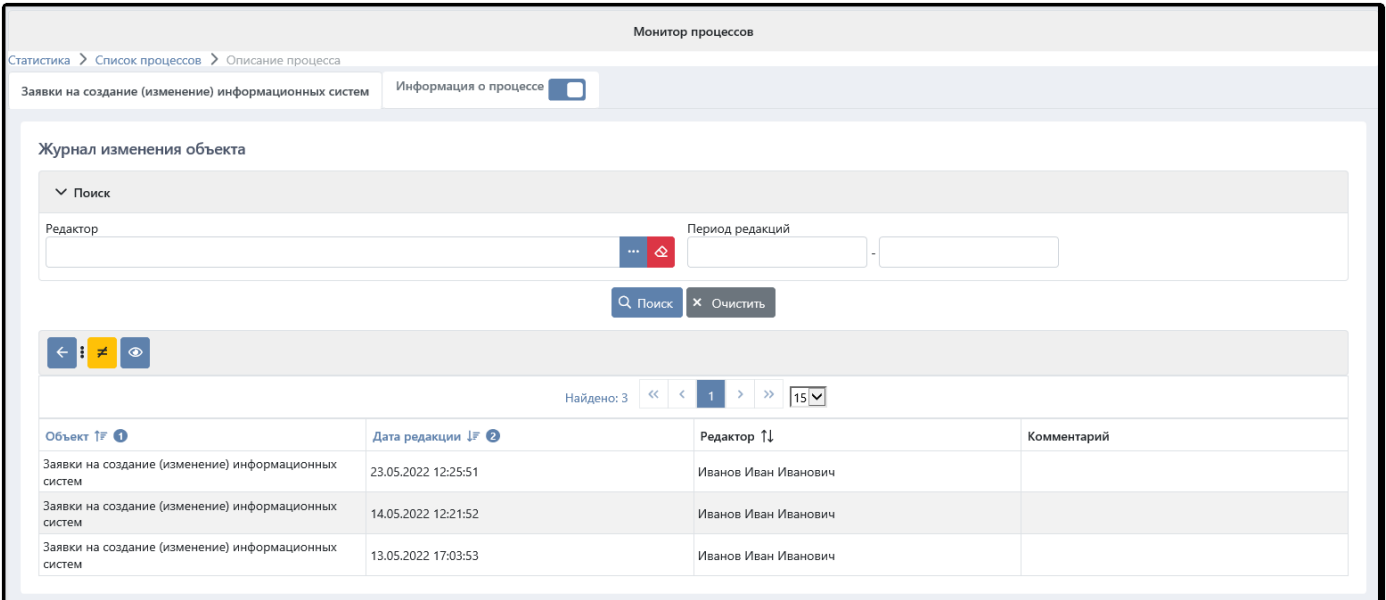
Ответственный работник: Должиков Константин Петрович | Подразделение-заказчик: Отдел информационного аудита

Подразделение-исполнитель: Служба сопровождения информационных систем | Срок реализации: 31.05.2022

Оценка стоимости: 1 300,00


Краткое описание требований \*: Заявка на изменение системы ПК "Аудит"

Рисунок 7.3-1. Просмотр объекта в мониторе процессов



Монитор процессов

Статистика > Список процессов > Описание процесса

Заявки на создание (изменение) информационных систем | Информация о процессе 

Журнал изменения объекта

Поиск: Редактор: [dropdown] | Период редакций: [start] - [end]

Поиск | Очистить

Найдено: 3 | 1 | 15




Объект 	Дата редакции 	Редактор 	Комментарий
Заявки на создание (изменение) информационных систем	23.05.2022 12:25:51	Иванов Иван Иванович	
Заявки на создание (изменение) информационных систем	14.05.2022 12:21:52	Иванов Иван Иванович	
Заявки на создание (изменение) информационных систем	13.05.2022 17:03:53	Иванов Иван Иванович	

Рисунок 7.3-2. Просмотр редакций объекта в мониторе процессов

Каждую редакцию можно открыть для просмотра. Кроме этого, редакции объектов можно сравнивать между собой: выбранная редакция сравнивается с предыдущей. Сеанс показывает изменения, которые были внесены в тот или иной реквизит.

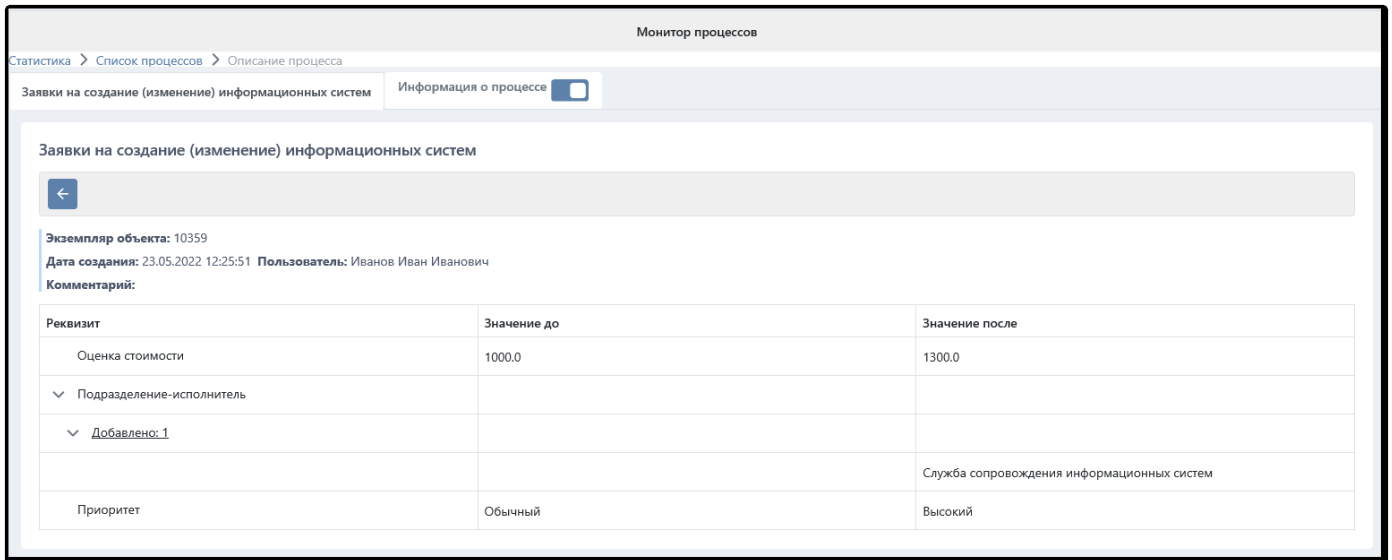


Рисунок 7.3-3. Сравнение редакций между собой

## 8. События и оповещения

### 8.1. Общие понятия

Под **событием** в ISIDA BPM понимается некое действие системы, значимое для участников бизнес-процесса (инициаторов, исполнителей, контролёров) или для сотрудников организации, не относящихся к процессу (начальников управлений, администраторов системы, сотрудников бухгалтерии и т.д.). Например, событиями могут быть: открытие точки маршрута, смена статуса объекта, прерывание бизнес-процесса, завершение бизнес-процесса, наступление срока исполнения точки, ввод отчета и т.д.

Под **выбросом события** понимается фиксация наступления некоторого события в реальном времени. **Слушатель событий** – это специальный программный компонент, который ловит все события, сохраняет в базе данных и запускает процесс их обработки.

Под **уведомлением о наступлении события** понимается текстовое описание произошедшего события, характеристик объекта, с которым это событие произошло, и ФИО инициатора этого события. В настоящее время ISIDA BPM поддерживает рассылку уведомлений только по электронной почте, в будущем планируется поддержка sms-оповещений и других каналов связи.

Под **получателями (подписчиками)** уведомлений понимаются пользователи ISIDA BPM, которые по каким-либо причинам хотят получать уведомления о наступлении событий. Получатели могут быть заданы программно, например, получателями уведомлений о событии открытия точки маршрута будут все исполнители этой точки. Также получатели могут задаваться администратором заранее при настройке конкретного события. Например, уведомления о смене прерывании автоматической точки маршрута должны получать администраторы (выбор роли или конкретных людей из справочника).

Под **регламентом обработки событий** понимается программный компоненты системы, обрабатывающий события по определенному расписанию. Сейчас в системе реализованы следующие регламенты:

- рассылка уведомлений на электронную почту сотрудников;
- удаление обработанных событий;
- контроль сроков исполнения документов.

ISIDA BPM различает следующие виды событий: **системные** и **прикладные**.

**Системными** считаются события, которые заранее встроены в алгоритмы работы системы, т.е. выбрасываются системой по умолчанию. Место их выброса, идентификаторы, наименования и другие параметры событий заданы программно. К таким событиям относятся, например: открытие точки маршрута, прерывание точки маршрута, приближение срока исполнения документа и т.д.

**Прикладные** события создаются вручную администратором при описании процесса, они тесно связаны с конкретными объектами и погружены в контекст решаемой задачи. Прикладные события наращиваются в процессе эксплуатации, для них создается описание, шаблон уведомления, настраиваются точки выброса, получатели уведомлений.

## 8.2. Справочник событий

Все события хранятся и настраиваются в специальном справочнике системы «Контрольные события».

Справочник событий			
Список событий			
Тип ↑ 1	Идентификатор ↑↓	Описание ↑↓	Статус ↑ 2
Прикладное	ActivityAgreedEvent	Объект согласован исполнителем	Опубликован
	ChildActivityAgreedEvent	Объект согласован в дочерней точке	Опубликован
	ChildActivityDisagreedEvent	Объект отклонен в дочерней точке	Опубликован
	ReportAddedEvent	Введен отчет в точке маршрута	Опубликован
	StatusChangedEvent	Изменен статус объекта	Опубликован
	StatusProjectTaskEvent	Проектный подход	Опубликован
Системное	ActivityLoopedEvent	Зациклилась автоматическая задача	Опубликован
	ActivityOpenedEvent	Открыта точка маршрута	Опубликован
	ActivitySkippedEvent	Точка пропущена в связи с истечением срока выполнения	Опубликован
	ActivitySuspendedEvent	Точка приостановлена в связи с истечением срока выполнения	Опубликован
	ActivityTerminatedEvent	Возврат в точке маршрута	Опубликован
	PlanTermApproachingEvent	Приближается срок планового события	Опубликован
	PlanTermCameEvent	Наступил срок планового события	Опубликован
	PlanTermChangedEvent	Изменение срока планового события	Опубликован
	PlanTermViolatedEvent	Превышен срок планового события	Опубликован
	TermApproachingEvent	Приближается срок исполнения точки	Опубликован
TermCameEvent	Наступил срок исполнения точки	Опубликован	
TermViolatedEvent	Превышен срок исполнения точки	Опубликован	



Рисунок 8.2-1. Справочник контрольных событий

По умолчанию в справочнике присутствуют следующие **системные** события:

- **Открыта точка маршрута (ActivityOpenedEvent):** событие генерируется в момент открытия пользовательской точки маршрута, в которой есть исполнители. Для автоматических точек не генерируется.
- **Возврат в точке маршрута (ActivityTerminatedEvent):** событие генерируется в момент возврата из текущей точки объекта, находящегося на процессе обработки.
- **Приближается срок исполнения точки (TermApproachingEvent):** генерируется специальным регламентом для тех открытых точек, срок исполнения которых наступит в ближайшие сутки от момента запуска регламента.
- **Наступил срок исполнения точки (TermCameEvent):** генерируется специальным регламентом для тех точек, которые еще не открыты, но срок исполнения уже наступил на момент запуска регламента. Для закрытых (пройденных) точек такое событие не генерируется.
- **Превышен срок исполнения точки (TermViolatedEvent):** генерируется специальным регламентом для открытых точек, срок исполнения которых наступил в предыдущие сутки от момента запуска регламента.
- **Приближается срок планового события (PlanTermApproachingEvent):** генерируется специальным регламентом для незавершенных плановых событий, срок исполнения которых наступит в ближайшие сутки от момента запуска регламента.

- **Наступил срок планового события (PlanTermCameEvent):** генерируется специальным регламентом для тех для незавершенных плановых событий, срок исполнения уже наступил на момент запуска регламента.
- **Превышен срок планового события (PlanTermViolatedEvent):** генерируется специальным регламентом для тех для незавершенных плановых событий, срок исполнения которых наступил в предыдущие сутки от момента запуска регламента.
- **Изменен срок планового события (PlanTermChangedEvent):** генерируется с момент изменения планового срока исполнителем точки маршрута.
- **Точка пропущена в связи с истечением срока выполнения (ActivitySkippedEvent):** событие генерируется в случае автоматического пропуска пользовательской точки маршрута из-за истечения срока ее выполнения. Настройка пропуска таких точек осуществляется в дизайнерах процессов в свойствах точки.
- **Точка приостановлена в связи с истечением срока выполнения (ActivitySuspendedEvent):** событие генерируется в случае автоматической остановки пользовательской точки маршрута из-за истечения срока ее выполнения. Настройка остановки таких точек осуществляется в дизайнерах процессов в свойствах точки.
- **Заиклилась автоматическая задача (ActivityLoopedEvent):** событие генерируется системой в случае, если процесс был приостановлен из-за заикливания какой-либо из задач. В этом случае администратору на почту будет отправлено соответствующее уведомление.

Каждое событие характеризуется **статусом**: опубликовано или черновик. Генерация (выброс) событий происходит только после их публикации. События открытия и возврата точки генерируются системой автоматически; события, связанные со сроками исполнения точек генерируются регламентами системы, администрирование которых описано [в разделе 8.5](#).

Для создания нового прикладного события нужно воспользоваться кнопкой . Заполнить все обязательные поля, тему и содержание уведомления, установить подписчиков. После создания события его нужно опубликовать (кнопка ). Выброс прикладных событий описан [в разделе 5.11](#) данного руководства.

### 8.3. Подписка на события

ISIDA BPM поддерживает два вида настройки подписки на события.

#### *Подписка в макете уведомления*

Для подписки можно выбрать, кто будет получателем уведомления: все исполнители открытой точки маршрута или все сотрудники из указанных ролей и дополнительного списка (рис.8.3-1,2).

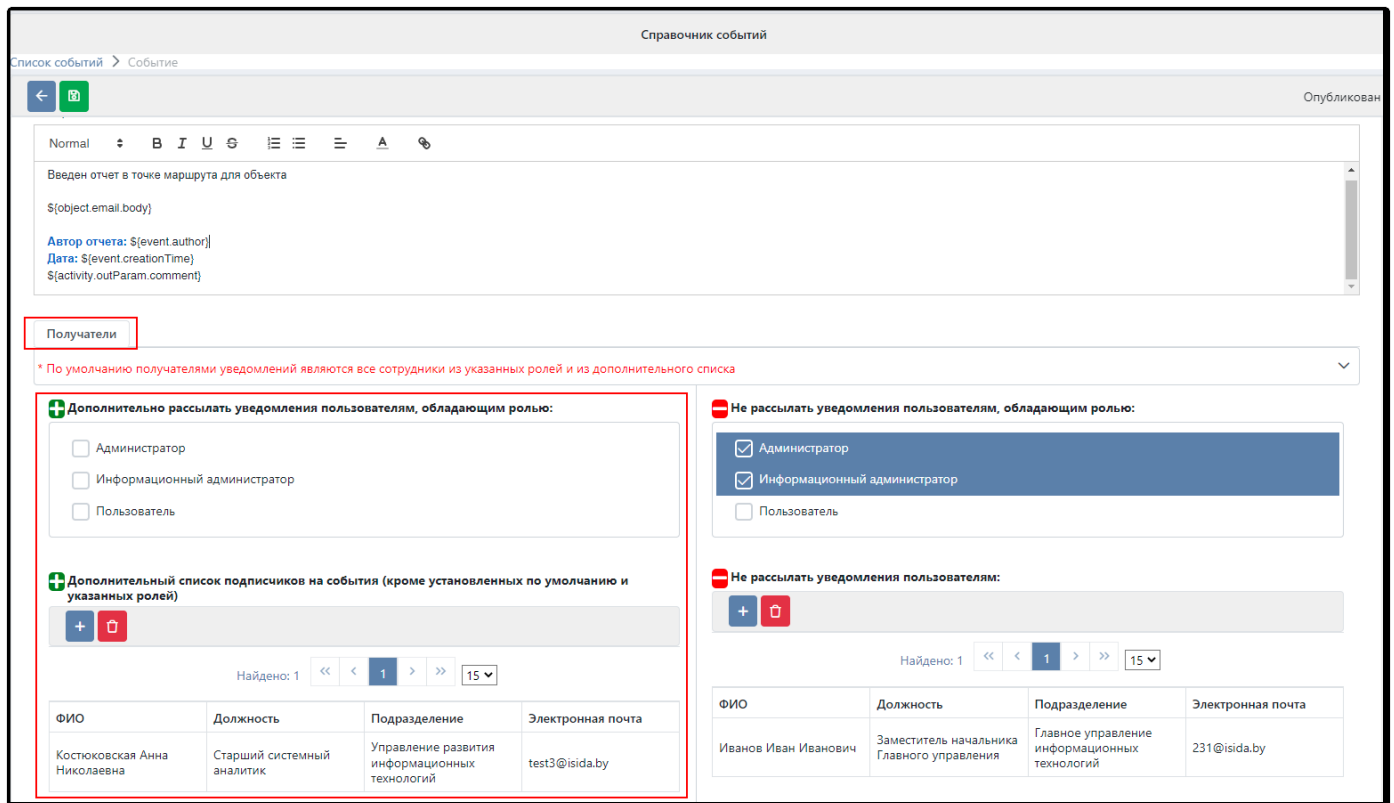


Рисунок 8.3-1. Подписка на событие в настройке макета уведомления

Установка режима получателей происходит с помощью выпадающего меню (рис.8.3-2).

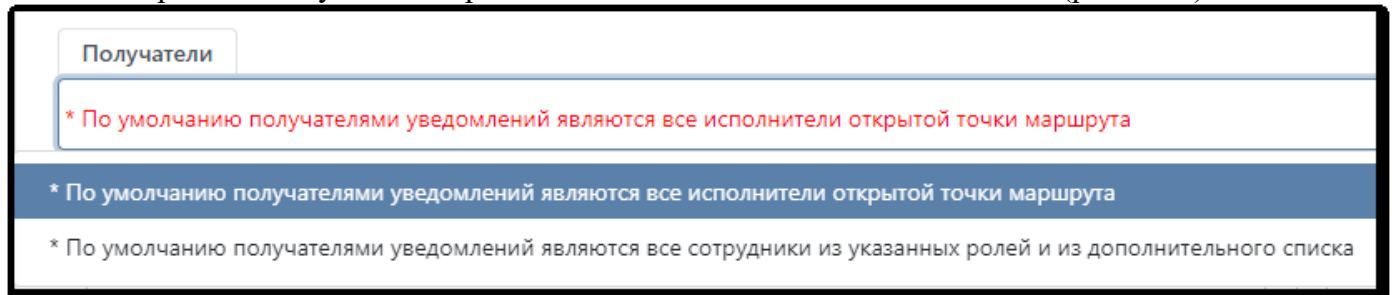


Рисунок 8.3-2. Выбор режима для установки получателей

### Подписка в сеансе настройки точки в дизайнера процессов

Любая ручная точка процесса может быть подписана на прикладные и системные события, которые выбрасываются другой точкой (рис.8.3-3,4).

Дизайнер процессов

Список процессов > Процесс 'Обработка заявок на создание (изменение) информационных систем'

Входные параметры:

Идентификатор	Наименование	Значение
status	Статус	Подготовка предложений

Выброс событий

Событие: Изменен статус объекта

Ожидание планового события

Точка: Статус "Рассмотрение предложений" (134)

Событие: Изменен статус объекта

Контрольный срок: 20 Рабочие дни

Описание планового события: срок подготовки предложений

Подписка на события

Событие	Подписчики
Приближается срок планового события	Исполнители выбранных точек: Подготовка предложений (8)
Наступил срок планового события	Исполнители выбранных точек: Создание заявки (3); Подготовка предложений (8)
Превышен срок планового события	Исполнители выбранных точек: Создание заявки (3); Подготовка предложений (8); Выбор ответственного исполнителя (55); Согласование руководителем исполнителя (71)

Рисунок 8.3-3. Настройка подписки на событие

### Подписка на событие

Событие: Превышен срок планового события

Способ назначения подписчиков:

- Исполнители выбранных точек
- По ФИО из справочника пользователей
- По роли

Точка маршрута:

- Создание заявки (3)
- Рассмотрение Менеджером процесса (7)
- Подготовка предложений (8)
- Ответ на предложения (9)

Сохранить Отменить

Рисунок 8.3-4. Настройка подписки на событие

Например, на событие «Превышен срок планового события», которое выбрасывается в точке «Статус «Подготовка предложений», подписаны все исполнители точек «Создание заявки», «Подготовка предложений».

Для событий, происходящих в дочерних точках, на которые должны быть подписаны родительские точки, предназначен способ назначения подписчиков: «Создатель дочерней точки» (рис.8.3-5).

Рисунок 8.3-5. Настройка подписки на событие, произошедшее в дочерней точке

#### 8.4. Настройка макета уведомления

Событие ISIDA BPM обладает двумя ключевыми характеристиками – это идентификатор и описание, а также блоком настроек для каждого канала оповещения: макет почтового уведомления и список подписчиков (рис.8.3-1).

Рисунок 8.3-1. Настройка уведомления для события

Основные разделы настроек:

**Тема письма:** текстовое сообщение, которое будет отображено в теме почтового уведомления.

**Содержание письма:** текстовое сообщение (текст почтового уведомления); может содержать подстановки для реквизитов объекта, свойств точек, срока исполнения и т.д. Применимы как для содержимого, так и для темы письма. Указываются в виде:

$\{Идентификатор\_объекта.Идентификатор\_поля\}$

Поддерживаются следующие группы подстановок:

- 1) Реквизиты **точки маршрута**, с которой связано событие: название задачи, описание, срок, сопроводительный комментарий и т.п.:  
*activity.name* - наименование точки маршрута;



*activity.description* - описание точки маршрута;  
*activity.term* - срок обработки точки в формате dd.ММ.уууу;  
*activity.executedBy* - ФИО исполнителя, который обработал точку;  
*activity.incParam.coveringNotes* - сопроводительный комментарий с ФИО;  
*activity.outParam.comment* – комментарий в точке.

- 2) Реквизиты **события**: автор события, дата и время:  
*event.creationTime* - дата и время события в формате dd.ММ.уууу НН:mm;  
*event.author* - ФИО автора события;  
*plan.event.description* – описание планового события;  
*plan.event.targetTime* – плановый срок события в формате dd.ММ.уууу;  
*plan.event.comment* - комментарий при переносе срока планового события.
- 3) Реквизиты **объекта**: краткое описание объекта, который связан с событием:  
*object.email.subject* – шаблон темы письма, настроенный в конструкторе объекта (описано в [разделе 2.4](#));  
*object.email.body* – шаблон содержания письма, настроенный в конструкторе объекта (описано в [разделе 2.4](#));  
*object.<поле объекта>* - поле объекта типа "строка", "число", "дата".
- 4) Ссылка на вход в систему:  
 $\${link.process.url}$

Поля объекта сложной структуры следует писать, указывая все реквизиты в иерархии через точку, например:

*Подразделение: \${object.department.name}*

*Информационная система: \${object.infSystem.fullName}*

Для полей **объекта** типа «дата» можно указывать формат дат. Например:  $\${object.date|dd.ММ.уууу}$  – дата объекта без учета времени,  $\${object.date}$  – дата объекта в формате dd.ММ.уууу НН:mm (с учетом времени).

Для полей **точек** типа «дата» указывается так:  $\${activity.term}$ . При этом, если в файле настроек *application.properties bpm.term.time=true*, в полях типа «дата» будет указано время.

Текст письма можно оформлять с помощью разных начертаний шрифтов: например, делать **жирным** или **выделять цветом**. Форматирование будет продублировано и в почтовом уведомлении получателя.

Например, если в макете указать:

#### Тема письма:

На ОБРАБОТКУ:  $\${object.email.subject}$

#### Содержание письма:

К вам на **ОБРАБОТКУ** поступила задача:  $\${activity.name}$

##### Реквизиты:

$\${object.email.body}$

**Сопроводительный комментарий:**  $\${activity.incParam.coveringNotes}$

**Срок исполнения задачи:**  $\${activity.term}$

В этом случае подписчик получит следующее уведомление:

#### Тема письма:

**Содержание письма:**

К вам на **ОБРАБОТКУ** поступила задача: *Визирование исполнителем.*


**Реквизиты:****Номер:** 40405**Дата:** 01.07.2022**Содержание:** Заявка на интеграцию с ПК «Персонал»**Тип заявки:** Изменение**Приоритет:** Обычный**Статус:** На согласовании**Подразделение:** Управление информационных технологий**Информационная система:** ISIDA DMS**Сопроводительный комментарий:** Прошу рассмотреть до указанного срока**Срок исполнения задачи:** **31.07.2022**

Раздел **«Получатели»** позволяет установить перечень сотрудников, которым следует получать уведомления. По умолчанию это все исполнители точки, для которой было сгенерировано событие, кроме самого инициатора этого события. Кроме этого, можно добавить дополнительных получателей из справочника сотрудников, а также отметить роли, которым необходимо или наоборот, не нужно рассылать уведомления. Система отправляет письмо каждому подписчику события, у которого в справочнике событий указан адрес электронной почты. В случае, если исполнитель точки маршрута является исполнителем предыдущей точки, уведомление ему не поступает.

Для использования ссылок в тексте письма для входа в систему, в *application.properties* следует указать:

*admin.event.link.process.url=http://xocm:nopm/bpm/studio-task/ExternalTaskViewInit.xhtml?t=*

В макете нужно сначала написать текст, который станет ссылкой, например «Обработать заявку».

Затем выделить этот текст и нажать на иконку . В открывшемся окне вписать *#{link.process.url}* и после этого нажать «Сохранить».

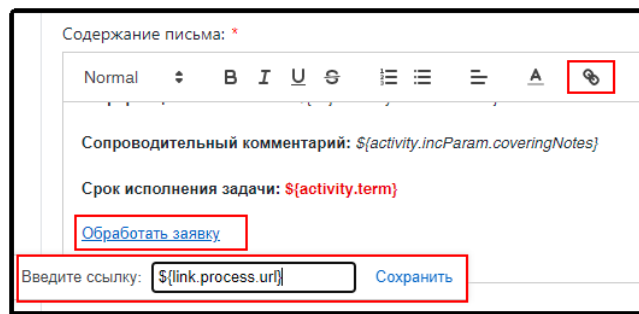



Рисунок 7.3-2. Использование ссылок в письме

После настройки и сохранения уведомления, событие нужно опубликовать (кнопка .

Все настройки, указанные для макета конкретного события, сохраняются в базе данных **в момент его генерации**. Если настройки текста почтового уведомления изменены, то они применяются только после того, как будет сгенерировано новое событие. Для событий, созданных до внесения правок в настройки письма, будут формироваться уведомления со «старыми» настройками.

## 8.5. Регламенты для работы с событиями и рассылка уведомлений

Для рассылки уведомлений на почтовые адреса пользователей, а также для генерации событий, связанных со сроком исполнения точек предназначен сеанс «Планировщик заданий» в разделе меню «Администрирование» (рис.8.4-1).

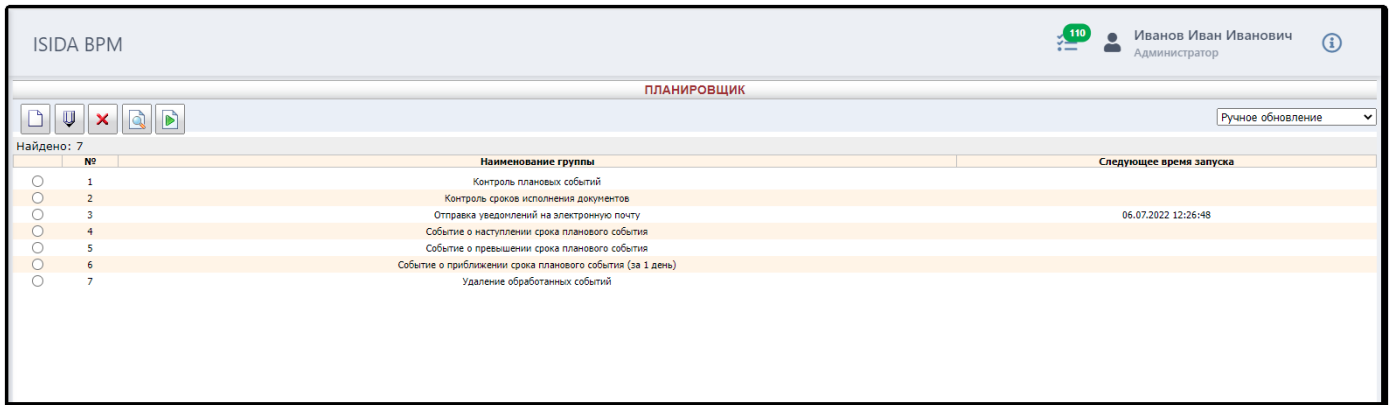


Рисунок 8.4-1. Сеанс «Планировщик заданий»

Сеанс позволяет создать регламент, т.е. некий исполняемый сценарий, настроить для него расписание запуска и указать одно или несколько заданий, которые регламент будет выполнять. Например, регламент «Контроль сроков исполнения документов» исполняет сразу три задачи, связанные с обработкой сроков исполнения (рис.8.4-2).

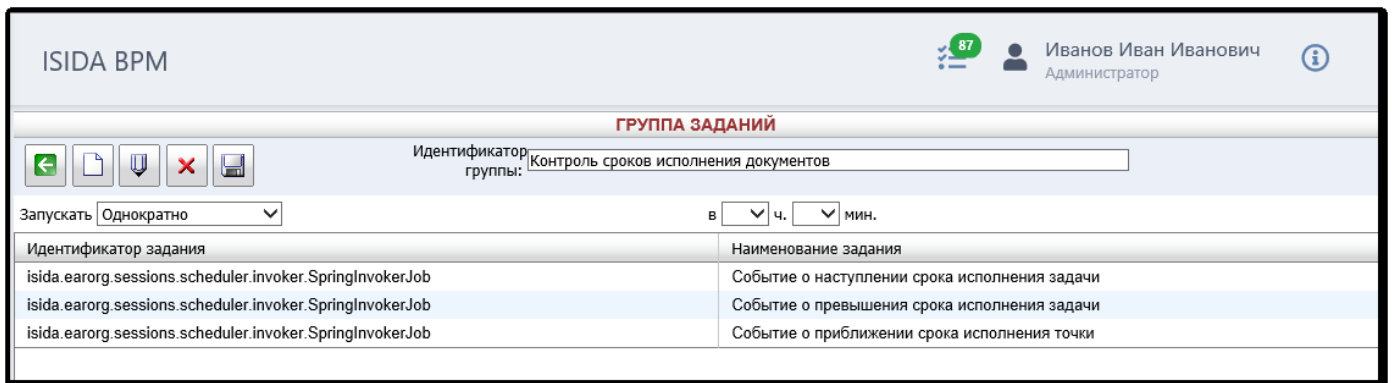


Рисунок 8.4-2. Настройки регламента «Контроль сроков исполнения документов»

При создании нового регламента нужно указать его идентификатор (русскоязычное наименование) и добавить задания из предложенного перечня (рис.8.4-3).

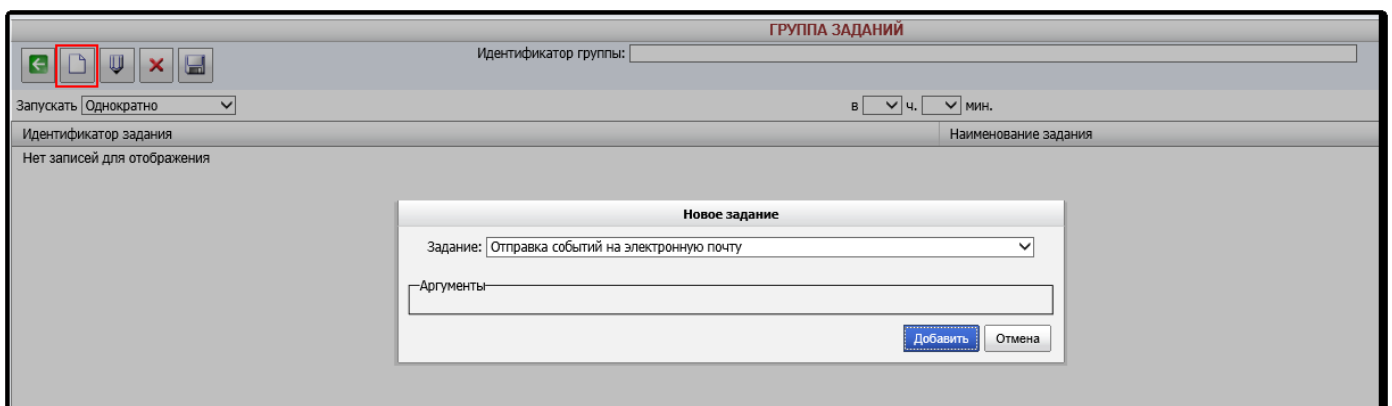


Рисунок 8.4-3. Выбор задания для нового регламента

На данный момент в ISIDA BPM реализованы следующие задания:

#### **Отправка событий на электронную почту**

Регламент последовательно отбирает события со статусом NEW, формирует почтовые уведомления с учетом данных из поля *f\_data* и рассылает получателям (исполнителям этой точки и тем сотрудникам, кто подписан на это событие). После успешной обработки событие принимает статус PROCESSED и больше не обрабатывается (т.е. повторно письма по этим событиям не рассылаются). Если событие обработано с ошибкой, оно принимает статус ERROR.

### Удаление обработанных событий

Регламент последовательно удаляет события из таблицы *admin\_events*, со статусом, указанным в настройках регламента, например, PROCESSED. Если указать «ALL», будут удалены события с любыми статусами, в том числе ERROR. Регламент позволяет также задать количество дней хранения событий. Например, если указано «7», регламент пропустит события, созданные в течение последней недели, но удалит все те, что были созданы до наступления прошедшей недели.

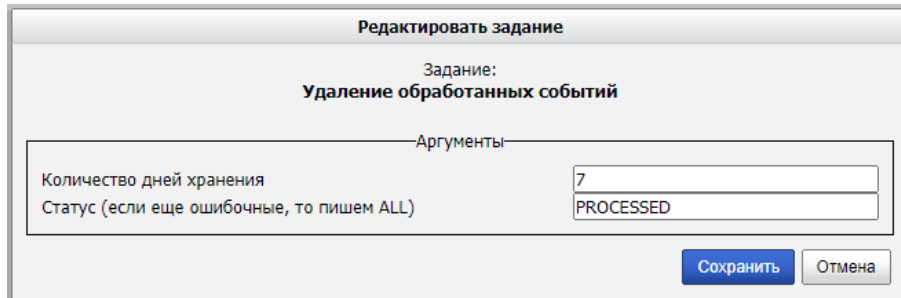


Рисунок 8.4-4. Настройка задания

### Генерация событий о приближении срока исполнения точки

Регламент последовательно проходит по всем открытым точкам всех объектов, и, если срок исполнения точки наступает в ближайшие сутки, создает новое событие *TermApproachingEvent*.

### Генерация событий о наступлении срока исполнения точки

Регламент последовательно проходит по всем точкам процесса, которые еще не открыты и не обработаны, но срок исполнения которых уже наступил (на момент запуска регламента). Генерирует для каждой такой точки новое событие *TermCameEvent*. Для закрытых (пройденных) точек такое событие не генерируется.

### Генерация событий о превышении срока исполнения точки

Регламент последовательно проходит по всем открытым точкам процесса, срок исполнения которых наступил в предыдущие сутки от момента запуска регламента. Генерирует для каждой такой точки новое событие *TermViolatedEvent*.

### Генерация событий о приближении срока планового события

Регламент последовательно проходит по всем незавершенным плановым событиям, срок исполнения которых наступит в ближайшие сутки. Генерирует новое событие *PlanTermApproachingEvent*.

### Генерация событий о наступлении срока планового события

Регламент последовательно проходит по всем незавершенным плановым событиям, срок исполнения которых уже наступил на момент запуска регламента. Генерирует новое событие *PlanTermCameEvent*.

### Генерация событий о превышении срока планового события

Регламент последовательно проходит по всем незавершенным плановым событиям, срок исполнения которых наступил в предыдущие сутки от момента запуска регламента. Генерирует новое событие *PlanTermViolatedEvent*.

### Отслеживание наступлений планируемых событий

Регламент последовательно проходит по всем незавершенным плановым событиям и ищет для них событие, которое они ожидают. Если такое событие найдено (то есть ожидаемое событие наступило), то регламент в записи о плановом событии меняет статус со значения «Ожидается» на «Наступило» и фиксирует дату наступления этого события.

Задания создаются командой разработчиков и добавляются в файл *quartz\_data.xml*:

*dss-компонент\install\wildfly\configuration\itrequest\main\quartz\_data.xml*

Файл XML обновляет администратор ISIDA BPM, следуя указаниям по развёртыванию, полученным вместе с поставкой обновлений системы.

Для каждого регламента должно быть настроено *правило запуска* (время, день недели, в какое он будет автоматически выполняться). Правила настройки расписания запуска регламентов отображены на рисунке 8.4-4.

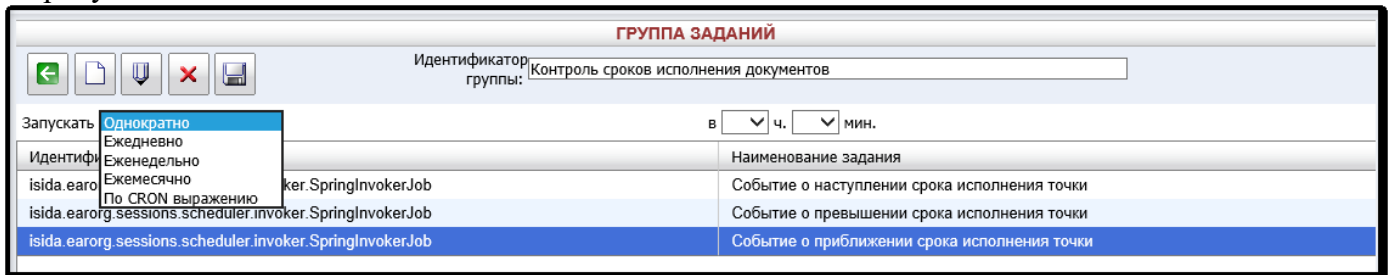



Рисунок 8.4-5. Установка правила запуска регламента

**Однократно:** регламент по умолчанию «выключен» и ручной запуск может быть произведен по кнопке . После запуска регламент также «выключен».

**Ежедневно:** запуск регламента осуществляется каждый день в заданное время: в  ч.  мин.

**Еженедельно:** запуск регламента осуществляется каждую неделю в заданное время в указанные администратором дни:

в  ч.  мин.  
 Пн  Вт  Ср  Чт  Пт  Сб  Вс

**Ежемесячно:** запуск регламента осуществляется в заданное время в указанные администратором дни месяца:

в  ч.  мин.  
 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19   
 20  21  22  23  24  25  26  27  28  29  30  31  В последний день

**По CRON выражению:** в формате «Секунды» «Минуты» «Часы» «День месяца» «Месяц» «День недели» «Год»

CRON выражения поддерживают специальные символы:

\* - все значения в пределах одного фрагмента cron-выражения. Например, в поле «Минуты» специальный символ \* обозначает, что задача будет выполняться каждую минуту;

? - неопределенное значение. Если необходимо, чтобы задача запускалась, к примеру 10 числа каждого месяца, но не важно, в какой день недели, то в поле «День месяца» необходимо установить значение «10», а в поле «День недели» - ?;

-- определение диапазонов. Например, «10-12» в поле «Часы» означает часы: 10, 11 и 12.

*Примеры cron-выражений:*

0 15 10 ? \* \* (запуск задачи в 10.15 каждый день);

0 15 10 ? \* 1-5 (запуск задачи каждый день с понедельника по пятницу включительно);

0 15 10 1 \* ? (запуск задачи в 10.15 первого числа каждого месяца);

0 0/15 \* \* \* ? (запуск задачи каждые 15 минут каждого часа каждый день).

Для подключения к серверу электронной почты в конфигурационном файле *application.properties* предусмотрены специальные параметры. Они описаны в [Приложении 1](#) данного руководства.

## Приложение 1. Параметры настроек application.properties

Параметр	Описание	Пример заполнения
<b>Параметры приложения</b>		
<i>studio.application.name</i>	Наименование приложения	ISIDA BPM
<i>studio.datasource.hibernate.mapping</i>	Каталог хранения файлов конфигурации для мапинга	mappings
<i>studio.object.form.mapping</i>	Каталог хранения файлов маппинга форм	forms
<i>studio.object.mark.deleted</i>	Помечать экземпляры объектов как удаленные	true/false
<i>studio.sql.names.maxlength</i>	Максимальная длина имен таблиц и колонок в БД	30
<i>studio.report.spoolDirectory</i>	Абсолютный путь к каталогу спула	C:/Programs/wildfly-20.0.1.Final-bpms/standalone/tmp
<i>studio.report.jasper.path.templates</i>	Абсолютный путь к каталогу, в котором хранятся шаблоны (JasperReport). Каталог должен существовать.	C:/Programs/wildfly-20.0.1.Final-bpms/bpm/templates
<i>studio.docs.usage.directory</i>	Абсолютный путь к каталогу, в котором хранится руководство пользователя	C:/Programs/wildfly-20.0.1.Final-bpms/modules/configuration/bpm/main/docs/usage
<i>bpm.task.process.incoming.group.field</i>	Группировка в поступлениях по полю «Русское наименование задачи» (taskName). По умолчанию группировка не происходит (если параметр не указан)	taskName
<b>BPM Core</b>		
<i>core.library.jsftemplate.LifetimeBean.rendered</i>	Настройки локализации приложения	true
<i>core.library.jsftemplate.LanguageBean.locale</i>	Настройки локализации приложения Доступные локали: ru, en	ru
<i>org.quartz.scheduler.instanceName</i>	Для запуска заданий с разных серверов. На каждом сервере должен быть настроен свой instanceName	QuartzScheduler
<i>org.quartz.threadPool.threadCount</i>	Количество потоков для запуска с разных серверов	10
<b>Настройки БД</b>		
<i>bpms.datasource.name</i>	Имя API для получения соединения	HBDataSource
<i>bpms.datasource.hibernate.dialect</i>	Диалект, который будет использоваться в приложении, зависит от типа используемой БД которая подключена к приложению	org.hibernate.dialect.SQLServerDialect
<i>bpms.datasource.hibernate.sql.show</i>	Вывод запросов отправляемых к БД в консоль	false
<i>bpms.cors.urls</i>	Открыть доступ к /api/rest клиенту, отправляющему запрос с данных источников	https://host1:port1,https://host2:port2
<b>Настройки BPM</b>		
<i>bpm.instanceId</i>	Идентификатор сервера, записываемый в процесс при его создании	dev1
<i>bpm.report.spoolDirectory</i>	Абсолютный путь к каталогу спула	C:/Programs/wildfly-20.0.1.Final-bpms/bpm/spool
<i>bpm.report.jasper.path.templates</i>	Абсолютный путь к каталогу, в котором хранятся шаблоны (JasperReport). Каталог должен существовать.	C:/Programs/wildfly-20.0.1.Final-bpms/bpm/templates
<i>bpm.flowcore.dispatcher.ClosingExpiredTaskService.enable</i>	Включить сервис автоматического закрытия просроченных точек	true
<i>bpm.dataObjectViewUrl.default</i>	Просмотр управляемого объекта без редактирования. Добавляются к ссылке параметры: System.currentTimeMillis() + "&objectId=?&objectPid=?"	/studio-object-form-generator/ObjectEdit.xhtml?mode=VIEW&t=
<i>bpm.dictionary.core.DictionaryCoreService.fieldNameId</i>	Название поля объекта с первичным ключом. Используется при работе с объектами	pk_pid

<i>bpm.flowcore.procedure.GenericTask.default.form</i>	Форма умолчанию	form1
<i>bpm.flowcore.procedure.GenericTask.default.form1.url</i>	Адрес стандартной формы отображения объекта если параметр "Форма управляемого объекта"(formId) отсутствует	/studio-object-form-generator/ObjectEdit.html
<i>bpm.flowcore.procedure.GenericTask.default.form1.name</i>	Название формы по умолчанию	Редактирование
<i>bpm.flowcore.workspace.noTransitionRedirectUrl</i>	Адрес, на который будет осуществляться перенаправление если не произошло открытие следующей точки маршрута	/bpm/studio-object-form-generator/TaskList.html
<i>bpm.term.time</i>	Учитывать время при расчете и отображении срока	true
<i>bpm.task.process.info.tab.available.role</i>	Видимость вкладки "Информация о процессе" в зависимости от идентификатора роли исполнителя ("*" - всем доступна)	*
<i>bpm.task.process.my.task.buttons.available.role</i>	Видимость кнопок секции "Мои задачи" в зависимости от идентификатора роли пользователя ("*" - всем доступна)	3
<i>bpm.task.process.incoming.group.field</i>	Группировка поступлений по наименованию задачи. По умолчанию группировки нет.	taskName
<b>Настройки работы с электронными документами</b>		
<i>bpm.flowcore.dispatcher.IProcessObjectService.eDocumentProcess</i>	ВРМ использует не только объекты, но и ЭД	true
<i>bpm.flowcore.procedure.GenericTask.form.\$edocument\$=form1,form2,form3</i>	Перечисление форм для электронного документа	
<i>bpm.flowcore.procedure.GenericTask.form.form1.\$edocument\$.url=/studio-object-form-generator/ObjectEdit.html?formId=form1</i> <i>bpm.flowcore.procedure.GenericTask.form.form1.\$edocument\$.name=Создание электронного документа</i> <i>bpm.flowcore.procedure.GenericTask.form.form2.\$edocument\$.url=/studio-object-form-generator/ObjectEdit.html?formId=form2</i> <i>bpm.flowcore.procedure.GenericTask.form.form2.\$edocument\$.name=Просмотр электронного документа</i> <i>bpm.flowcore.procedure.GenericTask.form.form3.\$edocument\$.url=/studio-object-form-generator/ObjectEdit.html?formId=form3</i> <i>bpm.flowcore.procedure.GenericTask.form.form3.\$edocument\$.name=Редактирование электронного документа</i>		
<b>Настройка почтовых уведомлений</b>		
<i>admin.event.channel.email.smtp.hostname</i>	Имя почтового хоста	hostname
<i>admin.event.channel.email.smtp.fromAddress</i>	Адрес, с которого будут рассылаться уведомления	bpm@host.org
<i>admin.event.channel.email.smtp.fromName</i>	Наименование «От кого» в почтовом уведомлении	ISIDA BPM
<i>admin.event.channel.email.smtp.charset</i>	Кодировка сообщений	UTF-8
<i>admin.event.channel.email.smtp.port</i>	Порт	25
<i>admin.event.channel.email.smtp.login</i>	Логин пользователя для доступа к почтовому аккаунту	test1
<i>admin.event.channel.email.smtp.password</i>	Пароль пользователя для доступа к почтовому аккаунту	test1
<i>admin.event.channel.email.smtp.timeout</i>	Тайм-аут соединения и тайм-аут чтения (в мс)	600000
<i>admin.event.channel.email.smtp.tls</i>	Включить защищенную передачу данных (ключ <i>admin.event.channel.email.smtp.ssl</i> должен быть "false")	false
<i>admin.event.channel.email.smtp.ssl</i>	Включить защищенную передачу данных (ключ <i>admin.event.channel.email.smtp.tls</i> должен быть "false")	false
<i>admin.event.channel.email.smtp.ssl.port</i>	Номер порта для ssl соединения	587
<i>admin.event.link.env.contextUrl</i>	Вставка внешней ссылки в письмо	https://host/context
<i>admin.event.link.process.url</i>	Вставка ссылки для перенаправления на задачу в письмо	http://vm-bpm-task.isida.local:8080/bpm/studio-task/ExternalTaskViewInit.html?t=