

RunaWFE Free. Практикум. Использование возможностей RunaWFE для решения задач

Версия 4.6.2

© 2015-2026, ООО "Процессные технологии"

Содержание

1. Цель занятия.....	1
2. Списки исполнителей.....	1
2.1. Инициализатор роли - "Исполнители из списка".....	2
2.2. Обработчики получения списка исполнителей.....	3
2.2.1. Получить список исполнителей группы.....	4
2.2.2. Получить список исполнителей по отношению.....	4
2.3. Пример работы со списками исполнителей.....	5
3. Полезные обработчики.....	12
3.1. Получение номера текущего экземпляра процесса.....	12
3.2. Получение текущей даты-времени.....	14
3.3. Получение ФИО пользователя и использование падежей.....	15
3.4. Использование формулы-обработчика для получения случайного числа (датчик или генератор случайных чисел).....	18
3.5. Использование формулы-обработчика для создания выводимых на форму задания текстов.....	19
4. Режим последовательного показа форм заданий.....	19
5. Отображение переходов по имени.....	22
6. Использование замещений.....	23
7. Ссылки.....	26

1. Цель занятия

Целью данного занятия является изучение возможностей RunaWFE Free для решения задач.

2. Списки исполнителей

В бизнес-процессах производится связывание узлов-действий с исполнителями заданий при помощи ролей. Во время выполнения бизнес-процесса ролям назначаются конкретные исполнители. Процедура назначения исполнителей на роль называется инициализацией роли.

В качестве инициализаторов роли могут быть использованы:

- исполнитель по имени
- исполнитель по коду
- такие функции над оргструктурой как "Руководитель", "Директор"
- инициализация с помощью отношения
- кастомизированные инициализаторы (пользователи и группы Active Directory) и т.д.

2.1. Инициализатор роли - "Исполнители из списка"

В версии RunaWFE Free был добавлен новый инициализатор роли - "Исполнители из списка", позволяющий инициализировать роль списком исполнителей (переменной формата *Список(Исполнитель)*). В отличие от инициализации роли конкретной группой, в данном случае есть возможность формировать (добавление, удаление исполнителей) список в процессе выполнения БП.

Для редактирования инициализатора роли, выделите роль и нажмите "Изменить".

Новый инициализатор доступен в общем списке инициализаторов на вкладке "Код"

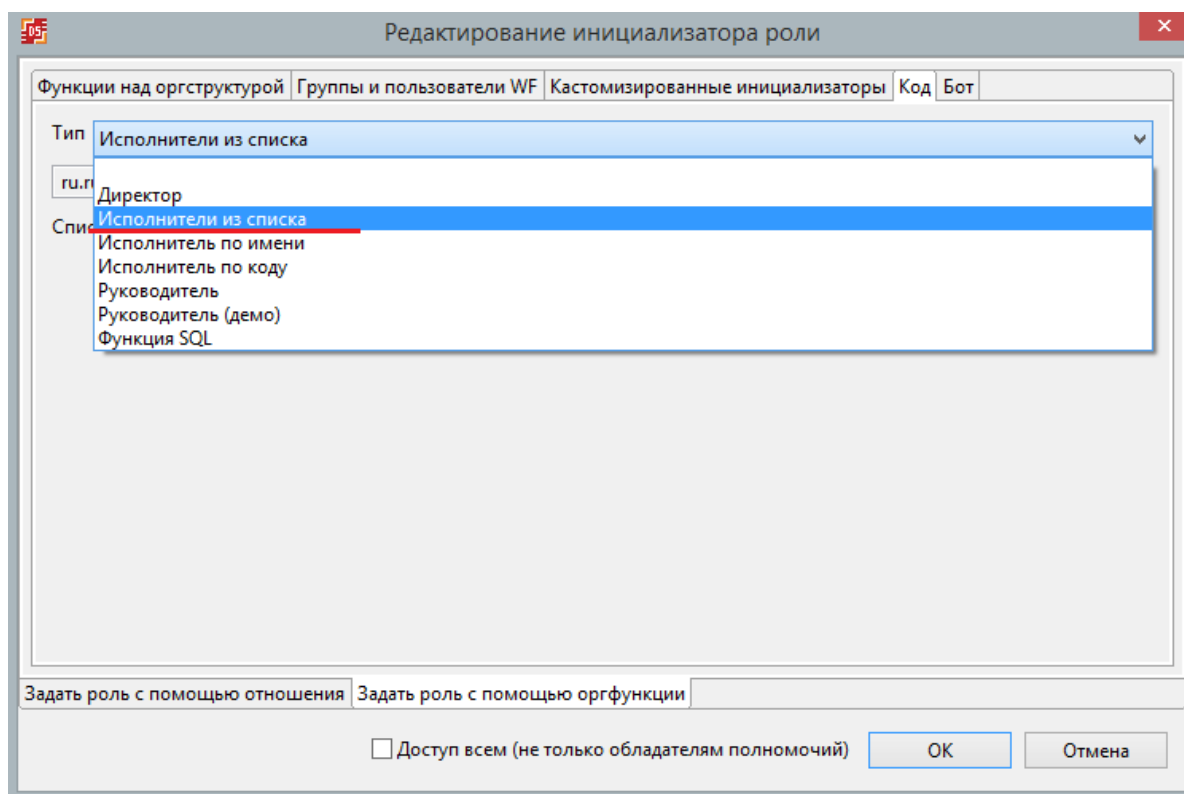


Рисунок 2.01. Выбор инициализатора "Исполнитель из списка"

В поле "Список исполнителей" задается переменная список, содержащая исполнителей.

Используется следующий синтаксис $\{Список\}$

Редактирование инициализатора роли

Функции над оргструкту... Группы и пользователи WF Кастомизированные иниц... Код Бот

Тип Исполнители из списка

ru.runa.wfe.extension.orgfunction.ExecutorsFromListFunction

Список исполнителей *: \$(Список)

Задать роль с помощью отношения Задать роль с помощью оргфункции

☐ Доступ всем (не только обладателям полномочий)

OK Отмена

[illegible]

2.2. Обработчики получения списка исполнителей

Для получения списка исполнителей, входящих в группу, в RunaWFE Free были добавлены обработчики - "Получить список исполнителей группы" и "Получить список исполнителей по отношению".

2.2.1. Получить список исполнителей группы

В качестве входного параметра обработчик принимает группу, исполнителей которой необходимо получить. Группа может быть задана переменной формата "*Группа*" или "*Строка*".

В выходном параметре "Результат" должна использоваться переменная формата *Список(Исполнитель)*. Ниже приведен пример конфигурации обработчика

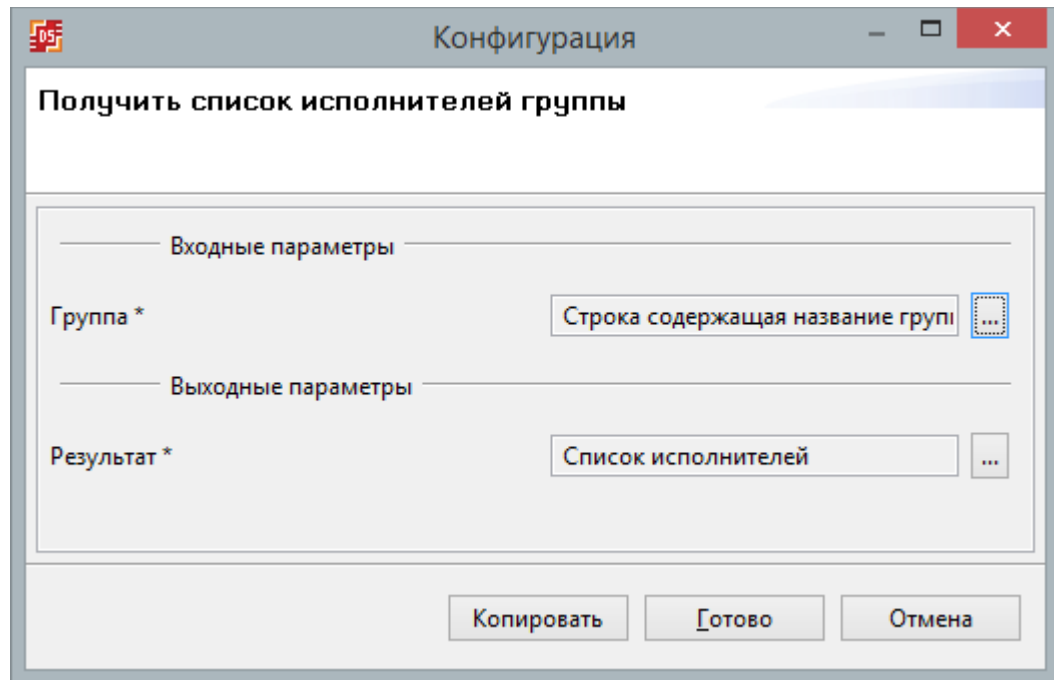


Рисунок 2.04. Конфигурация обработчика "Получить список исполнителей группы"

2.2.2. Получить список исполнителей по отношению

Входные параметры:

- "название отношения" - может быть задано как с помощью строковой переменной (содержащей название отношения), так и с помощью ввода названия отношения непосредственно в конфигураторе

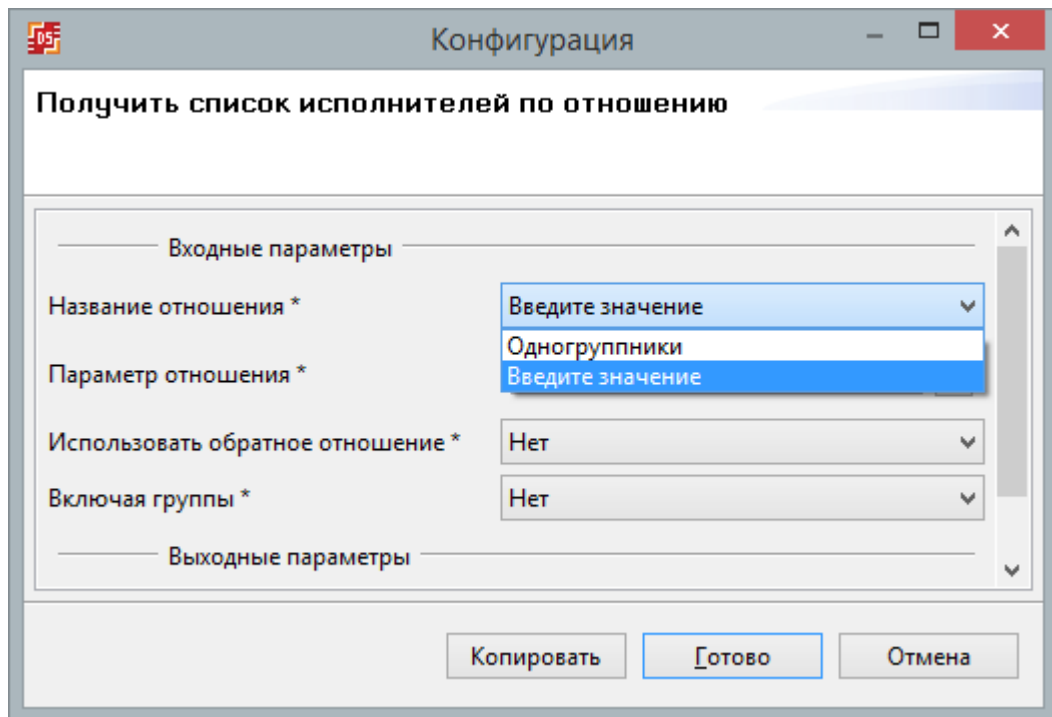


Рисунок 2.05. Конфигурация обработчика "Получить список исполнителей по отношению"

- "параметр отношения" - аналогично другим обработчикам, работающим с отношениями, в зависимости от выбранного значения в поле "Использовать обратное отношение" параметр используется как правая или левая часть в отношении.

Может быть задан с помощью роли или переменной формата *Исполнитель*

- "использовать обратное отношение" - Да/Нет
- "включая группы" - в случае "Нет" из списка исполнителей выбранных по отношению будут исключены группы.

Выходной параметр "Результат" - переменная формата Список(Исполнитель), будет содержать список исполнителей выбранных по отношению.

2.3. Пример работы со списками исполнителей

Рассмотрим пример использования вышеописанных обработчиков. Используем для этого бизнес-процесс "Игра в контакт".

В бизнес-процессе есть следующие роли.

- Ведущий
- Спрашивающий
- Контакттер
- Команда.

Ведущий запускает экземпляр бизнес-процесса. В стартовой форме он вводит загаданное слово и его первую букву. После этого все остальные члены команды (определяются членством в группе "Группа МИБ-1" за исключением "Ведущего", если он

является членом этой группы) получают задание "Придумать вопрос". В форме этого задания пользователь вводит вопрос и ответ на него. После того, как кто-то из членов команды выполнит задание (**Спрашивающий**), следующее задание "Установить контакт" получают все остальные отгадывающие (определяются отношением «Одногруппники», примененным к выполнившему заданию, за исключением "Ведущего" и "Спрашивающего"). В форме задания "Установить контакт" пользователь (**Контактер**) должен ввести свой ответ на вопрос, начинающийся на данную букву.

Не будем останавливаться на правилах подсчета результатов игры, а также опустим разработку форм, валидаторов и т.п., рассмотрим лишь формирование списков исполнителей и инициализаторы ролей.

После старта бизнес-процесса под ролью "Ведущий", должна быть сформирована "Команда" состоящая из членов группы "Группа МИБ-1" за исключением "Ведущего". Для реализации этого создадим переменную формата *Группа* и проинициализируем ее значением по умолчанию "Группа МИБ-1", а также переменную "Список исполнителей" формата *Список(Исполнитель)*.

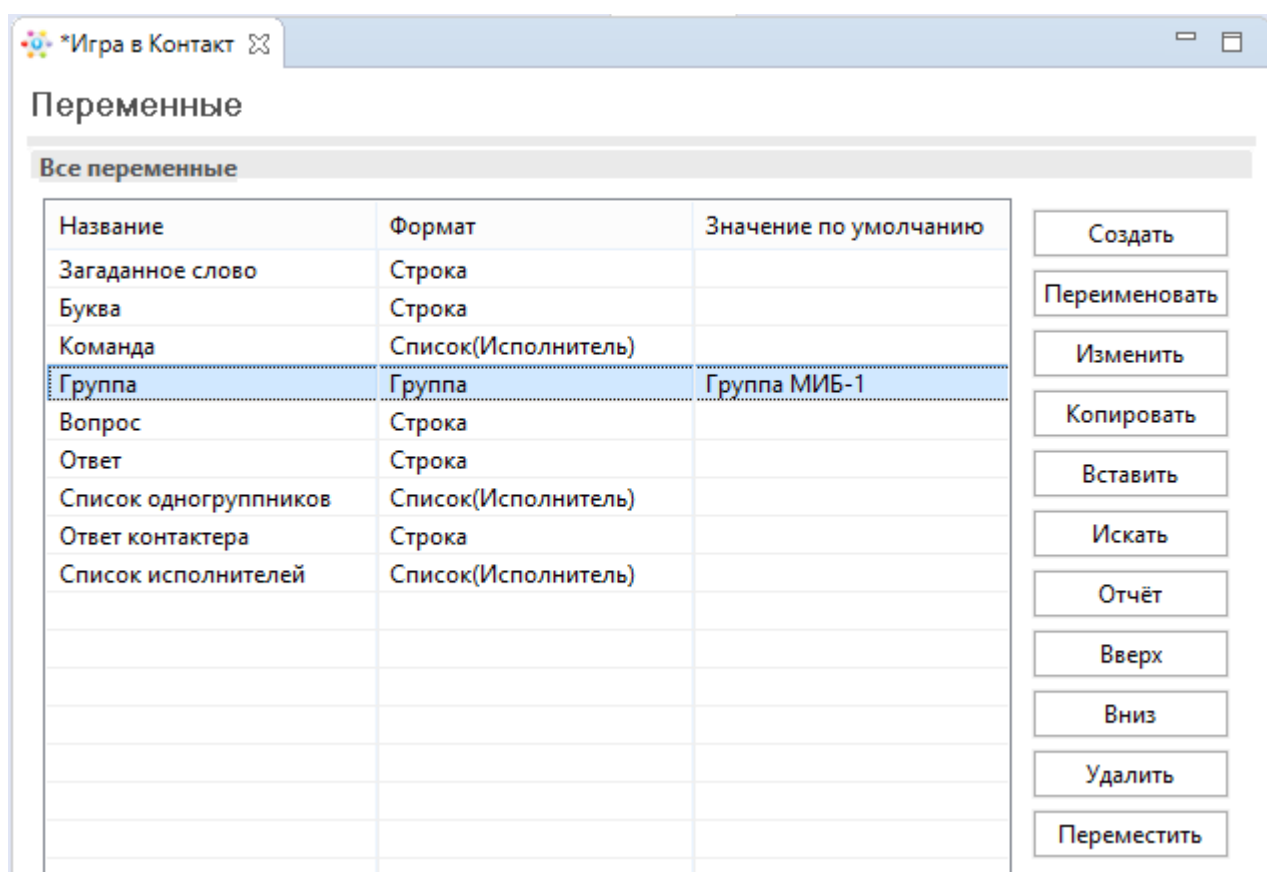
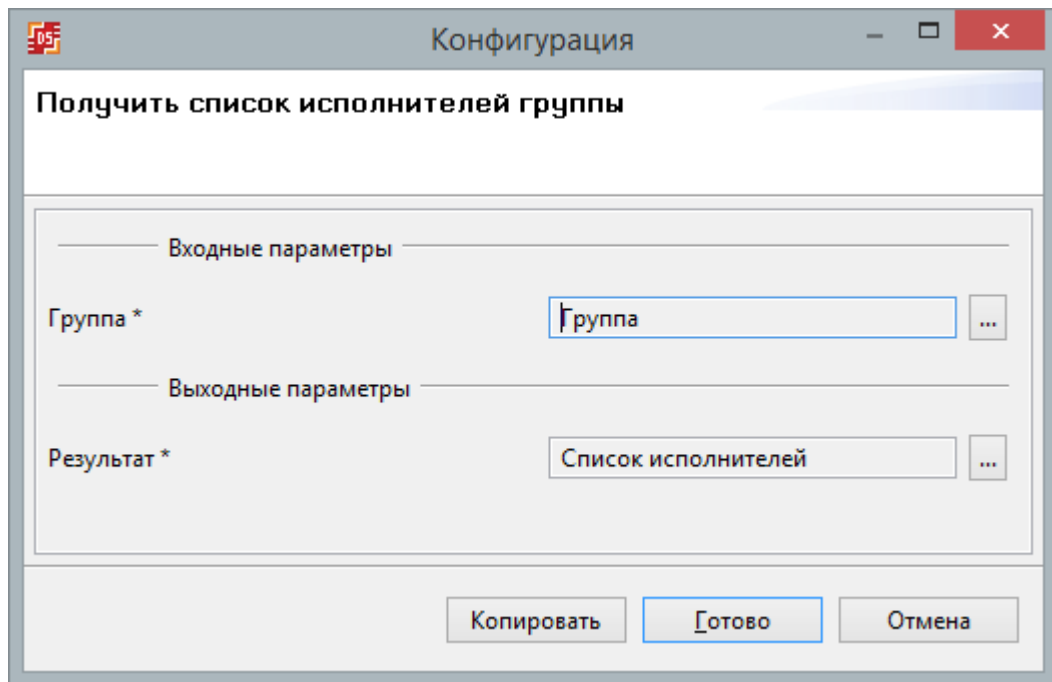


Рисунок 2.06. Переменная формата "Группа" со значением по умолчанию

Далее используем задачу сценария с обработчиком "Получить список исполнителей группы" со следующей конфигурацией



Конфигурация

Получить список исполнителей группы

Входные параметры

Группа *

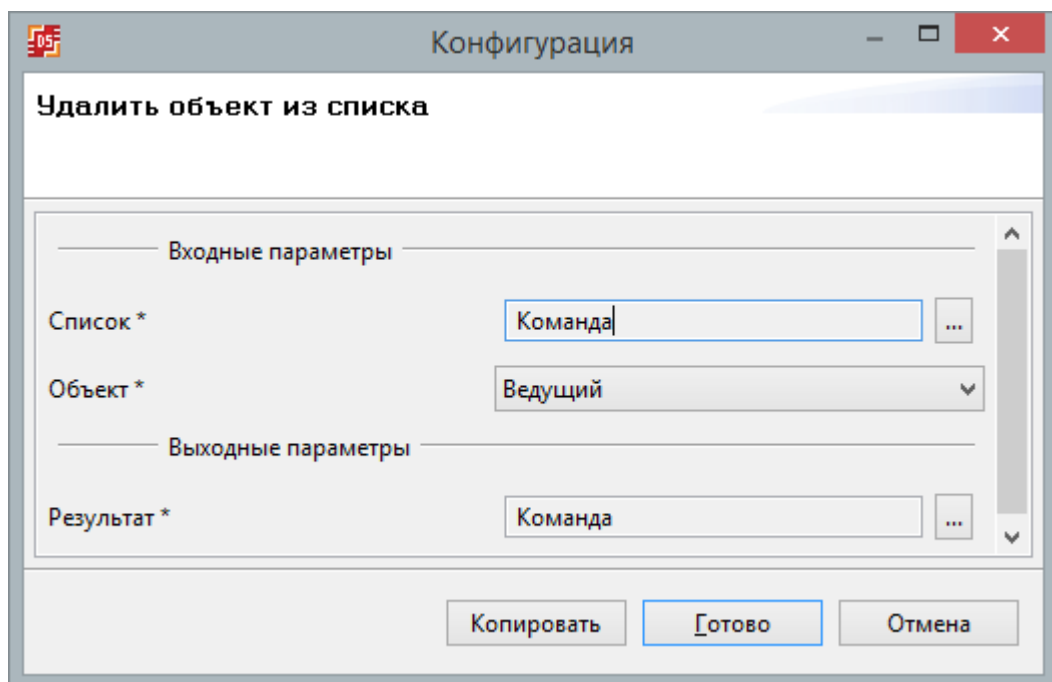
Выходные параметры

Результат *

Копировать Готово Отмена

Рисунок 2.07. Обработчик "Получить список исполнителей группы"

И исключаем "Ведущего" из полученного списка с помощью обработчика "Удалить объект из списка".



Конфигурация

Удалить объект из списка

Входные параметры

Список *

Объект *

Выходные параметры

Результат *

Копировать Готово Отмена

Рисунок 2.08. Обработчик "Удалить объект из списка"

Результат сохраняем в том же списке исполнителей - "Команда".

Для роли "Спрашивающий" используем инициализатор "Исполнитель из списка".

Рисунок 2.09. Инициализатор "Исполнитель из списка"

Посмотрим как это работает.

Группа МИБ-1:

Имя	Задачи сотрудников группы	Обладатели полномочий
Группа МИБ-1 *		
Описание		
Группа LDAP		

Применить

Группы исполнителя

Вид: Без фильтра ▼

Добавить

Имя /	Описание

Удалить

Члены группы

Вид: Без фильтра ▼

Добавить

Имя /	Полное Имя	Описание
☐	Гусеницын	Гусеницын Михаил Васильевич
☐	Личинкин	Личинкин Василий Федорович
☐	Мотыльков	Мотыльков Иван Петрович
☐	Мухин	Мухин Петр Иванович

Всего:4

Удалить

Рисунок 2.10. Список членов группы "Группа МИБ-1"

Запускает БП пользователь, входящий в группу, например, "Мухин". Далее в результате выполнения обработчика - формируется список "Команда", состоящая из членов группы.

Экземпляр процесса запущен пользователем Мухин
Создано задание Начало
Задание Начало назначено исполнителю Мухин (добавлены: null)
<u>Роль-дорожка Ведущий назначена исполнителю Мухин</u>
Переменная Буква проинициализирована значением а
Переменная Загаданное слово проинициализирована значением Информатика
Задание Начало выполнено пользователем Мухин
Управление ушло из Начало
Управление пошло по переходу Переход1
Управление пришло в получить список команды
Выполнено действие <code>ScriptNode{id=ID35, delegation=Delegation{class=ru.runa.wfe.extension.handler.ExecutorsListFromGroupHandler, configuration=<config> <input> <param name="group" variable="Группа"/> </input> <output> <param name="executors" variable="Команда"/> </output> </config>}}</code>
Переменная Команда.size проинициализирована значением 4
Переменная Команда[0] проинициализирована значением Мухин
Переменная Команда[1] проинициализирована значением Гусеницын
Переменная Команда[2] проинициализирована значением Мотыльков
Переменная Команда[3] проинициализирована значением Личинкин
Управление ушло из получить список команды

Рисунок 2.11. Формирование списка "Команда"

После выполнения обработчика "Удалить объект из списка" из списка "Команда" удаляется "Мухин", запустивший экземпляр процесса. И роль "Спрашивающий" (задача "Придумать вопрос") назначается на временную группу, уже без "Ведущего".

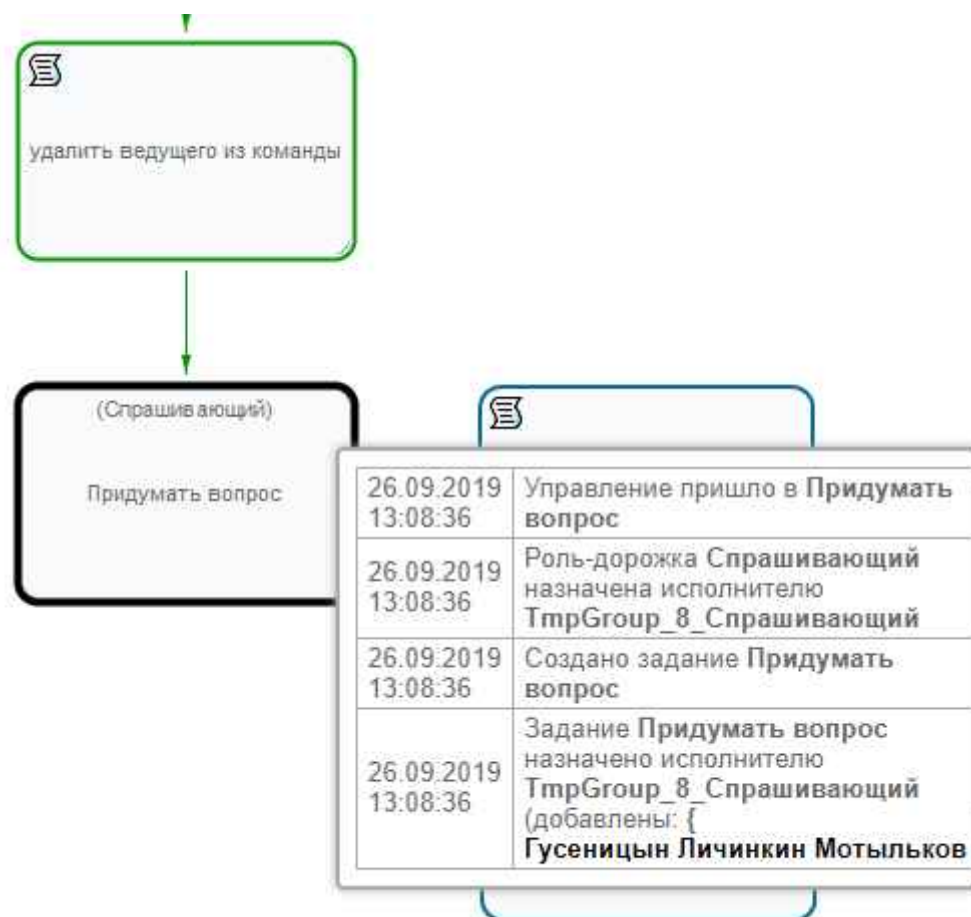


Рисунок 2.12. Назначение задания "Придумать вопрос"

Следующее задание "Установить контакт" должны получить все одnogруппники (определяются отношением «Одногруппники», примененным к выполнившему заданию "Придумать вопрос" под ролью "Спрашивающий", за исключением "Ведущего" и "Спрашивающего"). Для этого воспользуемся обработчиком "Получить список исполнителей по отношению", а также уже использованным ранее обработчиком удаления объекта из списка.

The dialog box is titled 'Конфигурация' and contains the following fields:

- Входные параметры (Input parameters):**
 - Название отношения *** (Relationship name): 'Одногруппники' (One-group members)
 - Параметр отношения *** (Relationship parameter): 'Спрашивающий' (Asking)
 - Использовать обратное отношение *** (Use reverse relationship): 'Нет' (No)
 - Включая группы *** (Including groups): 'Нет' (No)
- Выходные параметры (Output parameters):**
 - Результат *** (Result): 'Список одnogруппников' (List of one-group members)

Buttons at the bottom: 'Копировать' (Copy), 'Готово' (OK), 'Отмена' (Cancel).

Рисунок 2.13. Получение списка исполнителей по отношению "Одногруппники"

The dialog box is titled 'Конфигурация' and contains the following fields:

- Входные параметры (Input parameters):**
 - Список *** (List): 'Список одnogруппников' (List of one-group members)
 - Объект *** (Object): 'Ведущий' (Leader)
- Выходные параметры (Output parameters):**
 - Результат *** (Result): 'Список одnogруппников' (List of one-group members)

Buttons at the bottom: 'Копировать' (Copy), 'Готово' (OK), 'Отмена' (Cancel).

Рисунок 2.14. Исключение "Ведущего"

Для роли "Контактер" используем уже знакомый инициализатор "Исполнитель из списка"

Редактирование инициализатора роли

Функции над оргструктур... Группы и пользователи WF Кастомизированные иници... Код Бот

Тип: Исполнители из списка

ru.runa.wfe.extension.orgfunction.ExecutorsFromListFunction

Список исполнителей *: \${Список одnogруппников}

Задать роль с помощью отношения Задать роль с помощью оргфункции

☐ Доступ всем (не только обладателям полномочий) OK Отмена

Рисунок 2.15. Инициализатор для роли "Контактер"

Посмотрим как это работает.

Параметры отношения

Имя отношения: Одnogруппники *

Описание:

Сохранить

Одnogруппники

Вид: Без фильтра

Создать пару

Левая часть /	Правая часть /
☐ Гусеницын	☐ Личинкин
☐ Гусеницын	☐ Мотыльков
☐ Гусеницын	☐ Мухин
☐ Личинкин	☐ Гусеницын
☐ Личинкин	☐ Мотыльков
☐ Личинкин	☐ Мухин
☐ Мотыльков	☐ Гусеницын
☐ Мотыльков	☐ Личинкин
☐ Мотыльков	☐ Мухин
☐ Мухин	☐ Гусеницын
☐ Мухин	☐ Личинкин
☐ Мухин	☐ Мотыльков

Удалить

Рисунок 2.16. Отношение "Одnogруппники"

Пусть задачу "Придумать вопрос" выполнил "Личинкин", тогда применяя к нему отношение, получаем список одногруппников: "Гусеницын", "Мотыльков", "Мухин".

Выполнено действие ScriptNode{id=ID54, delegation=Delegation{class=ru.runa.wfe.extension.handler.user.GetExecutorsByRelationHandler, configuration=<config> <input> <param name="name" value="Одногруппники"/> <param name="parameter" variable="Спрашивающий"/> <param name="inversed" value="false"/> <param name="recursively" value="false"/> </input> <output> <param name="result" variable="Список одногруппников"/> </output> </config>}}	
Переменная Список одногруппников.size	проинициализирована значением 3
Переменная Список одногруппников[0]	проинициализирована значением Гусеницын
Переменная Список одногруппников[1]	проинициализирована значением Мухин
Переменная Список одногруппников[2]	проинициализирована значением Мотыльков
Управление ушло из Получение одногруппников	
Управление пошло по переходу Переход1	
Управление пришло в удалить ведущего из одногруппников	

Рисунок 2.17. Получение списка одногруппников "Личинкина"

Из полученного списка удаляется "Мухин" и роль "Контактер" назначается на временную группу без "Ведущего".

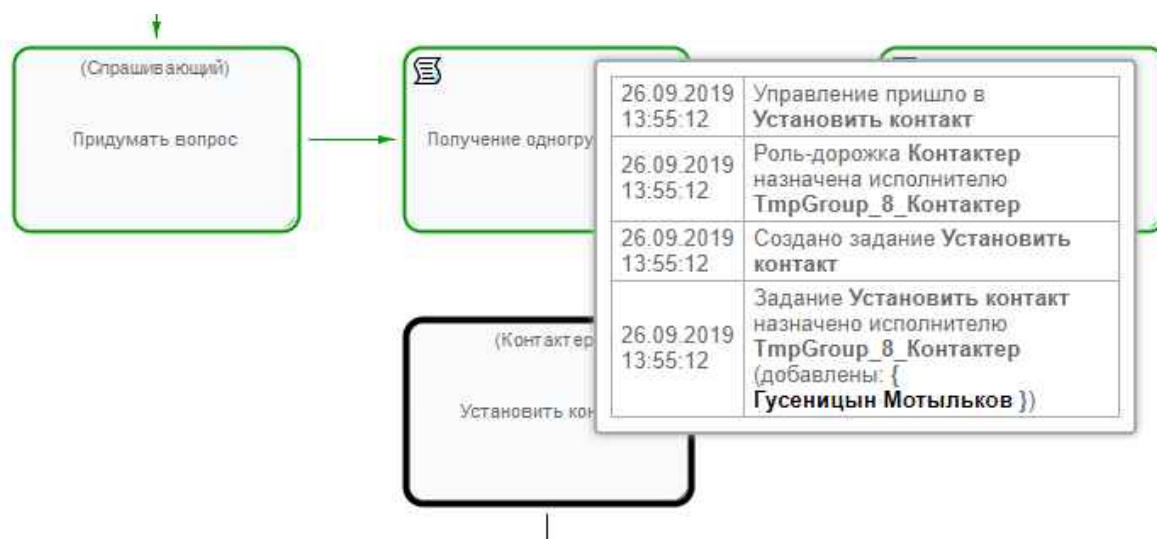


Рисунок 2.18. Назначение задачи "Установить контакт"

3. Полезные обработчики

3.1. Получение номера текущего экземпляра процесса

Функция `get_process_id()` обработчика "Выполнить формулу" позволяет получить некоторый уникальный идентификатор, являющийся номером текущего запущенного экземпляра БП.

Полезность использования данной функции можно посмотреть на примере БП из практикума по внешнему хранилищу данных (*Практическое занятие "Разработка бизнес-процесса "Заказ автотранспорта"*).

Сотрудники предприятия подают заявки на заказ транспорта из автопарка (БП "Заявка"). Одобренные руководителем заявки сохраняются во внешнем хранилище для дальнейшей обработки в другом БП ("Планирование"), в рамках которого диспетчер

выполняет отбор заявок и рассмотрение возможности выполнения. Сотрудник должен получить информацию о статусе обработки и результате рассмотрения его заявки. Для этого каждая заявка должна иметь свой уникальный номер, связанный с экземпляром БП, из которого она была подана.

Для получения этого номера используется элемент "Задача сценария" с обработчиком "Выполнить формулу".

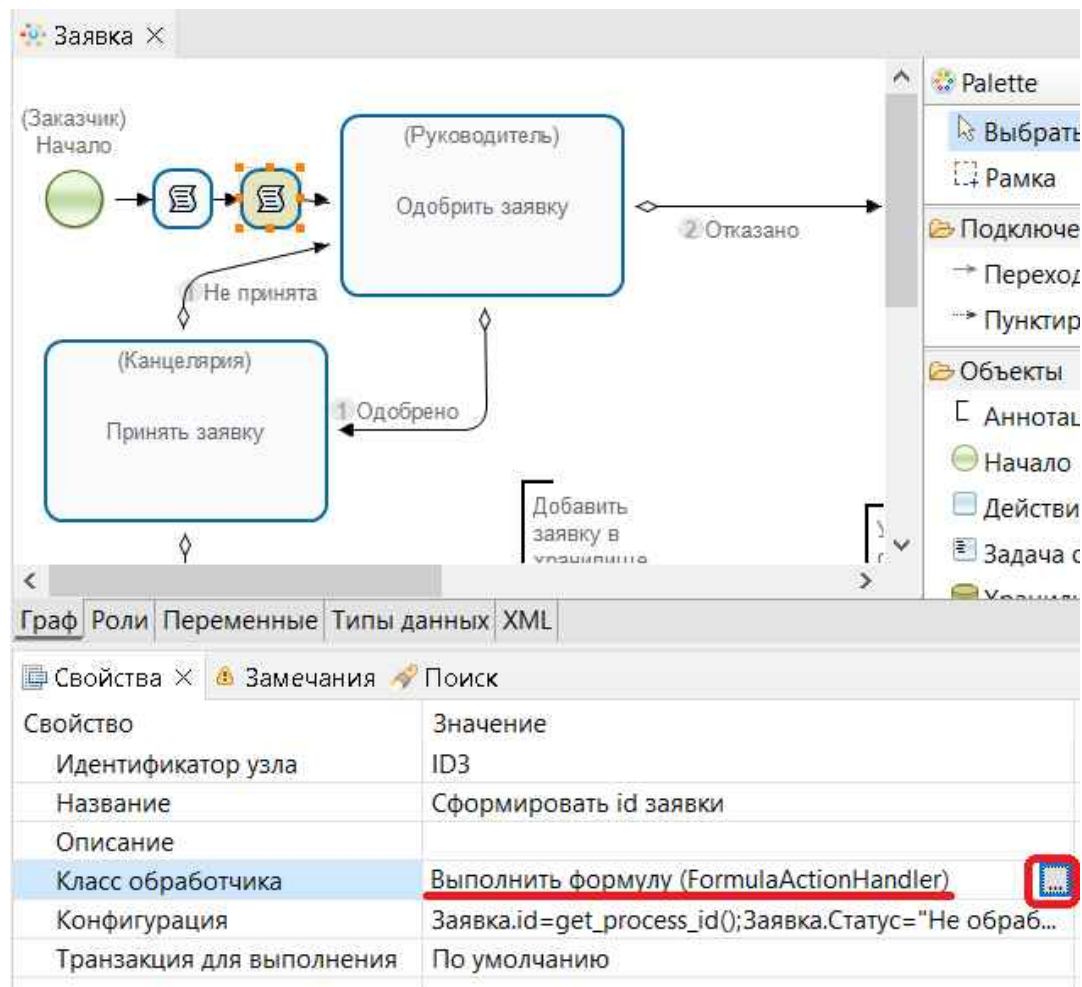


Рисунок 3.01. Выбор класса обработчика для задачи "Сформировать id заявки"

В конфигурации обработчика используется функция *get_process_id()*.

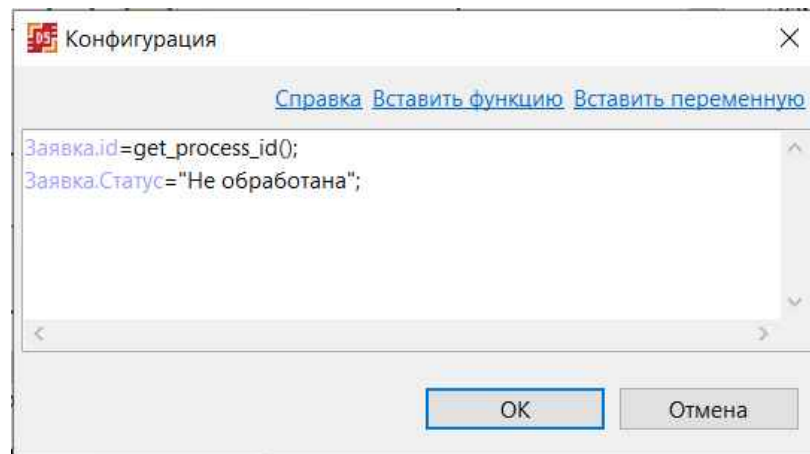


Рисунок 3.02. Конфигурация задачи "Сформировать id заявки"

3.2. Получение текущей даты-времени

Для возвращения текущих даты, времени, даты-времени используются соответствующие функции обработчика "Выполнить формулу"

- `current_date()`
- `current_time()`
- `current_date_time()`.

Пример возможного применения данной функции можно найти в практическом занятии "Работа с MS Word и MS Excel документами", где полученная текущая дата используется в служебной записке о начислении премии.

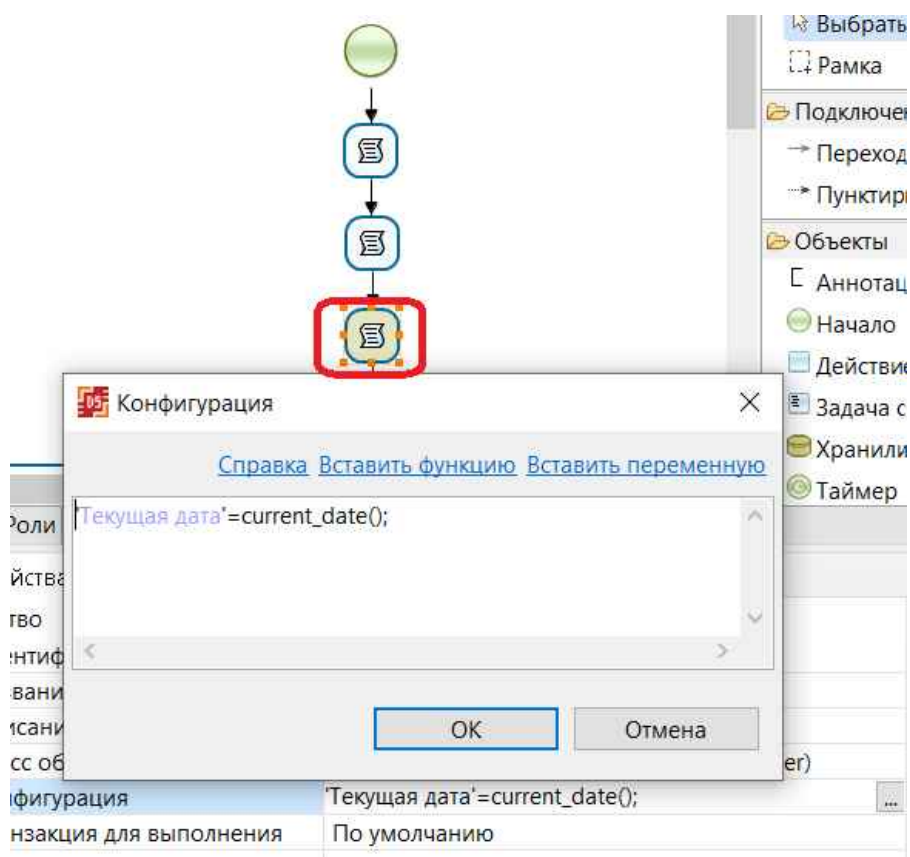


Рисунок 3.03. Определение текущей даты

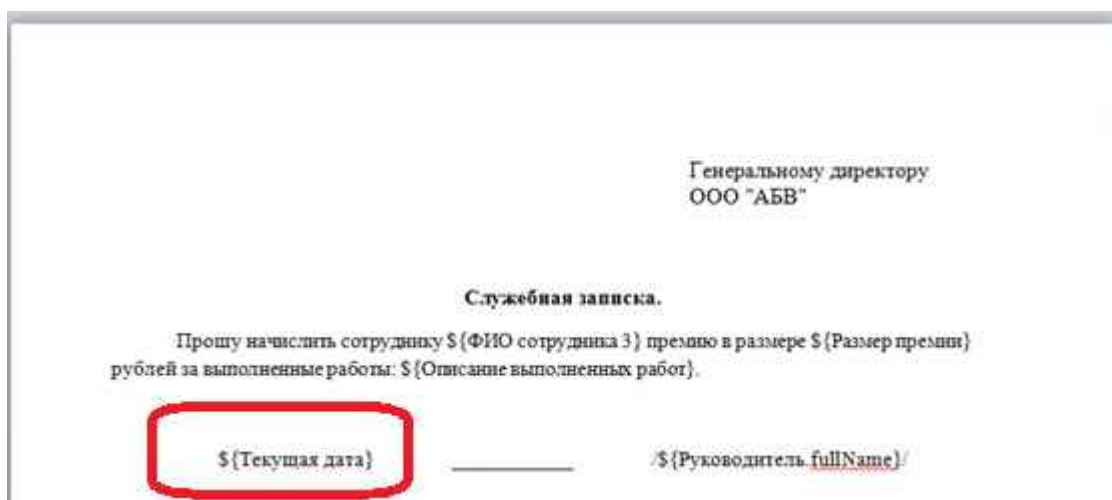


Рисунок 3.04. Использование текущей даты в шаблоне DOCX-документа

3.3. Получение ФИО пользователя и использование падежей

Часто в БП необходимо получить информацию об исполнителе из его профиля, например его ФИО. В практическом занятии "Работа с MS Word и MS Excel документами" ФИО исполнителей (сотрудника, руководителя) используется в формируемой с помощью обработчика DOCX служебной записке о начислении премии. При этом ФИО сотрудника в документе должно быть представлено в дательном падеже.

Для получения ФИО пользователя используется обработчик "Получить информацию об исполнителе".

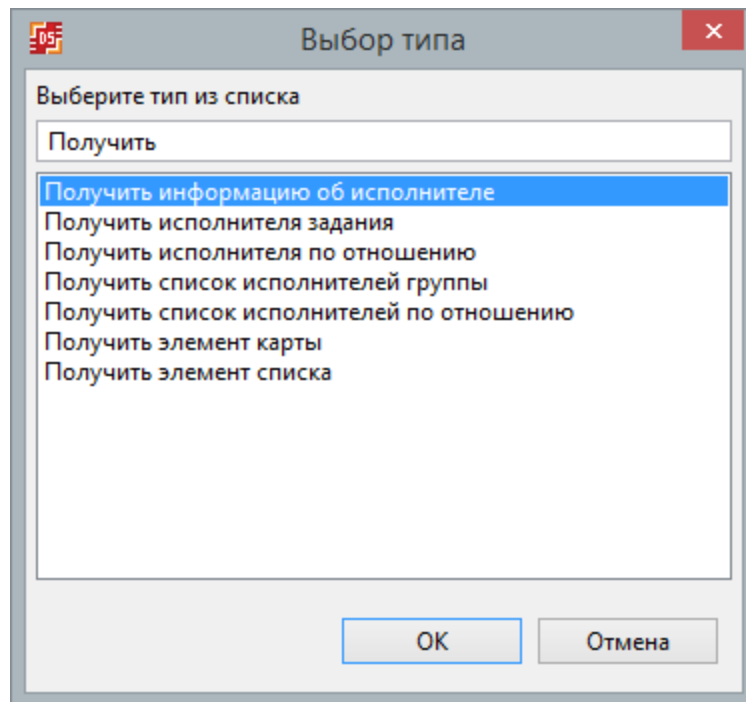


Рисунок 3.05. Обработчик "Получение информации об исполнителе"

Во входном параметре "Исполнитель" доступны для выбора переменные формата "Исполнитель", "Пользователь", а также Роль.

В параметре "Формат" необходимо выбрать информацию, которую требуется получить

- active - статус активности
- code - код
- department - подразделение
- description - описание
- email
- fullName - полное имя пользователя (ФИО)
- id
- name - имя
- phone - телефон.

Результат сохраняется в переменную формата "Строка".

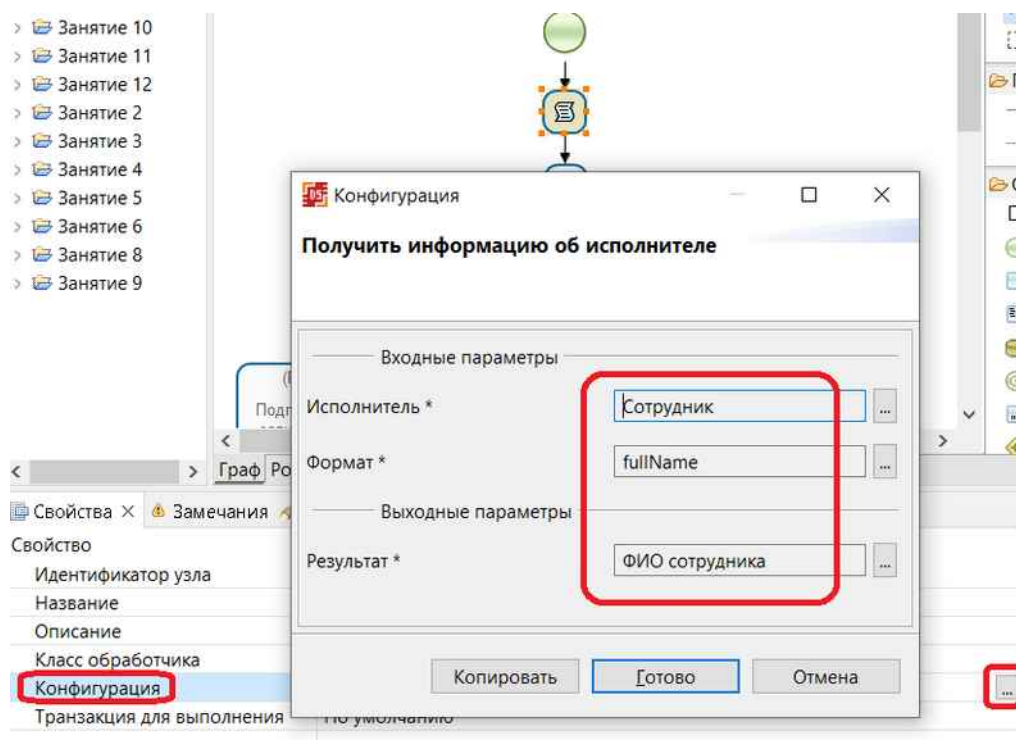


Рисунок 3.06. Конфигурация обработчика для получения ФИО

Для склонения ФИО используется обработчик "Выполнить формулу" с функцией *FIO_case_ru(fio, caseNumber, mode)*.

Параметры функции:

- *fio* - фамилия, имя и отчество в именительном падеже (переменная формата "Строка")
- *caseNumber* - номер падежа от 1 до 6
- *mode* - строка форматирования:

Символы *F*, *I* и *O* заменяются на фамилию, имя или отчество в соответствующем падеже.

Символы *f*, *i* и *o* заменяются на первую букву фамилии, имени или отчества.

Например, при следующих параметрах

```
'ФИО сотрудника 3'=FIO_case_ru('ФИО сотрудника', 3, "F i o")
```

функция вернет ФИО сотрудника в дательном падеже, первую букву Имени и первую букву Отчества, например Иванову И.И. в переменную "ФИО сотрудника 3".

Еще примеры:

<code>FIO_case_ru("Иванов Иван Иванович", 1, "F N P :)")</code>	Иванов Иван Иванович :)
<code>FIO_case_ru("Иванова Ира Ивановна", 2, "F N p")</code>	Ивановой Иры И.
<code>FIO_case_ru("Иванов Иван Иванович", 3, "F n P")</code>	Иванову И. Ивановичу
<code>FIO_case_ru("Иванова Ира Ивановна", 4, "F n p")</code>	Иванову И. И.
<code>FIO_case_ru("Иванов Иван Иванович", 5, "f N P")</code>	И. Иваным
<code>FIO_case_ru("Иванова Ира Ивановна", 6, "o P n")</code>	о Ивановне И.

3.4. Использование формулы-обработчика для получения случайного числа (датчик или генератор случайных чисел)

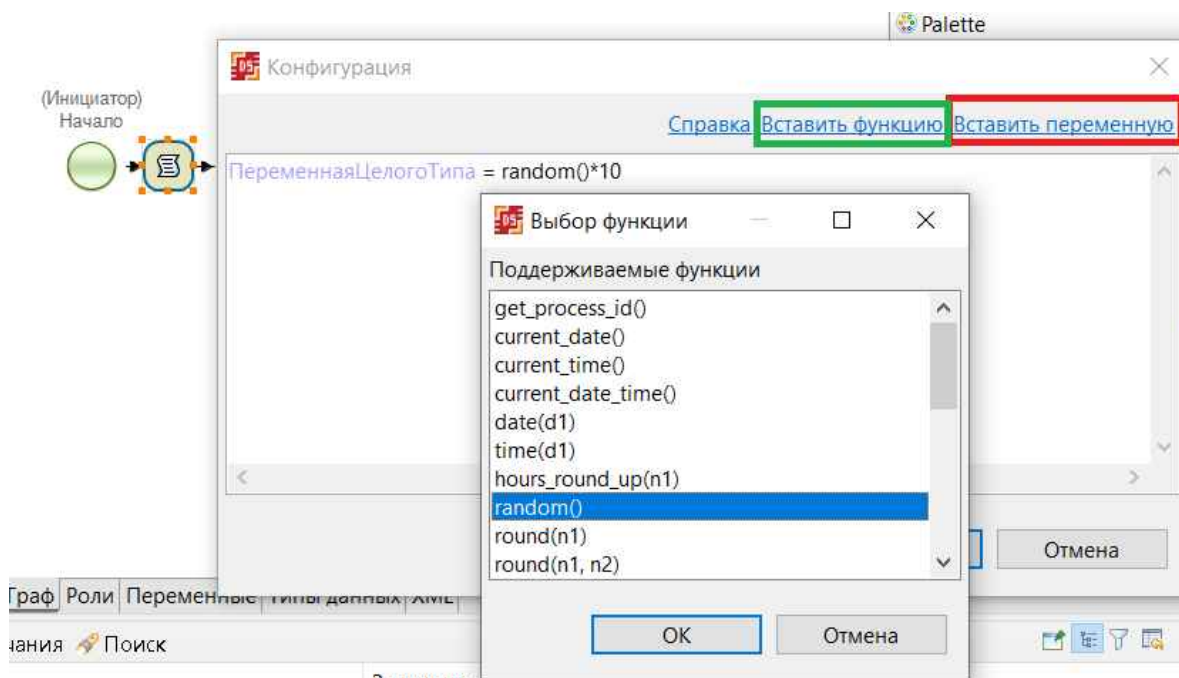
random() - возвращает псевдослучайное число с плавающей запятой из диапазона [0, 1), то есть, от 0 (включительно) до 1 (но не включая 1), которое затем можно отмасштабировать до нужного диапазона.

- Поместить Задачу сценария на схему
- Для удобства и компактности в опциях Задачи сценария (кликнув по ПКМ) выбрать Компактный вид
- Выбрать класс обработчика - Выполнить формулу
- По клику на Задаче сценария заполнить конфигурацию

Выбрать переменную, которой надо приравнять сгенерированное случайное число, кликнув Выбрать переменную (красная обводка) - в выпадающем меню появятся созданные в процессе переменные.

Генерация случайного числа происходит с помощью функции random(). Написать её можно вручную или выбрать в конфигурации, кликнув Выбрать функцию (зелёная обводка) - random().

Например, для получения целого числа в диапазоне от 0 до 9 нужно написать в конфигурации обработчика `ПеременнаяЦелогоТипа = random()*10`, где `ПеременнаяЦелогоТипа` - созданная в процессе переменная целого типа.



Выполнить формулу - random()

3.5. Использование формулы-обработчика для создания выводимых на форму задания текстов

В обработчике "Выполнить формулу" можно формировать текст выводимый на формы, используя конкатенацию переменных.

В следующем примере сформируем текст служебной записки о начислении премии сотруднику.

Пусть в строковой переменной "ФИО сотрудника 3" содержится ФИО в дательном падеже, а в переменной формата "Число повышенной точности" - "Размер премии" содержится размер премии. Текст сохраним в переменной "Текст" (формата "Текст" или "Строка"):

Текст = "Прошу начислить сотруднику " + 'ФИО сотрудника 3' + " премию в размере " + 'Размер премии' + " рублей"

Полученную переменную выведем на форму с помощью компонента "Отобразить переменную" (форма отображения - "как строку").

The screenshot shows the RUNA WFE Workflow System interface. At the top left is the logo and name 'RUNA WFE WORKFLOW SYSTEM'. At the top right, it says 'Вы вошли как Administrator' and 'Выход'. On the left is a menu with items like 'Список заданий', 'Запустить процесс', etc. The main area is titled 'Форма задания' and contains a table with columns: 'Имя / Описание', 'Имя корневого процесса', 'Номер экземпляра корневого процесса', 'Владелец', 'Роль', 'Время окончания', and 'Создана'. The table has one row with data: 'Действие 1', 'процесс', '103', 'Administrator', 'Инициатор', '26.08.2026 23:01', and '26.08.2026 21:01'. Below the table, there is a section 'Форма задания' with the text 'Прошу начислить сотруднику Иванову И. И. премию в размере 5000.00 рублей' and a button 'Задание исполнено'.

Рисунок 3.07. Вывод текста о начислении премии, сформированного в обработчике "Выполнить формулу"

4. Режим последовательного показа форм заданий

По умолчанию после выполнении очередного задания интерфейс автоматически перебрасывает пользователя на страницу со списком его задач. Но есть возможность изменить этот режим на так называемый режим последовательного показа форм, при котором выполняется автоматическое отображение формы следующей задачи.

Для переключения режима потребуется войти в настройки веб-интерфейса (пункт "Настройки" доступен пользователям из группы Administrators), перевести параметр *task.form.autoShowNext* в значение *true* и нажать "Сохранить" внизу таблицы с параметрами. Такой способ настройки позволяет применить новое значение параметра сразу, без перезагрузки сервера.

Меню	Настройки веб-интерфейса
Список заданий	Параметр
Запустить процесс	Описание
Запущенные процессы	Значение
Архивные процессы	group.subprocess.enabled
Исполнители	Доступна ли группировка по подпроцессам в фильтре?
Отчёты	task.form.autoShowNext
Отношения	Автоматически показывать форму следующего задания (в том же токене)
Бот станции	task.form.highlightRequiredFields
Источники данных	Обозначать обязательные поля на форме задания
Внутреннее хранилище	task.form.ajaxFileInputEnabled
Система	Использовать в качестве элемента загрузки файла (обеспечивает сохранение загруженного файла при неуспешной проверке данных и возможность очистки файла) не стандартный элемент ввода (используется jquery file upload)
Ошибки	process.variables.displayJavaType
Зависшие процессы	Отображать java-тип переменной в списке переменных процесса
Настройки	view.logs.limit.lines.count
Логи сервера	Максимальное количество отображаемых строк лога
Задания сотрудников	view.logs.limit.line.characters.count
Отправить сигнал	view.logs.timeout.autoreload.seconds
	Таймаут перезагрузки лога на странице просмотра
	15

Рисунок 4.01. Включение режима последовательного отображения форм заданий

Рассмотрим пример.

БП с элементами *Начало* -> *Задание 1* -> *Задание 2* -> *Окончание*, стартовый узел не имеет формы.

Пользователь запускает на выполнение БП, вверху страницы отображается сообщение о запуске нового экземпляра и автоматически открывается форма следующей задачи "Задание 1".

Меню	Форма задания
Список заданий	☐ Имя / Описание Имя корневого процесса Номер экземпляра корневого процесса Владелец Роль Время окончания Создана
Запустить процесс	☒ Задание 1 Последовательное отображение 105 Administrator Инициатор 26.08.2026 23:08 26.08.2026 21:08
Запущенные процессы	Делегировать задачу
Архивные процессы	
Исполнители	
Отчёты	
Отношения	
Бот станции	
Источники данных	
Внутреннее хранилище	
Система	
Ошибки	
Зависшие процессы	

Рисунок 4.02. Автоматическое отображение формы следующей задачи

После выполнения задачи "Задание 1" будет сразу отображена форма следующей задачи "Задание 2".

Меню	Форма задания
Список заданий	☐ Имя / Описание Имя корневого процесса Номер экземпляра корневого процесса Владелец Роль Время окончания Создана
Запустить процесс	☒ Задание 2 Последовательное отображение 105 Administrator Инициатор 26.08.2026 23:10 26.08.2026 21:10
Запущенные процессы	Делегировать задачу
Архивные процессы	
Исполнители	
Отчёты	
Отношения	
Бот станции	
Источники данных	
Внутреннее хранилище	
Система	
Ошибки	
Зависшие процессы	

Рисунок 4.03. Автоматическое отображение формы следующей задачи

Замечание. Если между узлами действиями будет расположен один из следующих элементов - "Таймер", "Обработка события", "Подпроцесс"(но не "Композиция"), "Мультиподпроцесс", "Параллельный шлюз", "Задача бота" - то режим последовательного отображения не сработает и для пользователя будет открыта страница "Список заданий".

Также стоит обратить внимание на еще одну особенность. По умолчанию в RunaWFE Free используется режим **коротких транзакций** ([описание работы транзакций при исполнении процессов на сервере](#)) при котором каждый элемент работает в новой транзакции. В этом случае расположенный между двумя узлами действия элемент: "Исключающий иллюз", "Задача сценария", "Мультидействие", "Генерация события" также приведет к тому, что следующая форма не будет отображена автоматически. Но в случае использования **длинных транзакций** режим последовательного отображения форм будет работать.

Переключиться на использование длинных транзакций можно как в отдельно взятом БП, отдельном элементе БП, так и в целом во всех узлах всех бизнес процессов ([см. Настройки для изменения границ транзакций](#)).

Для переключения границ транзакций на уровне процесса необходимо

- открыть БП в DevStudio,
- перейти в свойства БП,
- изменить "Транзакция для выполнения по умолчанию".

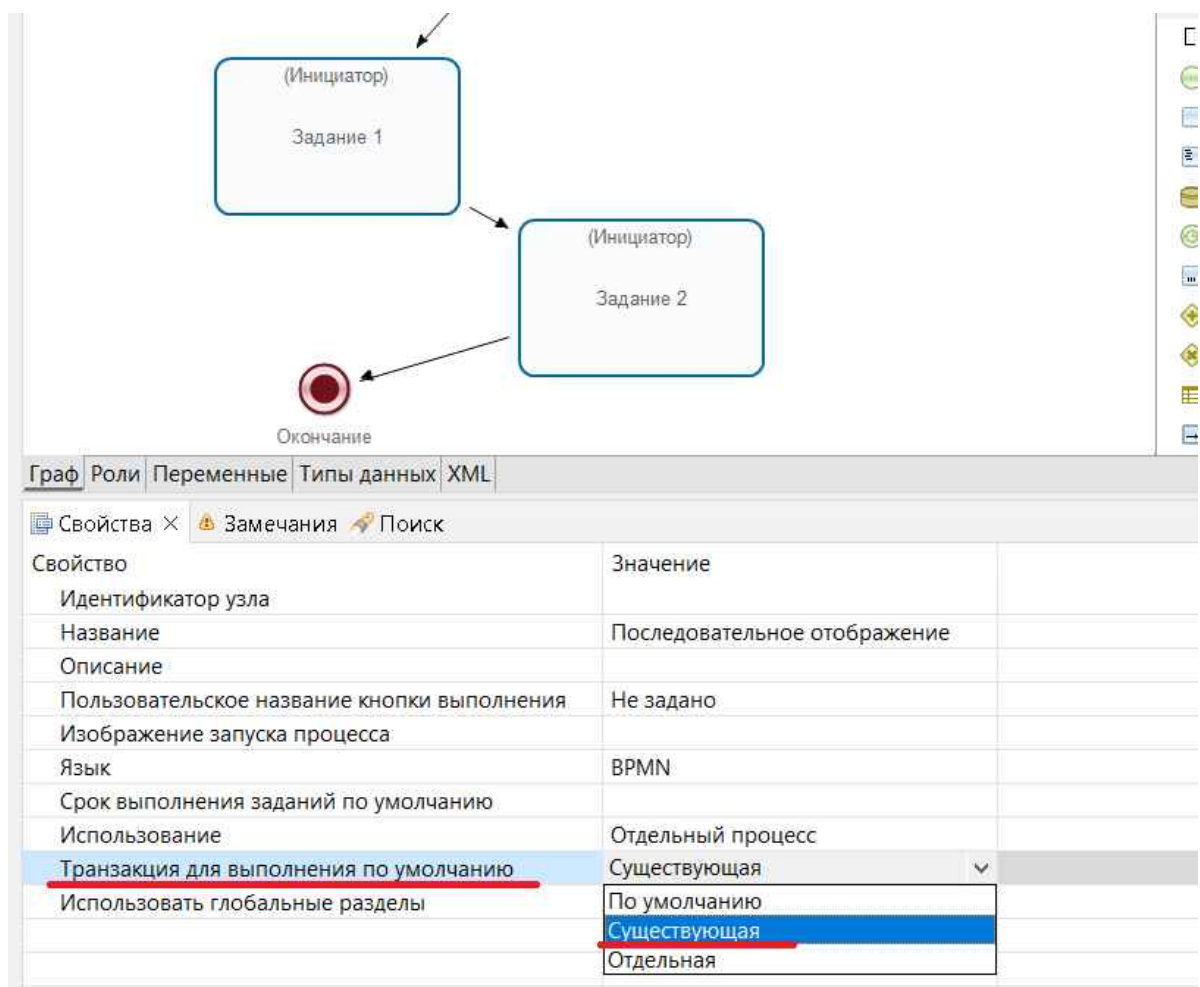


Рисунок 4.04. Изменение границ транзакций на уровне БП

В случае выбора "Существующая" будет использоваться режим длинных транзакций, в случае "Отдельная" - коротких. Такое же свойство имеет и каждый элемент графа.

Изменить режим транзакций на уровне сервера можно в веб интерфейсе - пункт меню "Настройки", раздел "Основные настройки", где перевести параметр *process.execution.node.async.default* в *false* (режим длинных транзакций) или *true* (режим коротких, по умолчанию).

Меню

Список заданий
Запустить процесс
Запущенные процессы
Архивные процессы
Исполнители
Отчеты
Отношения
Бот станции
Источники данных
Внутреннее хранилище
Система
Ошибки
Зависшие процессы
Настройки
Логи сервера
Задания сотрудников
Отправить сигнал

Основные настройки

Параметр	Описание	Значение
strong_passwords.regexp	Для усиления защиты системы можно наложить ограничения на пароли пользователей системы. Если определен этот параметр, то все заводимые пароли пользователей должны соответствовать регулярному выражению. Если пароль не отвечает требованиям безопасности, то смена пароля произведена не будет.	
escalation.enabled	Эскалация заданий включена?	true
escalation.default.hierarchy.loader	Орг. функция или отношение, для определения иерархии эскалации заданий	ru.runa.wfe.extension.orgfunction.TestOrgFuncio
task.default.deadline	Время исполнения задания по умолчанию (если не установлено явно в Среде разработки)	2 hours
task.almostDeadlinePercents	Процент истечения времени исполнения после которого цвет задания в списке меняется на более приоритетный	90
task.delegation.enabled	Включить делегирование задач	true
token.maximum.depth	Максимально разрешенная вложенность токенов, для предотвращения логических ошибок	100
token.maximum.length	Максимальное количество переходов токенов, для предотвращения логических ошибок (-1 для отключения проверки)	-1
file.variable.local.storage.enabled	Разрешить использование локального хранилища в файловой системе для больших файлов	true
file.variable.local.storage.enableforfilesgreaterthan	Если размер файловой переменной превышает установленный - то файл сохраняется в файловой системе, а не в БД	100000
date.format.pattern	Формат даты в системе (также используется при форматировании даты со временем)	dd.MM.yyyy
scriptingServiceAPI.executeGroovyScript.enabled	Разрешить выполнение метода ru.runa.wfe.service.ScriptingService.executeGroovyScript(User, String	false
undefined.variables.allowed	Разрешить создание в процессе переменных, не определенных в Среде разработки?	false
strong.variables.format.enabled	Режим строгой типизации. Требовать соответствие типа определенному в Среде разработки при установке значений переменных	true
variables.autocast.enabled	Выполнять приведение типа к определенному в Среде разработки в случае несоответствия типа значения переменной в режиме строгой типизации	true
trusted.authentication.enabled	Разрешена ли доверенная аутентификация с помощью сервисного аккаунта (AuthenticationService.authenticateByTrustedPrincipal(User, String))	false
auto.invocation.local.botstation.enabled		
definition.comments.collisions.allowed	Разрешить загрузку определения процесса, в котором отсутствуют некоторые комментарии из прежней версии определения	false
definition.comments.empty.allowed	Разрешить загрузку определения процесса, в котором нет новых комментариев	true
variables.invalid.default.values.allowed		true
variables.invalid.default.values.allowed.before		01.01.1970
datasource.password.export	Разрешить выгрузить/загрузить пароли источников данных	true
process.execution.node.async.default	По умолчанию использовать режим коротких транзакций для выполнения узлов	false
preferred.messages.language	Предпочтительный язык сообщений	
statistic.report.enabled		true
processlog.cleanButton.enabled	Включить кнопку очистки логов процессов	false
definition.update.delete.tokens.for.missing.nodes	Удалить точки управления при обновлении определения БП для несуществующих узлов	false
definition.compatibility.check.enabled	Проверять совместимость определения БП при обновлении версии	true
reassign.swimlane.to.initializer	Определить значение по умолчанию для настройки Переименование роли	false
reassign.swimlane.to.task.performer	Определить значение по умолчанию для настройки Доинициализация роли	true

СохранитьОтменить

Рисунок 4.05. Изменение границ транзакций на уровне сервера

5. Отображение переходов по имени

По умолчанию кнопка выполнения в случае единственного перехода из действия отображается как “Задание исполнено”, независимо от имени перехода, указанном в DevStudio. В RunaWFE есть возможность переключить в режим отображения пользовательского названия. Для этого необходимо щелкнуть в любом свободном месте процесса и в свойствах поменять в соответствующем поле режим с “Не задано” на “По названию перехода”.

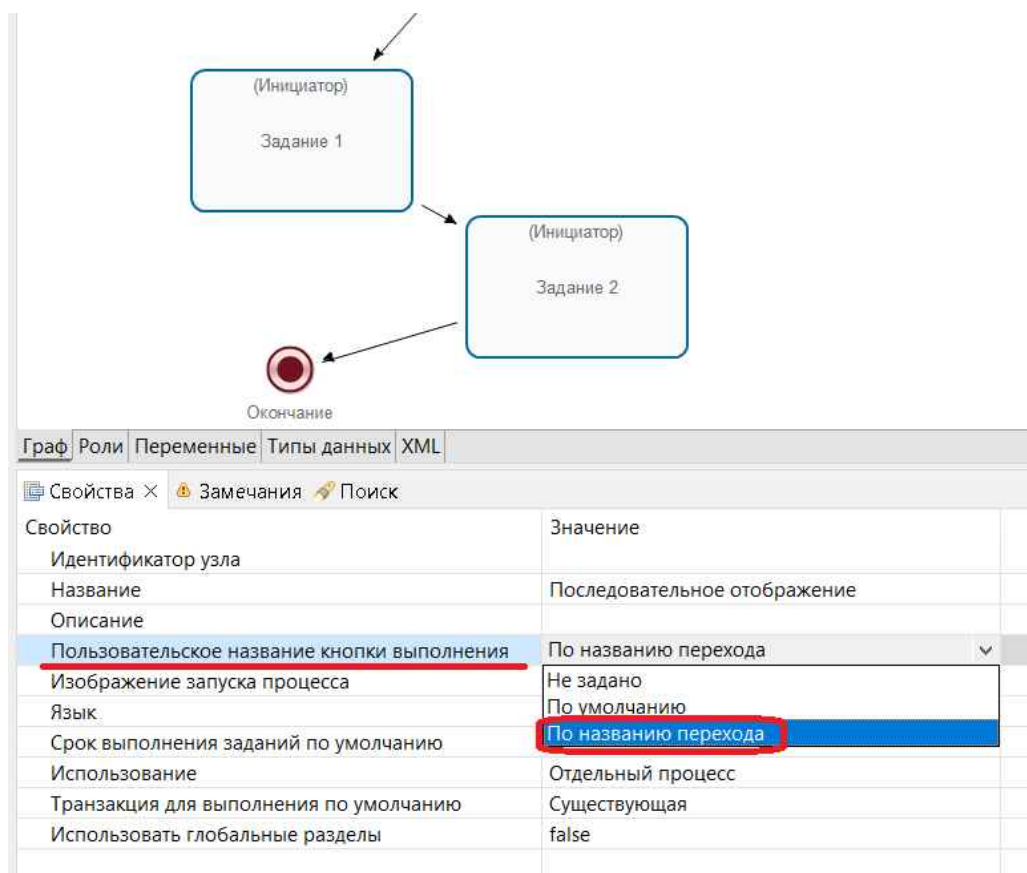


Рисунок 5.01. Включение отображения кнопки выполнения по названию перехода

6. Использование замещений

Замещение пользователей применяется в случаях, когда исполнитель, которому предназначено задание, не имеет возможности его выполнить, например, находится в отпуске. В таких случаях подсистема замещения исполнителей перенаправляет задание другому пользователю.

В системе RunaWFE Free правила замещения "привязаны" к исполнителям задач. Для каждого исполнителя в системе есть возможность снять статус "Активен", например по причине отъезда в командировку или болезни.

Свойства исполнителя

Задачи сотрудника

Обладатели полномочий

Имя	Иванов
Полное имя	Иванов Иван Иванович
Описание	
Код	-29
Адрес электронной почты	☒ Отправлять уведомления о задачах ☒ Отправлять уведомления о сообщениях в чате
Телефон	
Должность	
Подразделение	

Применить

Свойства цифровой подписи

Создать

Статус

Активен	☐
---------	--------------------------

Применить

Пароль

Новый пароль	
Повторите пароль	

Применить

Рисунок 6.01. Статус "Активен" в свойствах пользователя

При снятии данной опции задания исполнителя будут перенаправлены другому пользователю согласно правилам замещения, описанным в секции "Заместители" в свойствах неактивного исполнителя. Для каждого замещаемого сотрудника правила замещения просматриваются в некотором порядке до тех пор, пока либо не будет найдено подходящее правило замещения, либо будет выяснено, что ни одного подходящего правила нет ([подробнее о принципах работы механизма и о замещениях в целом](#)).

Рассмотрим пример настройки замещений, в котором все задания пользователя "Иванов" (в случае его отсутствия - сброшен статус "Активен") будут перенаправлены Администратору.

Текущий список заданий пользователя "Иванов":

Вы не активны в системе, правила замещения не настроены

Вы вошли как *Иванов*
Выход

Меню

Список заданий
Запустить процесс
Запущенные процессы
Архивные процессы
Исполнители
Отчёты
Логи сервера
Задания сотрудников

Версия 4.6.2

Сборка master, e563649-dirty, 25.07.2026

Задачи

Вид: Без фильтра

Справка

Всего:3

☐	Имя	Описание	Имя корневого процесса	Номер экземпляра корневого процесса	Владелец	Роль	Время окончания	Создана
☒	Просмотреть данные считанные из Excel документа		Пример 6-3	107	Иванов	Студент	26.08.2026 23:27	26.08.2026 21:27
☒	Действие 2		Пример 1-1	106	Иванов	Роль1	26.08.2026 23:25	26.08.2026 21:25
☒	Ввести сообщение для отправки		Пример 8-1	108	Иванов	Сотрудник	26.08.2026 23:24	26.08.2026 21:24

Всего:3

Рисунок 6.02. Текущий список заданий пользователя "Иванов"

Добавим правило замещения, для этого необходимо:

- войти в веб интерфейс RunaWFE Free под "Administrator"
- открыть свойства исполнителя "Иванов"
- перейти к секции "Заместители"
- открыть ссылку "Добавить правило":

Группы исполнителя

► Вид Без фильтра ▼

Добавить

☐	Имя /	Описание
☐	Верстальщик	
☐	Гараж	
☐	Сотрудники	

Всего:3

Всего:3

[Удалить](#)

Отношения, в которые исполнитель входит в правой части

Имя отношения	Описание
Руководитель организации	

Отношения, в которые исполнитель входит в левой части

Имя отношения	Описание
Заместители	

[Добавить правило](#) [Добавить терминатор](#)

☐	Оргфункция	Критерий	Применять
☐			

[Удалить](#)

Рисунок 6.03. Секция "Заместители" в свойствах пользователя

На странице добавления нового правила определяем следующие параметры:

Критерий замещения

Критерий	замещать всегда ▼
Применять	☒
Оргфункция	Исполнитель по имени ▼ *
Имя исполнителя	Administrator * Выбрать

[Добавить](#)

Рисунок 6.04. Добавление нового правила замещения исполнителя

- Критерий - "Замещать всегда"
- Применять - "Да"
- Оргфункция - "Исполнитель по имени".

В качестве имени пользователя - выбрать из списка пользователей - "Administrator".
Щёлкнуть "Добавить".

Новое правило будет в списке правил замещения пользователя "Иванов":

Заместители

Добавить правило
Добавить терминатор

☐	Оргфункция	Критерий	Применять
☐	Исполнитель по имени(Administrator)	замещать всегда	☒

Удалить

Рисунок 6.05. Секция с правилами замещения исполнителя

Уберем статус "Активен" у пользователя "Иванов", перейдем в список заданий (пользователя Administrator) и увидим следующее:

Задачи

Вид

Без фильтра

Справка

Всего:5

☐	Имя /	Описание	Имя корневого процесса	Номер экземпляра корневого процесса	Владелец	Роль	Время окончания	Создана
☒	Действие 1		Последовательное отображение	110	Administrator	Инициатор	26.08.2026 23:31	26.08.2026 21:31
☒	Действие 1		Последовательное отображение	109	Administrator	Инициатор	26.08.2026 23:31	26.08.2026 21:31
☒	Просмотреть данные считанные из Excel документа		Пример 6-3	107	Иванов	Студент	26.08.2026 23:27	26.08.2026 21:27
☒	Действие 2		Пример 1-1	106	Иванов	Роль1	26.08.2026 23:25	26.08.2026 21:25
☒	Ввести сообщение для отправки		Пример 8-1	108	Иванов	Сотрудник	26.08.2026 23:24	26.08.2026 21:24

Всего:5

Рисунок 6.06. Список заданий пользователя, полученных по замещению

Как можно наблюдать на скриншоте выше, Администратор получил все задачи пользователя "Иванов". Задачи, полученные по замещению, выделяются желтым цветом.

7. Ссылки

- Документация Runa WFE [официальный сайт проекта]. URL: <https://runawfe.ru/Docs#docs>