

RunaWFE. Графический редактор бизнес-процессов. Руководство пользователя.

Версия 3.0

© 2004-2011, ЗАО “Руна”. RunaWFE является системой с открытым кодом и распространяется в соответствии с LGPL лицензией (<http://www.gnu.org/licenses/lgpl.html>).

Введение

RunaGPD — это графический редактор бизнес-процессов для открытой системы управления бизнес-процессами RunaWFE. RunaGPD является частью открытого проекта RunaWFE, свободно распространяется под LGPL лицензией. RunaGPD может выполняться на различных платформах (Linux, Windows и т.д.).

Редактор можно свободно загрузить вместе с исходными кодами с портала sourceforge по адресу <http://sourceforge.net/projects/runawfe/files> ^[1]

В настоящем документе описывается как разрабатывать бизнес-процессы при помощи RUNA GPD.

Инструкция по установке

RunaGPD распространяется как часть системы RunaWFE в следующих вариантах:

1. В виде специализированных дистрибутивов для конкретных операционных систем
2. В виде исполнимых файлов java-машины.
3. В исходных кодах.

Специализированный дистрибутив для ОС Windows

Вставьте диск в дисковод (в случае дистрибутива на CD-диске) или запустите на выполнение файл RunaWFE-Installer.exe (в случае дистрибутива в виде исполняемого файла). Следуйте инструкциям появившегося на экране мастера установки. В процессе установки не забудьте отметить галочкой компонент — графический редактор бизнес-процессов.

После установки RunaGPD с графическим редактором можно работать через меню системы: “Пуск/Программы/RunaWFE/Process designer” и значок с надписью “Process designer” на рабочем столе.

Специализированный дистрибутив для ОС Linux

Для установки редактора процессов из rpm или deb пакетов наиболее целесообразно использовать какой-либо менеджер пакетов, например apt-get или yum. Использование менеджера пакетов позволит автоматически загрузить и установить зависимости для устанавливаемых пакетов.

Графический редактор устанавливается при помощи установки пакета runawfe-gpd (например, можно выполнить `apt-get install runawfe-gpd`)

После установки RunaGPD, с графическим редактором можно работать через меню системы и значок с надписью “Process designer” на рабочем столе.

Вариант распространения в виде исполнимых файлов java-машины

Необходимое программное обеспечение

- JRE или JDK 5.0 (или выше), может быть загружено: <http://java.sun.com/j2se/1.5.0/download.jsp> ^[2]
- Web-браузер

Замечание. В случае операционной системы AltLinux 4.0 и Web-браузера Mozilla Firefox для корректной работы встроенного в редактор конструктора форм в папке, в которую установлена мозилла надо найти файл конфигурации firefox.js (путь к файлу: `usr/lib/firefox/defaults/pref/firefox.js`). В этом файле надо установить:

`Dom.disable_window_status_change - false`

Замечание. Если используется специализированный дистрибутив для AltLinux, то изменять файл не нужно. Эта проблема в специализированном дистрибутиве решена.

Установка RunaGPD

Разархивируйте `runa-gpd-*.zip` архив и перейдите в `gpd-x.x.x` папку

Запуск RunaGPD

Запустите исполнимый файл `runa-gpd` (в случае Windows — `runa-gpd.exe`) Замечание. Для запуска редактора в многопользовательском режиме надо вызвать исполнимый файл с опцией `-data` и параметром - ссылкой на папку, в которой будут находиться данные:

`runa-gpd -data "ссылка"`

Например: `./runa-gpd -data "I:/MyProcesses"`

Сборка из исходных кодов

Необходимое программное обеспечение

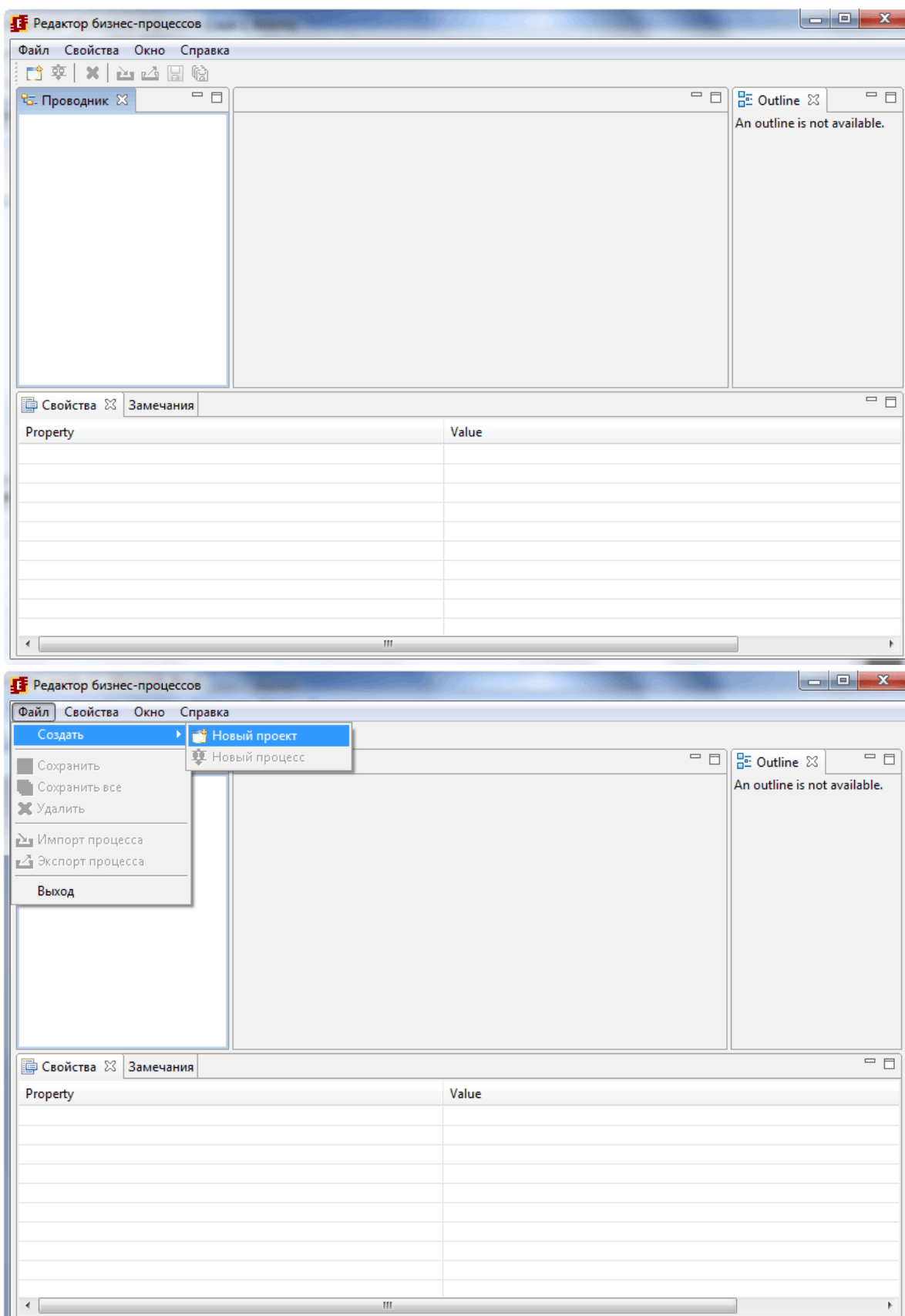
- JDK 5.0 или выше, может быть загружено: <http://java.sun.com/j2se/1.5.0/download.jsp> ^[2]
- Eclipse IDE 3.5, может быть загружено: <http://www.eclipse.org/downloads/> ^[3]

Установка RunaGPD

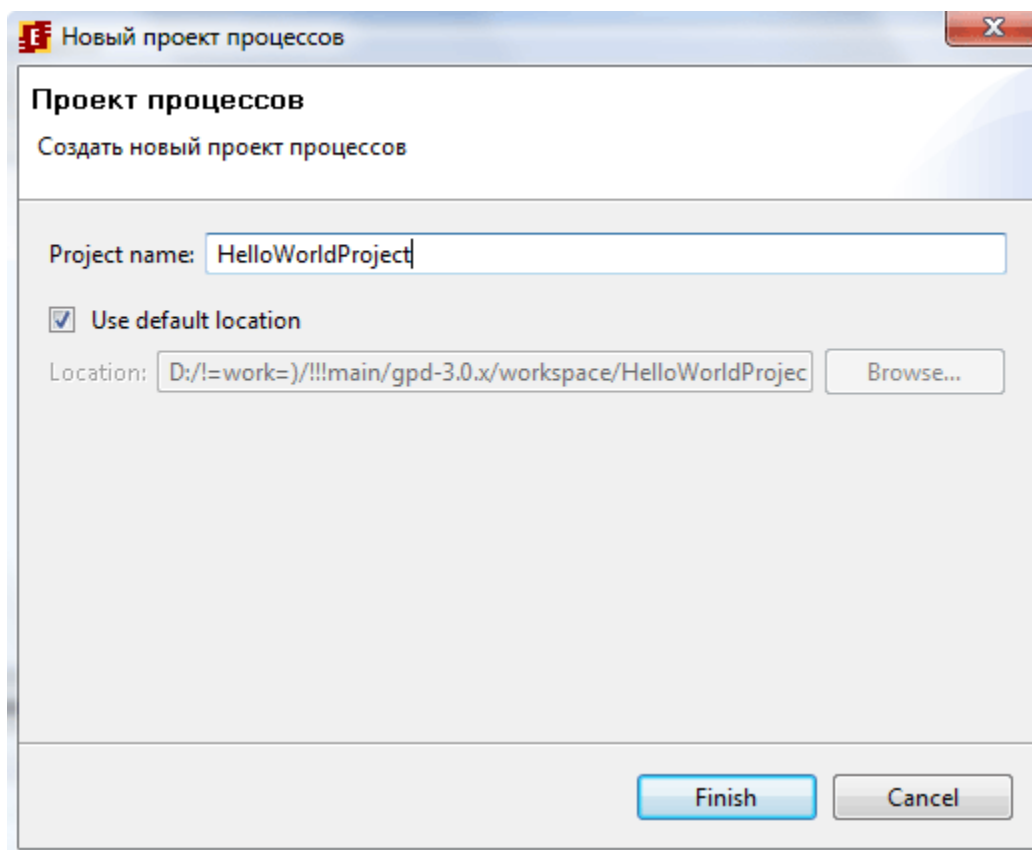
Установка подробно описана в документе «RunaWFE. Графический редактор бизнес-процессов. Руководство разработчика»

Создание нового проекта процессов

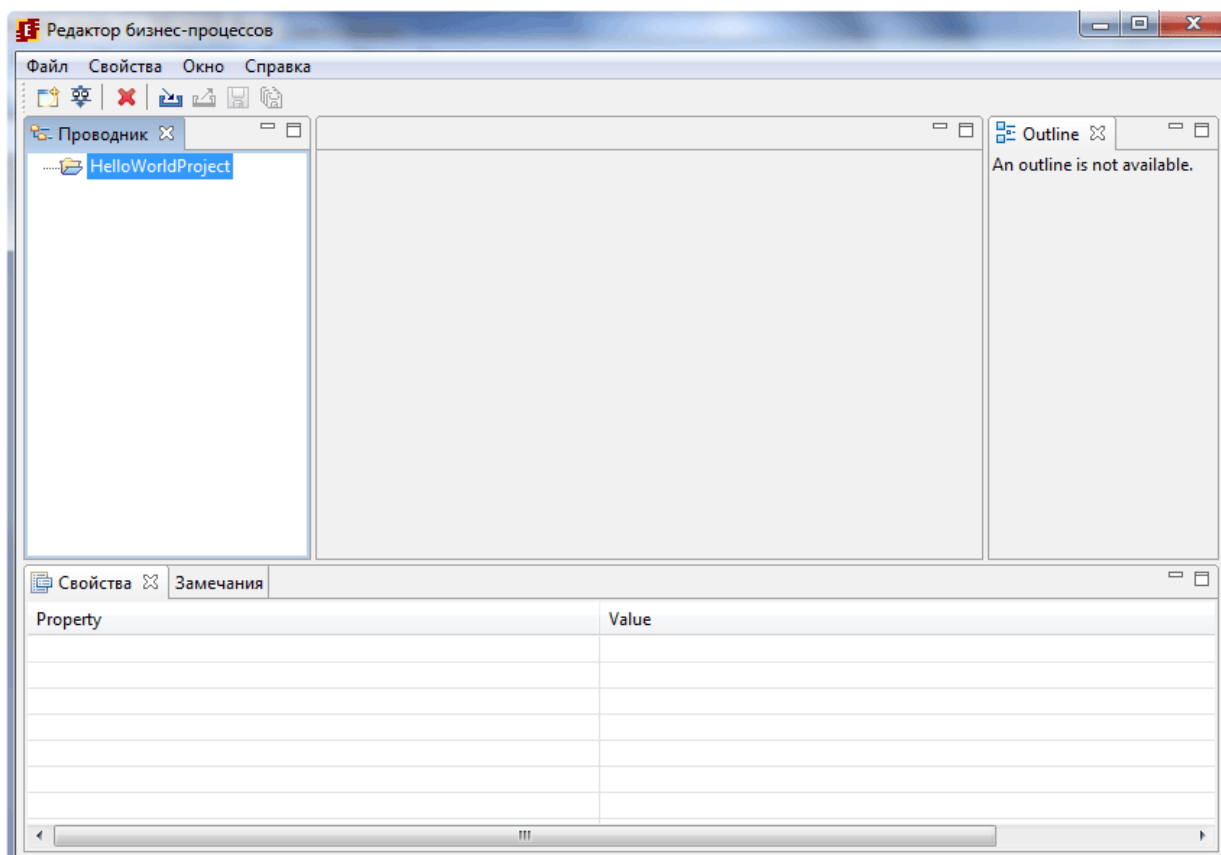
1. Выберите пункт меню **Файл > Создать > Новый проект** - появится wizard заведения нового проекта процессов.



2. Введите имя проекта «HelloWorldProject».



Будет создан проект HelloWorldProject.



Создание процесса HelloWorld

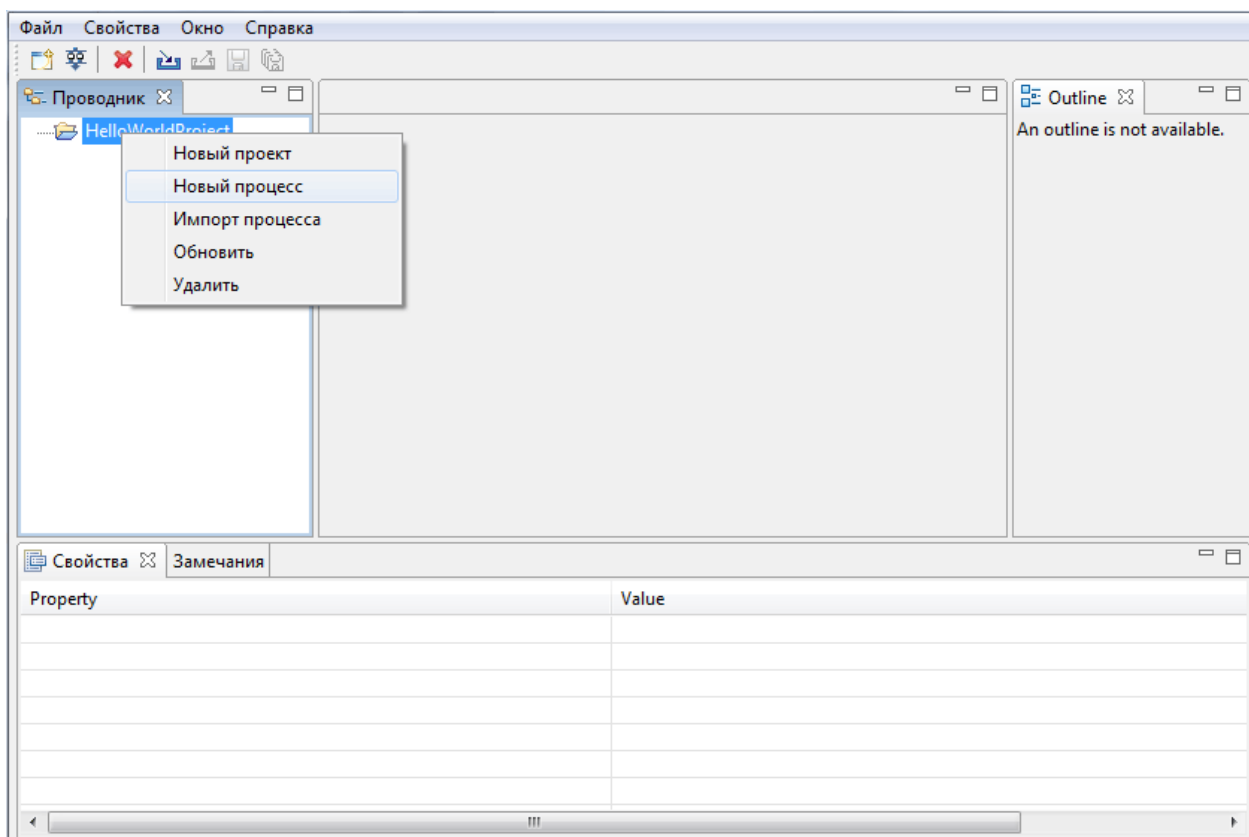
Процесс состоит только из двух узлов: Start-state и Stop-state

Сценарий

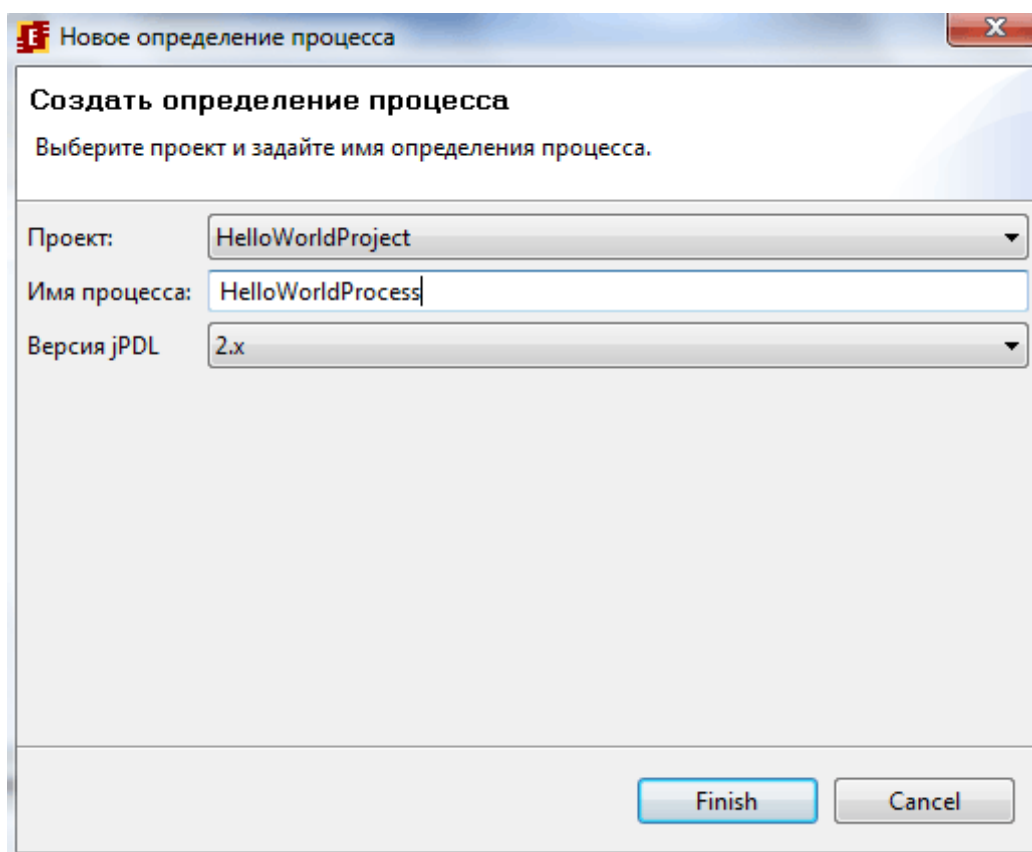
- При запуске процесса появляется стартовая форма HelloWorld.
- После клика на команду “запустить” процесс запускается и тут же переходит в состояние завершения.

Разработка графа процесса

Откройте контекстное меню правой кнопкой мыши, кликнув на HelloWorldProject, затем кликните на «Новый процесс».

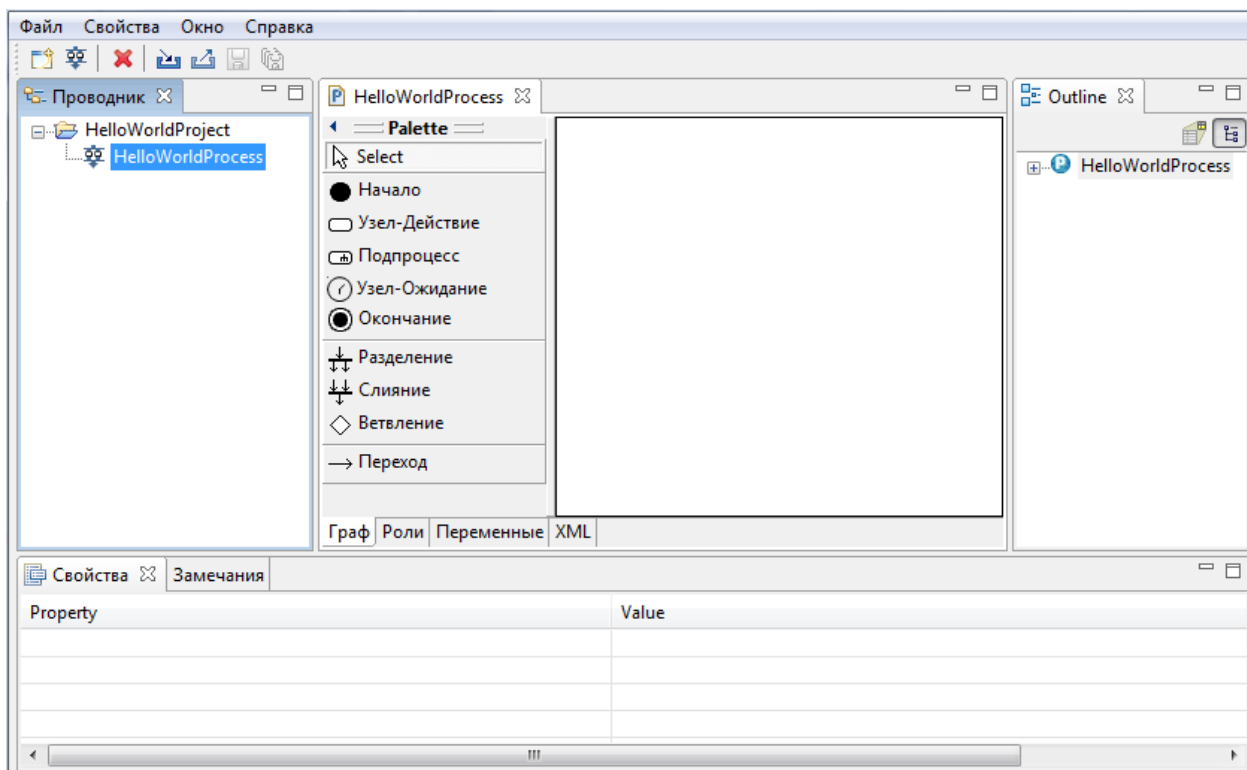


Введите HelloWorldProcess в качестве имени процесса:

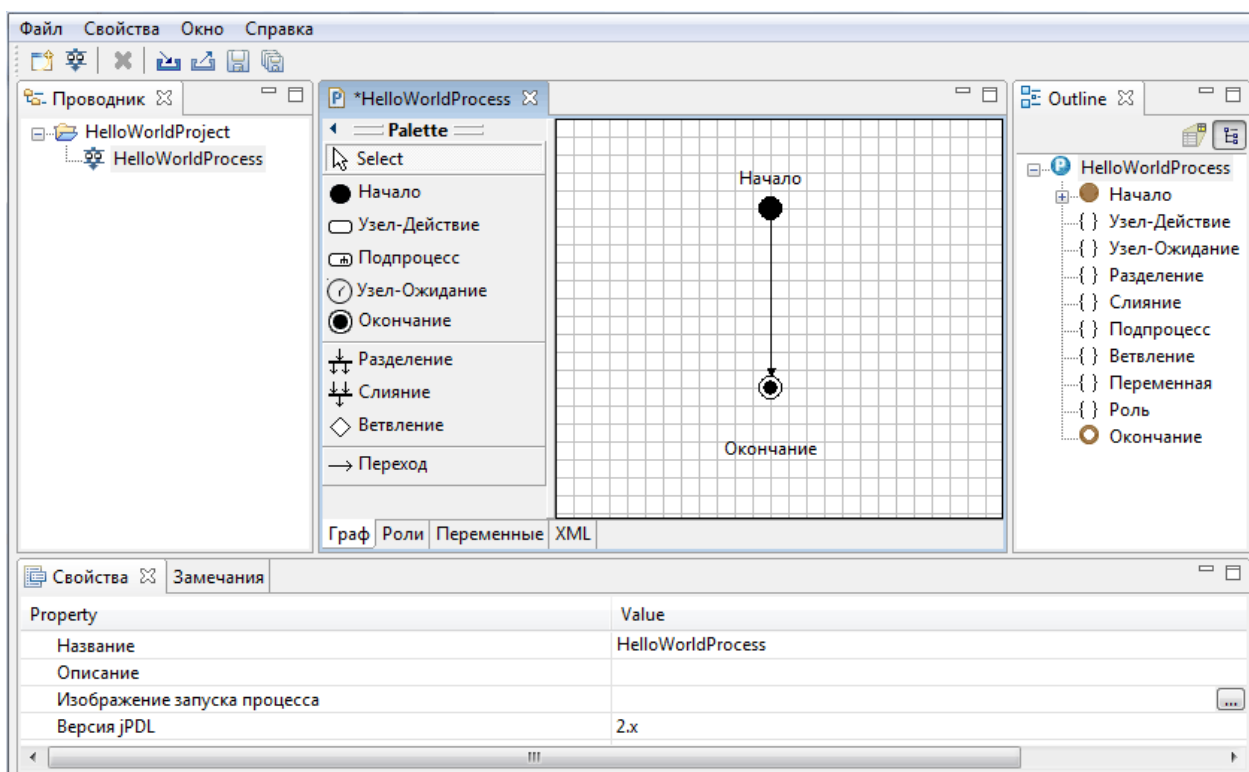


и нажмите «Finish». Будет создан процесс HelloWorldProcess.

Щелкните двойным кликом на HelloWorldProcess. Откроется диаграмма процесса:



Кликните на свойства/показать сетку, появится сетка. Кликните на элементе “Начало” палитры, затем кликните на окне диаграммы. Стартовый узел процесса появится в окне диаграммы. Аналогично поместите на диаграмму завершающий узел процесса “Окончание”, кликните на элемент “Переход” и соедините узлы «Начало» и «Окончание».



Кликните на свойства/показать сетку — сетка исчезнет, в соответствующих полях свойств введите краткое описание процесса (не обязательно) и пиктограмму изображения процесса (не обязательно). - Граф процесса готов.

Создание ролей

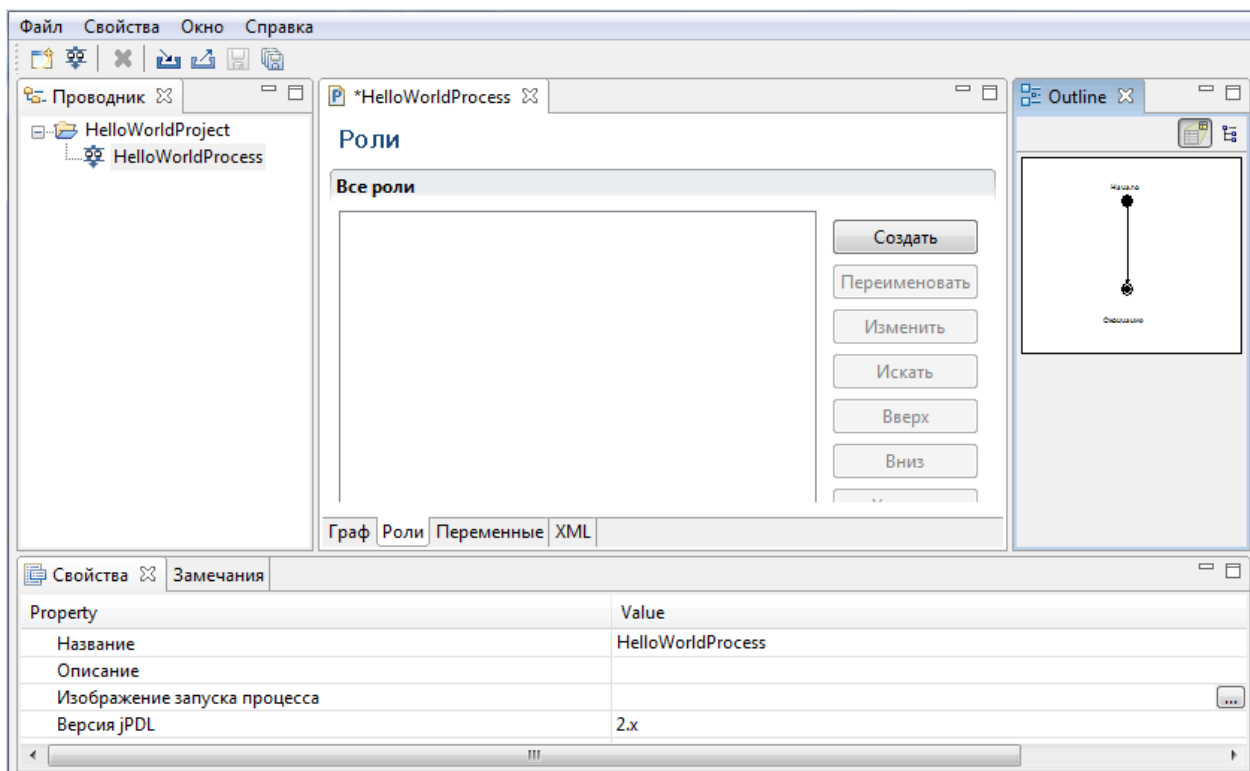
Введение

Swimlanes (роли) соответствуют ролям бизнес-процесса. В JBoss jBPM swimlane (роли) являются специальным видом переменных бизнес-процесса. Каждому узлу-действию в бизнес-процессе должна быть поставлена в соответствие роль. Начальный узел процесса (start state) имеет отличное от остальных узлов поведение в части инициализации ролей: Стартовый узел не использует инициализатор - роль, указанная в стартовом узле, инициализируется пользователем, запустившим бизнес-процесс. В отличие от этого остальные узлы-Действия используют роли для того, чтобы определить, кто может выполнять задание в данном узле-Действии.

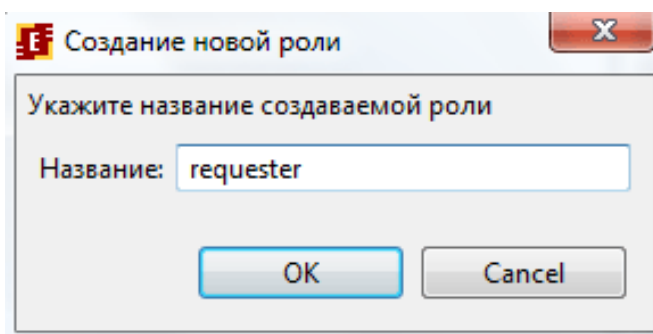
Задание роли для процесса HelloWorld

Процесс состоит только из двух узлов: Начало и Окончание. Для состояния окончания процесса роль не нужна. Поэтому в процессе будет только одна роль.

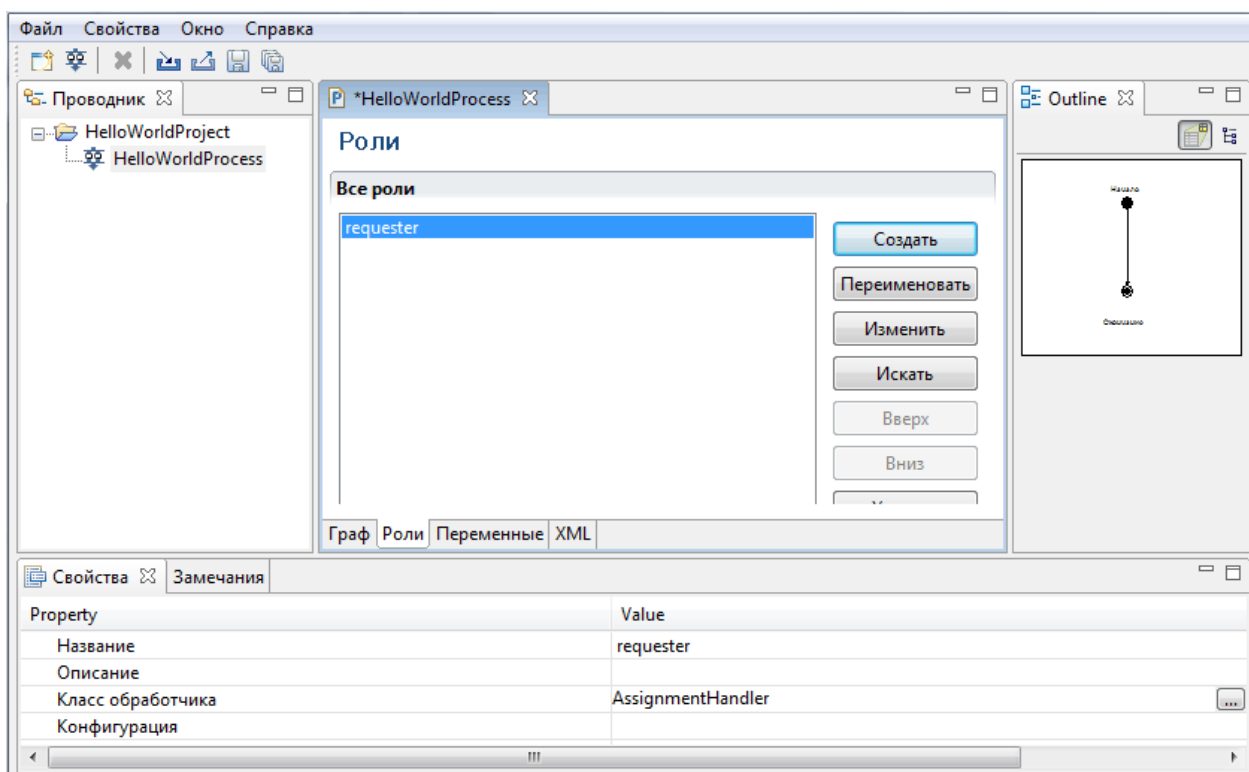
Выберите вкладку «Роли». Появится следующая форма:



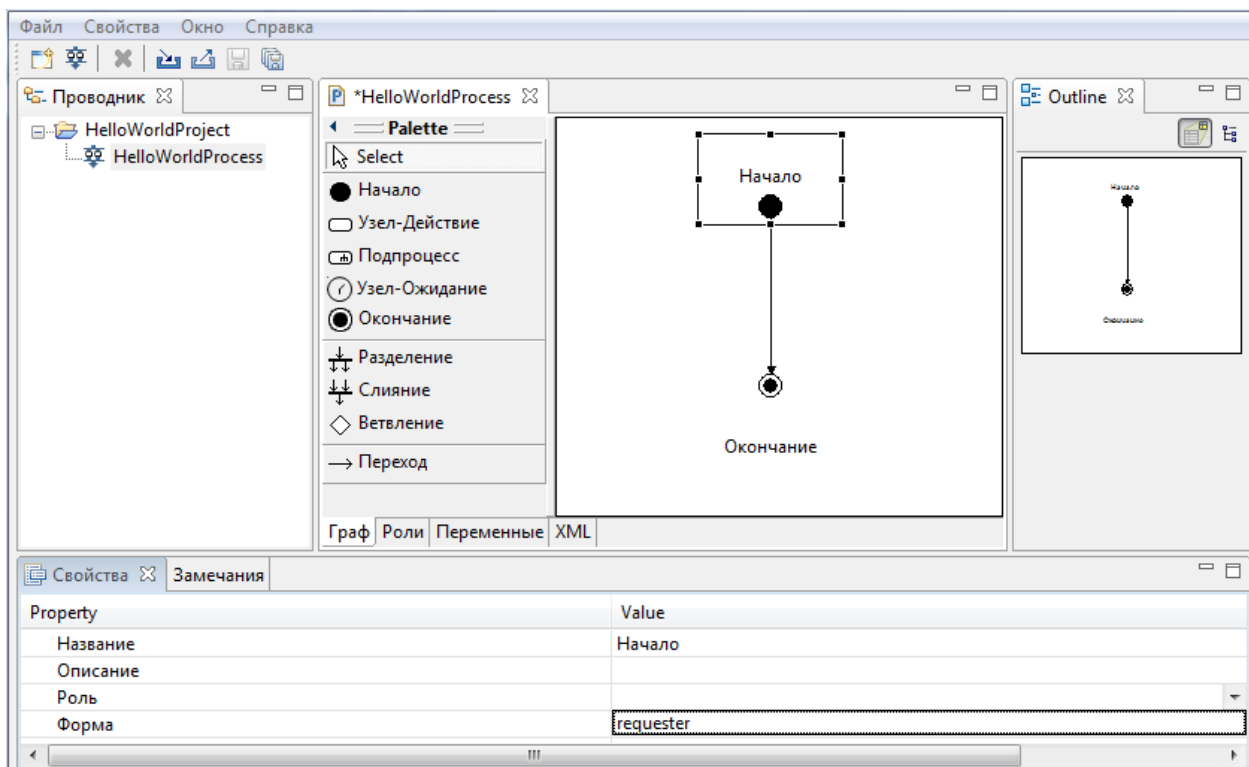
Кликните на кнопке “Создать”. В появившейся форме введите “requester”.



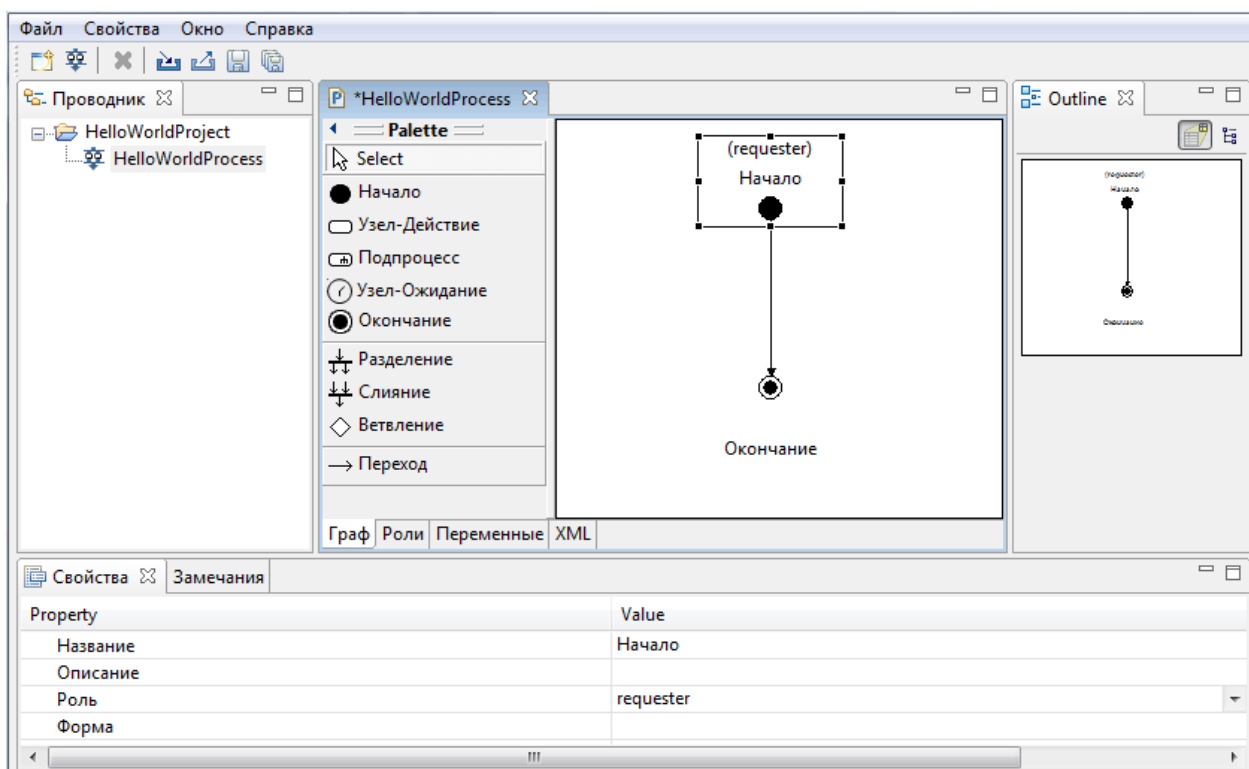
Кликните на OK.



Кликните на закладке «Граф». На графе процесса кликните на узле «Начало», в свойствах узла кликните на поле, находящееся на пересечении Роль-Value, откройте появившийся список и выберите в нем строку requester:



Имя роли появится в круглых скобках над названием узла процесса:



Создание графической формы

Формы

Узлам-действиям, исполнителями заданий в которых являются сотрудники (не боты), должны быть поставлены в соответствие формы. Можно использовать формы двух видов: HTML-страницы с расширением в виде FreeMarker тегов (или FTL-тегов) и HTML-страницы с расширением в виде дополнительного тега: `<customtag>`.

Расширения используются для отображения специфических элементов и отображения значений переменных бизнес-процесса в формах.

В настоящее время рекомендуется в бизнес-процессах использовать формы - HTML-страницы с расширением в виде FreeMarker тегов. Описание FreeMarker можно найти, например, по данной ссылке - <http://ru.wikipedia.org/wiki/FreeMarker>

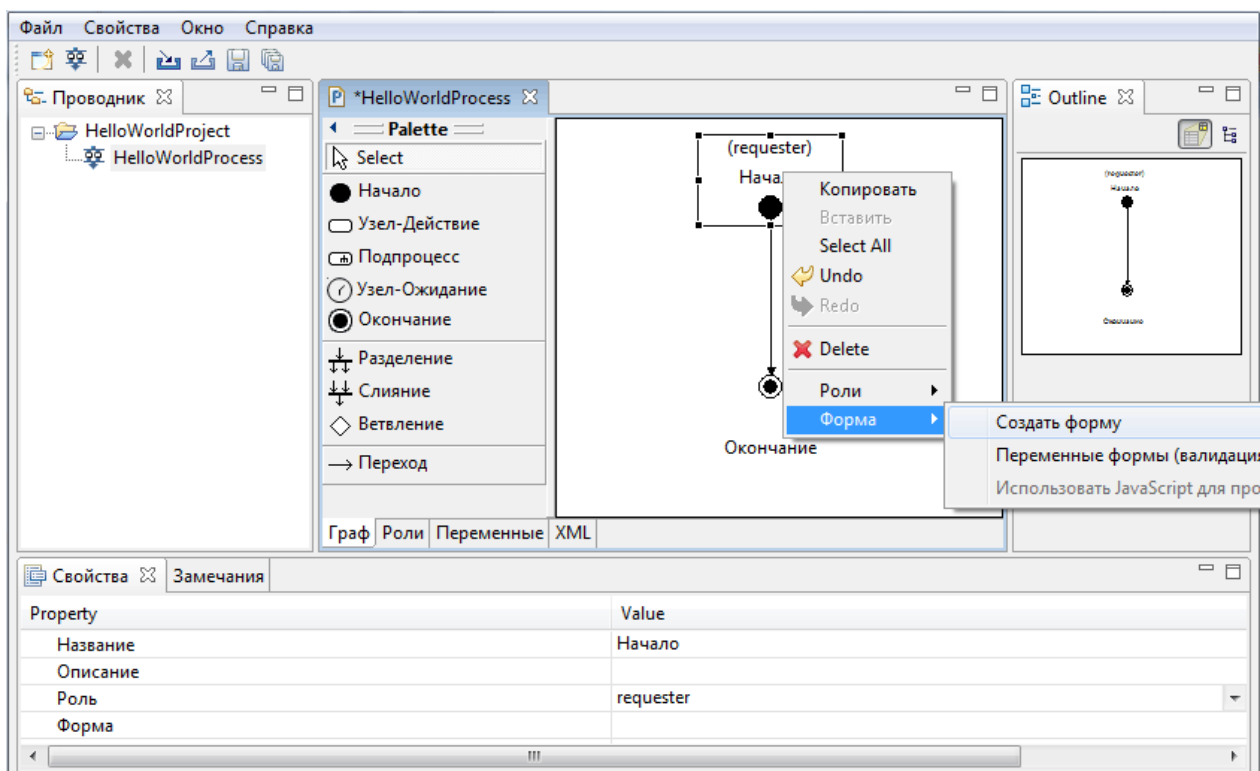
HTML-страницы с расширением в виде дополнительного тега: `<customtag>` поддерживаются для обратной совместимости бизнес-процессов с новыми версиями RunaWFE.

У тега `<customtag>` есть следующие атрибуты:

- `var` – имя переменной бизнес-процесса
- `delegation` – имя Java класса, использующегося для работы с переменной

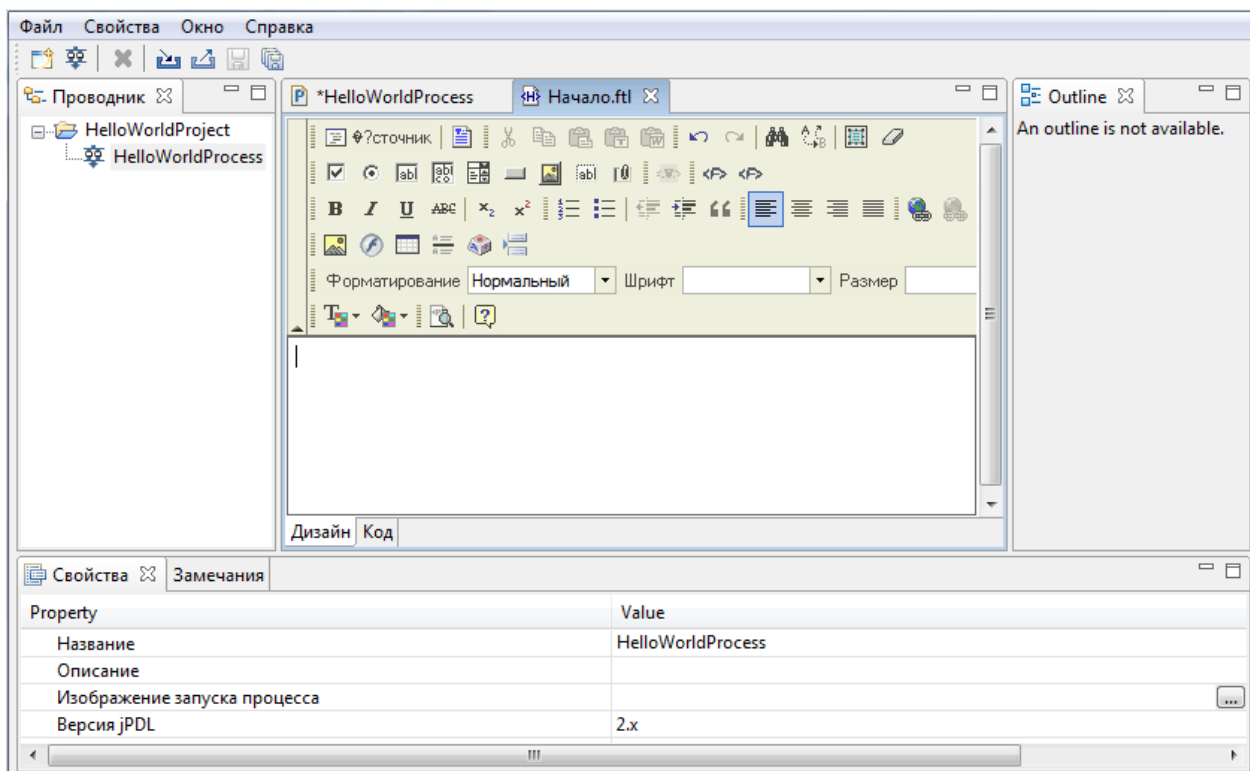
Создание формы

У процесса HelloWorld нет переменных и есть только одна форма – стартовая форма. Кликните на узел «start» графа бизнес-процесса правой кнопкой мыши и выберите команду «Форма» > «Создать форму» (левой кнопкой мыши):



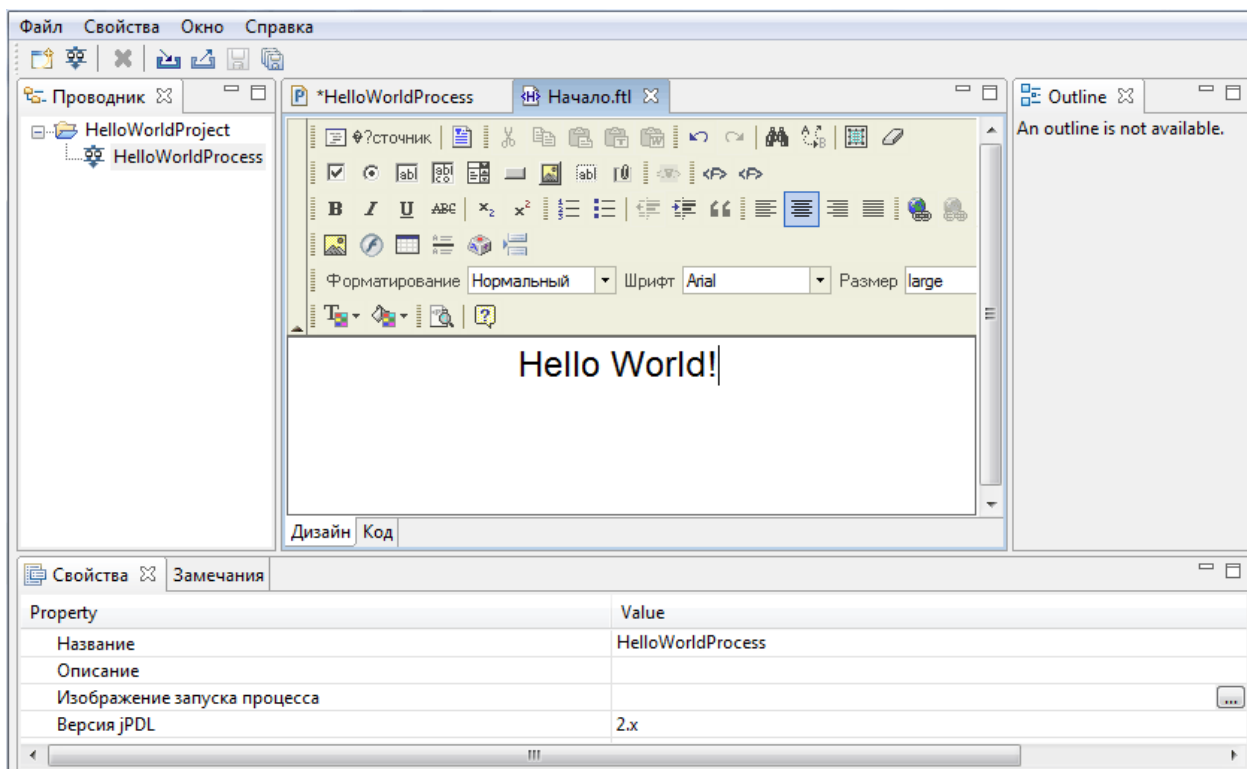
В появившемся выпадающем списке выберите тип формы «fml».

После появления окна редактирования кликните на вкладку дизайн:

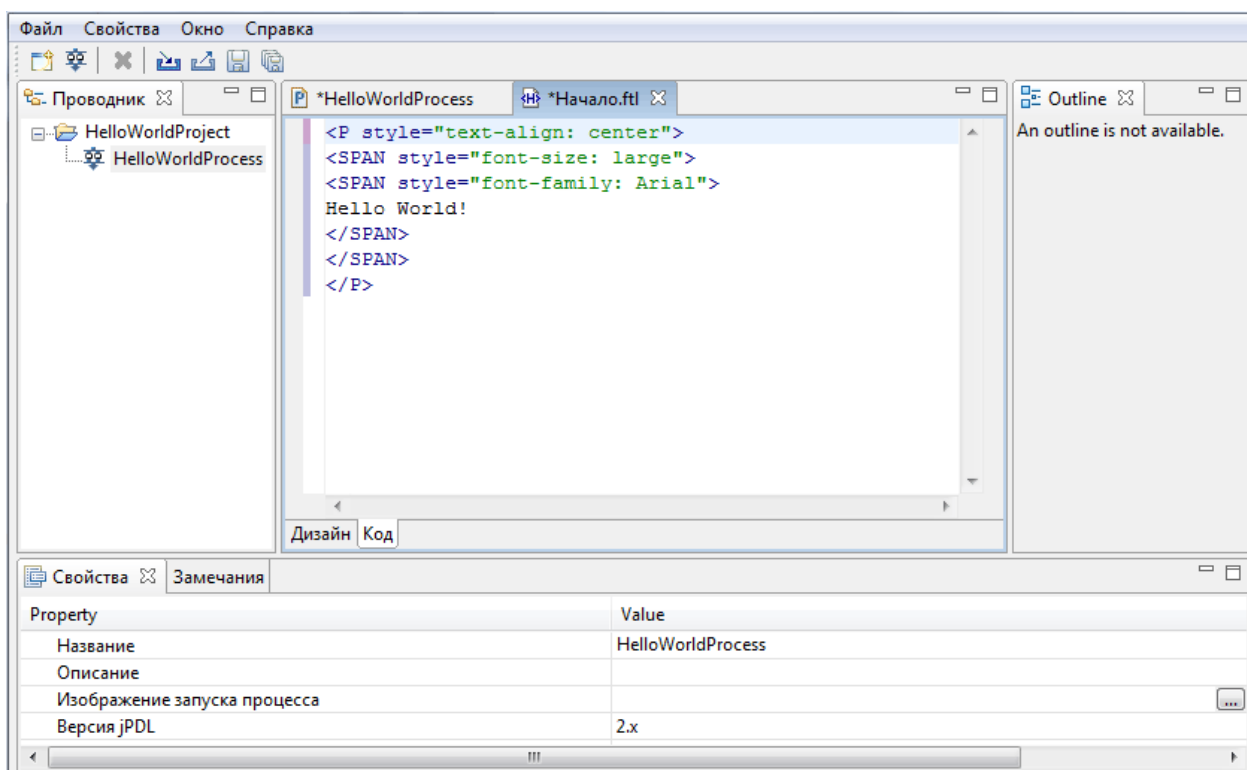


В окне конструктора форм введите Hello World!

Выберите соответствующие шрифт, размер, положение на странице:



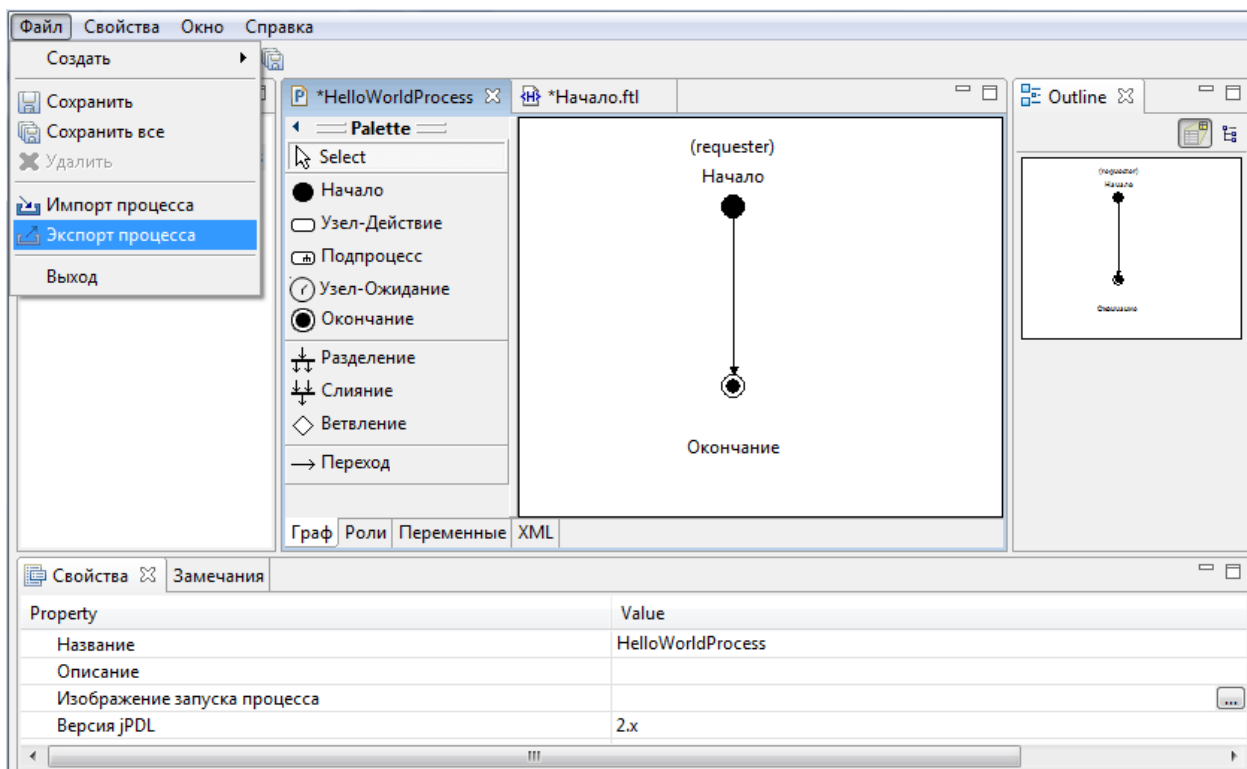
Замечание. Перейдя на вкладку «Код», можно увидеть страницу в виде HTML:



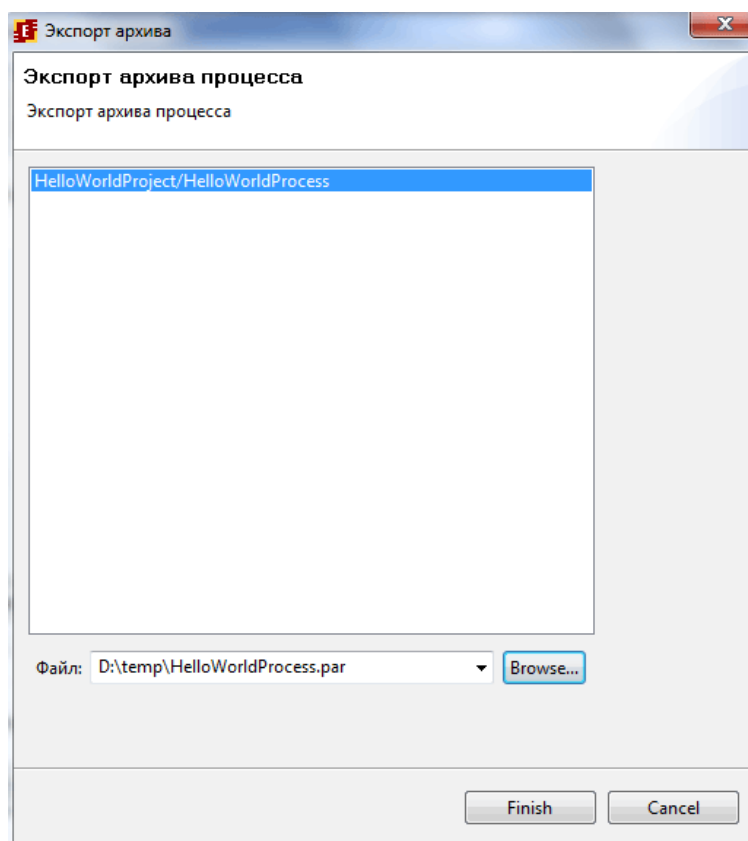
Закройте вкладку формы, подтвердив ее сохранение в диалоге.

Создание файла-архива бизнес-процесса

Выберите HelloWorldProcess, выполните команду Файл/Экспорт чтобы создать файл .par с архивом процесса.



В появившейся форме нажмите кнопку «Browse...», укажите каталог и введите имя файла архива бизнес-процесса:

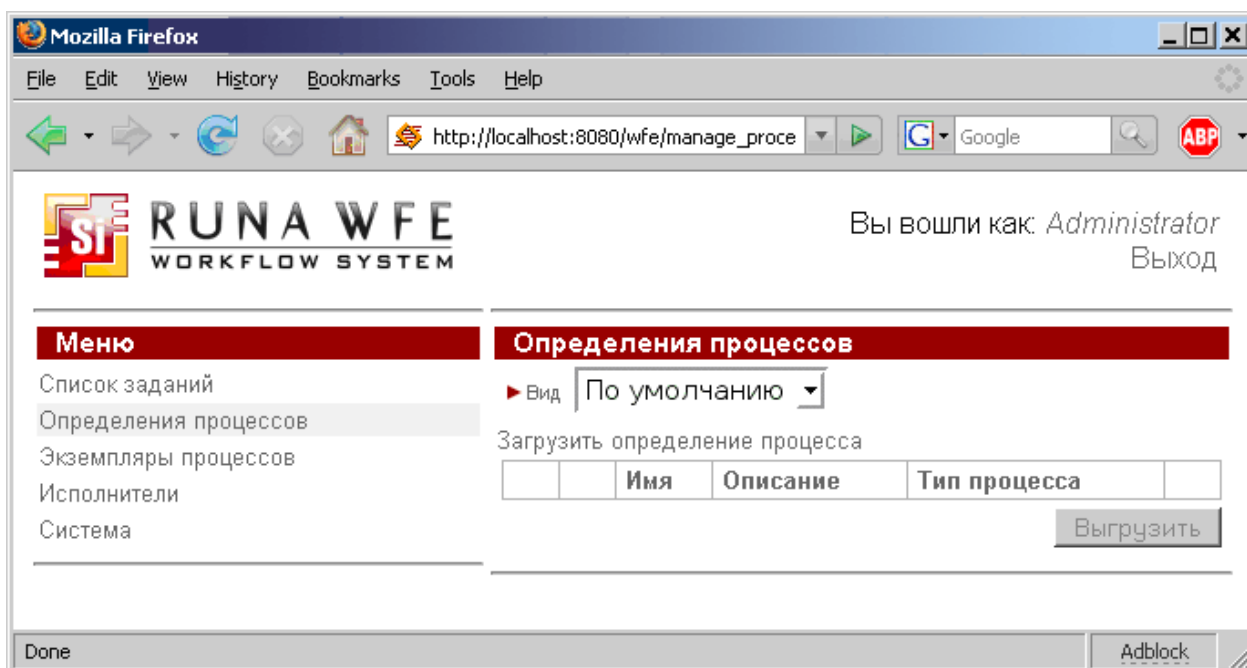


Выполните "finish". В файловой системе будет сформирован файл-архив бизнес-процесса HelloWorldProcess: HelloWorldProcess.par.

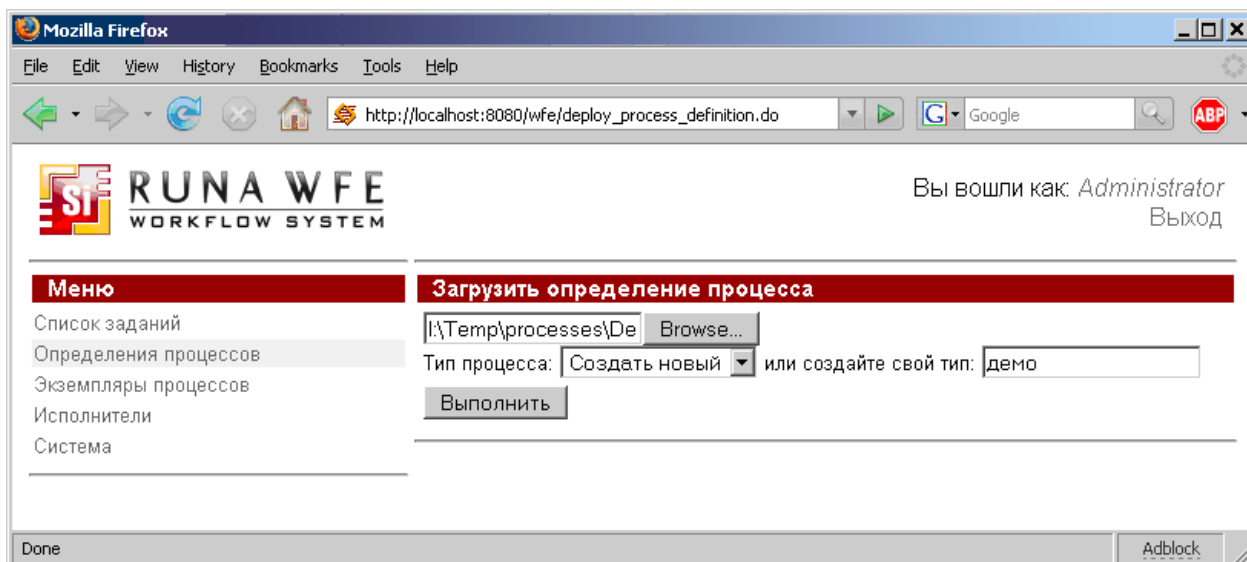
Загрузка бизнес-процесса в систему управления бизнес-процессами

Войдите в систему управления бизнес-процессами RunaWFE как Administrator (Пароль администратора по умолчанию - «wf», см. инструкцию по установке RunaWFE 3.0).

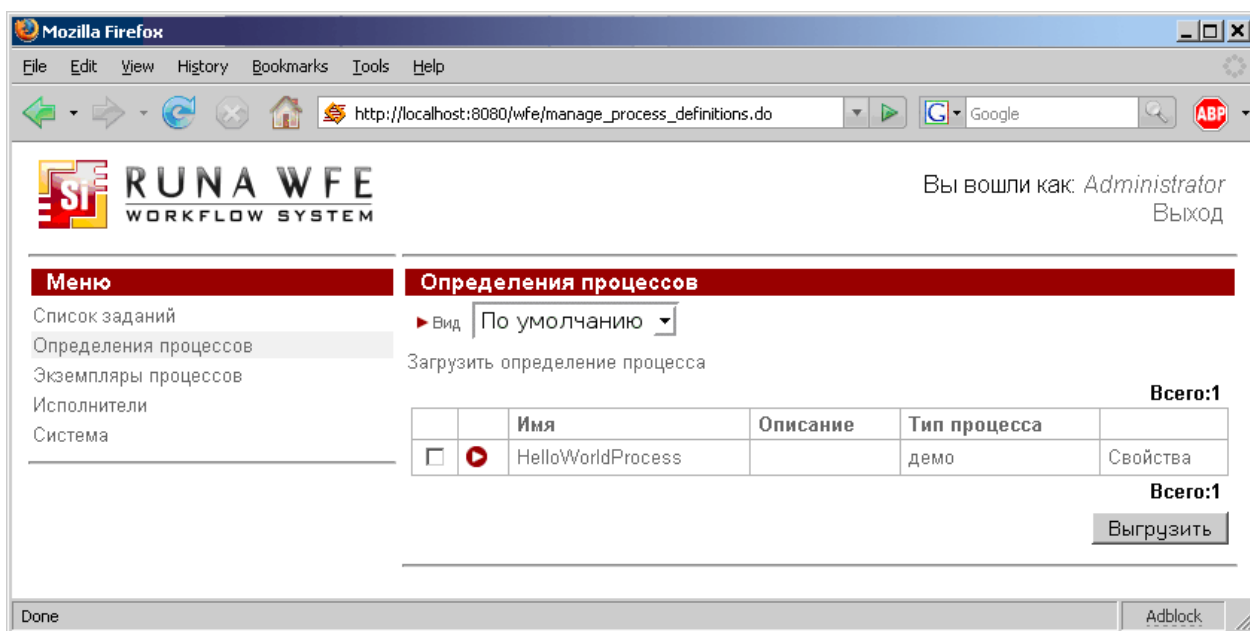
Войдите в меню «Определения процессов».



Кликните на ссылку «Загрузить определение процесса»^[4]. В появившемся диалоге выберите в файловой системе сформированный в редакторе файл-архив бизнес-процесса - HelloWorldProcess.par, впишите тип процесса - «демо» и нажмите «выполнить».

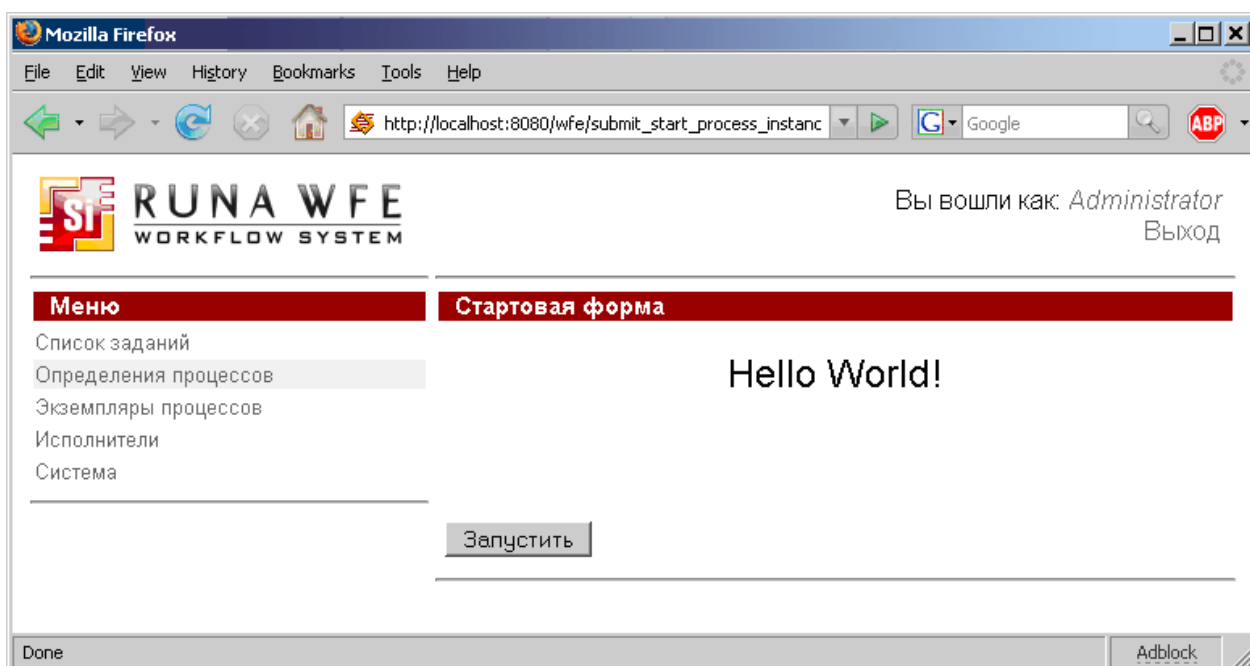


Процесс HelloWorldProcess будет загружен в систему:

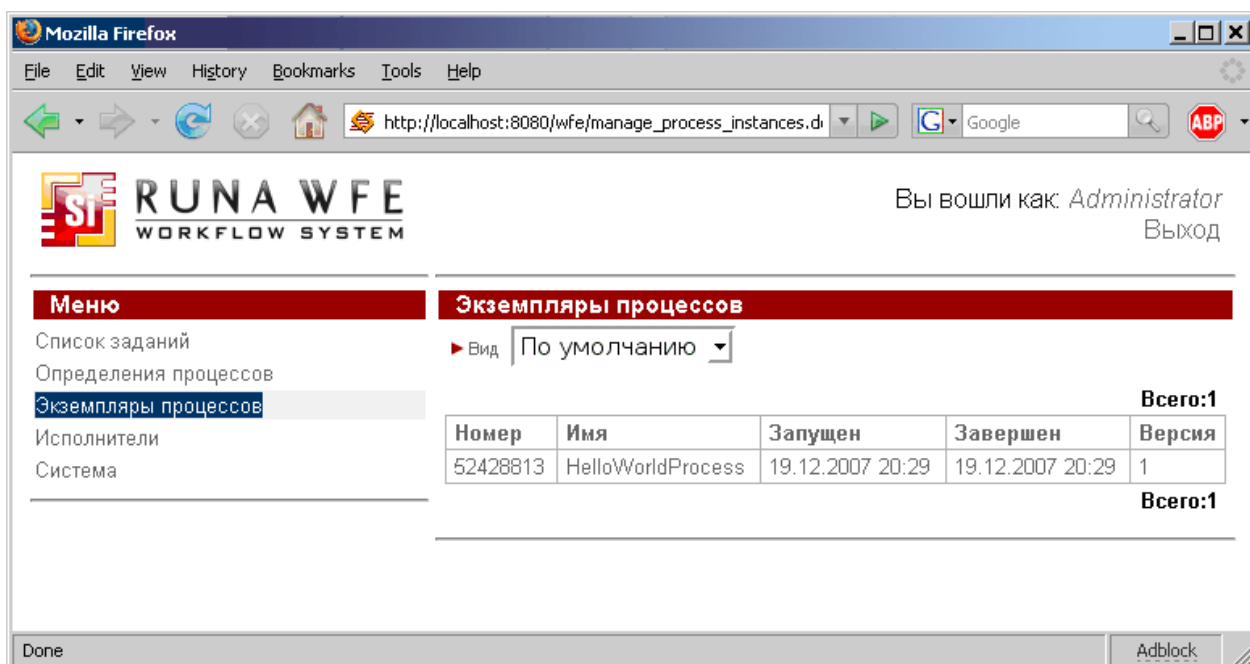


Выполнение процесса

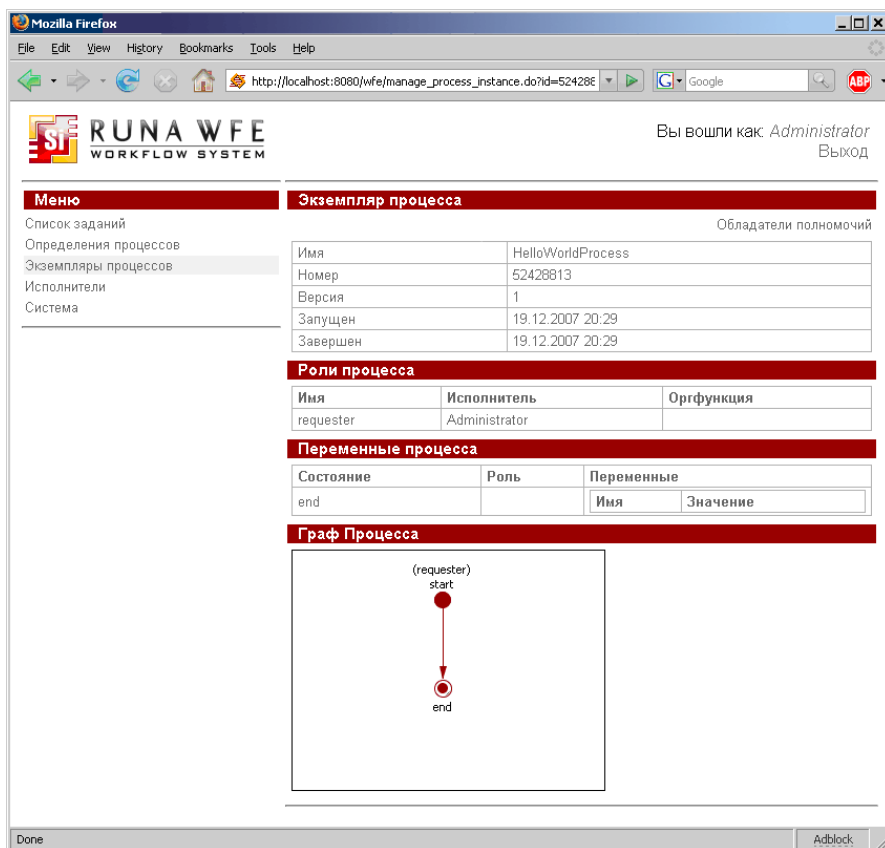
Кликните на имя процесса. Вы увидите стартовую форму:



Кликните на команду «Запустить». Процесс будет запущен и тут же завершится. Можно будет посмотреть экземпляр процесса в меню «Экземпляры процессов»:



Кликните на экземпляр процесса – вы увидите свойства процесса:



Создание процесса «Overtime Work»

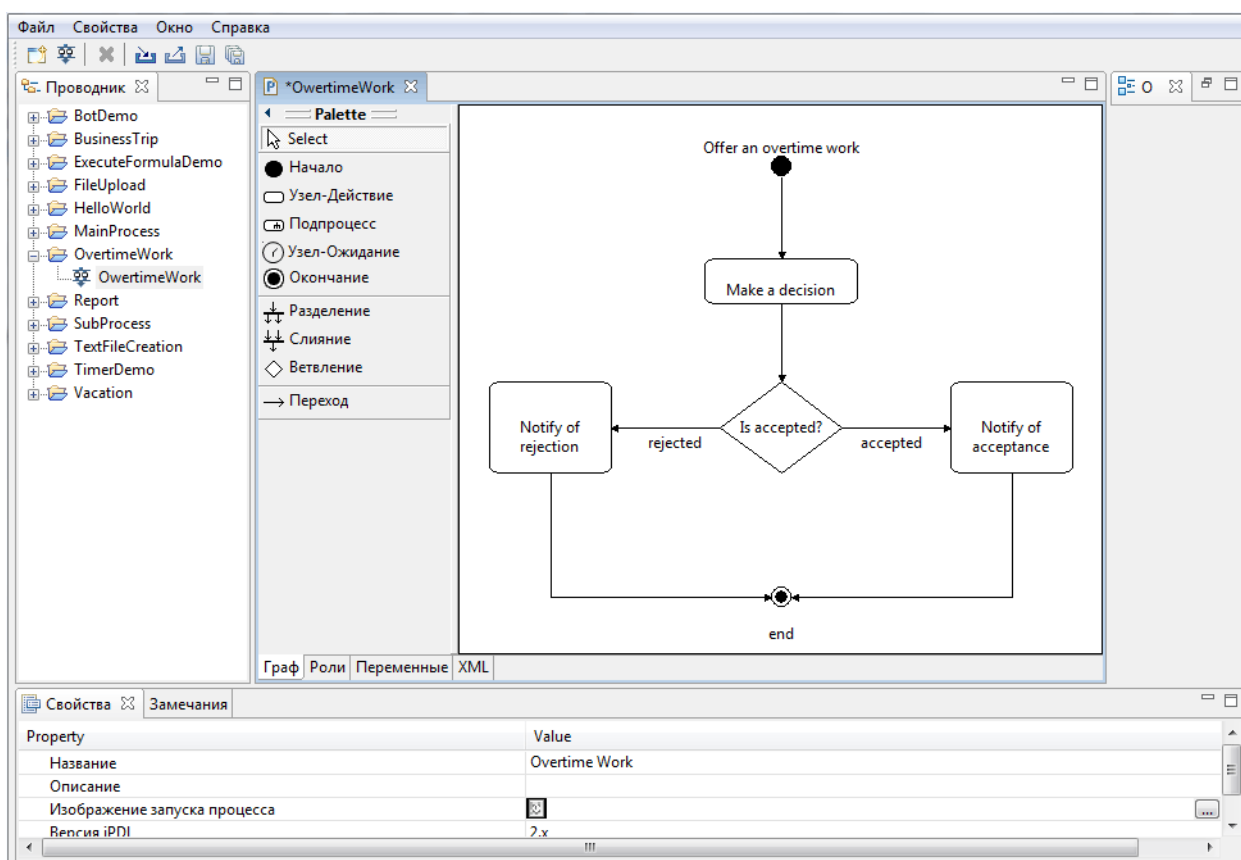
Сценарий процесса

Руководитель предлагает сотруднику выйти на сверхурочную работу. Сотрудник соглашается или отказывается. Далее руководитель знакомится с решением сотрудника.

Предполагается, что руководитель является членом группы “manager”, а сотрудник является членом группы “staff”.

Разработка графа бизнес-процесса

Создайте новый проект «OvertimeWork», затем кликните левой кнопкой мыши на команду «Новый процесс». Введите в появившемся диалоге “over time work demo” в качестве имени процесса. Сделайте двойной клик на появившейся строке «over time work demo». Появится окно диаграммы бизнес-процесса. Выбирая элементы палитры, нарисуйте следующий граф бизнес-процесса:



Для задания подписи элемента отличной от подписи по-умолчанию, выделите элемент кликнув на него, и еще раз кликните на элементе.

Для переходов, выходящих из узла исключаяющего выбора «Is accepted?» в их свойствах задайте явные имена «accepted» и «rejected»:

Создание ролей

Роли

Поведение ролей в начальном и конечном состояниях бизнес-процесса было описано в предыдущем разделе. В других состояниях при помощи ролей определяются исполнители заданий.

Роль (или роль-дорожка) — это специальный тип переменной бизнес-процесса. Роль может быть инициализирована пользователем или группой пользователей в любой момент выполнения бизнес-процесса. К моменту перехода управления в некоторый узел роль, связанная с этим узлом, должна быть инициализирована. Если роль инициализирована пользователем, то после прихода управления в этот узел только этот пользователь получит задание. Если роль инициализирована группой пользователей, то после прихода управления в этот узел все члены группы получают задание. Однако выполнить задание сможет только один член группы, - тот, который первым кликнет на «выполнить». После этого будет произведена доинициализация роли этим пользователем и далее роль будет инициализирована уже не группой, а данным пользователем.

Для автоматической инициализации роли в момент прихода управление в узел-действие служит специальный механизм инициализации, основанный на функциях над организационной структурой, которые возвращают пользователя или группу пользователей, которому будет направлено задание. В системе можно определить набор таких функций. Эти функции также могут зависеть от параметра.

Также можно явно инициализировать роль некоторым значением, например, присвоить значение роли как переменной бизнес-процесса через графическую форму.

«Внутри» бизнес-процесса функции над организационной структурой (далее - оргфункции) должны быть определены в соответствии с используемым в системе RunaWFE языком jPdl. Для работы с оргфункциями в редакторе бизнес-процессов разработаны соответствующие наборы форм.

Если параметрами оргфункций являются значения переменных бизнес-процесса, то имена этих переменных должны быть «окружены» следующей конструкцией: `${<имя переменной>}`.

Задание ролей для процесса «over time work demo»

В бизнес-процессе есть две роли:

- manager
- staff

Описание ролей:

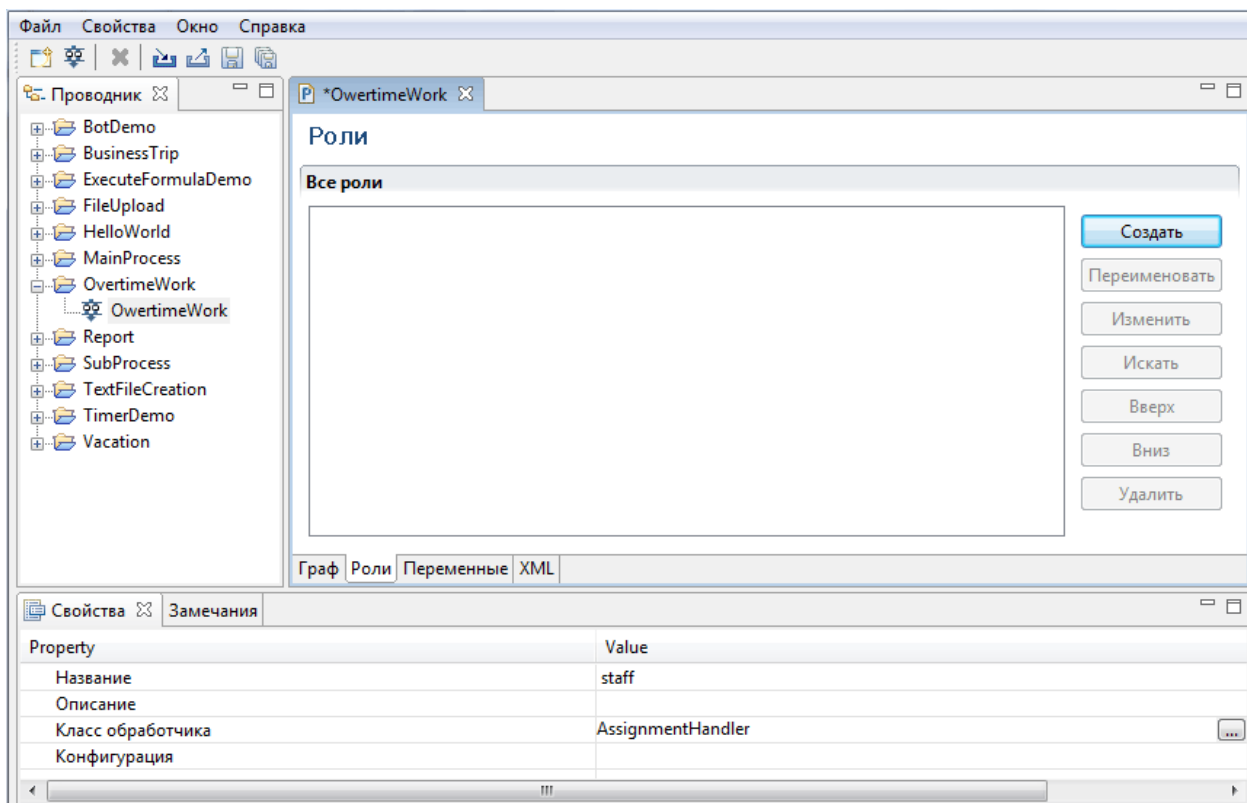
Роль	Описание
manager	Руководитель, который запускает процесс
staff	Сотрудник, которого руководитель выбирает в стартовой форме и которому направляется предложение о сверхурочной работе

Связь узлов графа бизнес-процесса и ролей:

Узел	Роль
Offer an overtime work (предложить сверхурочные работы)	manager
Make a decision (принять решение)	staff
Notify for declining (ознакомиться с сообщением об отклонении)	manager
Notify for acceptance (ознакомиться с сообщением о согласии)	manager

Создание ролей

Кликните на вкладку «роли».



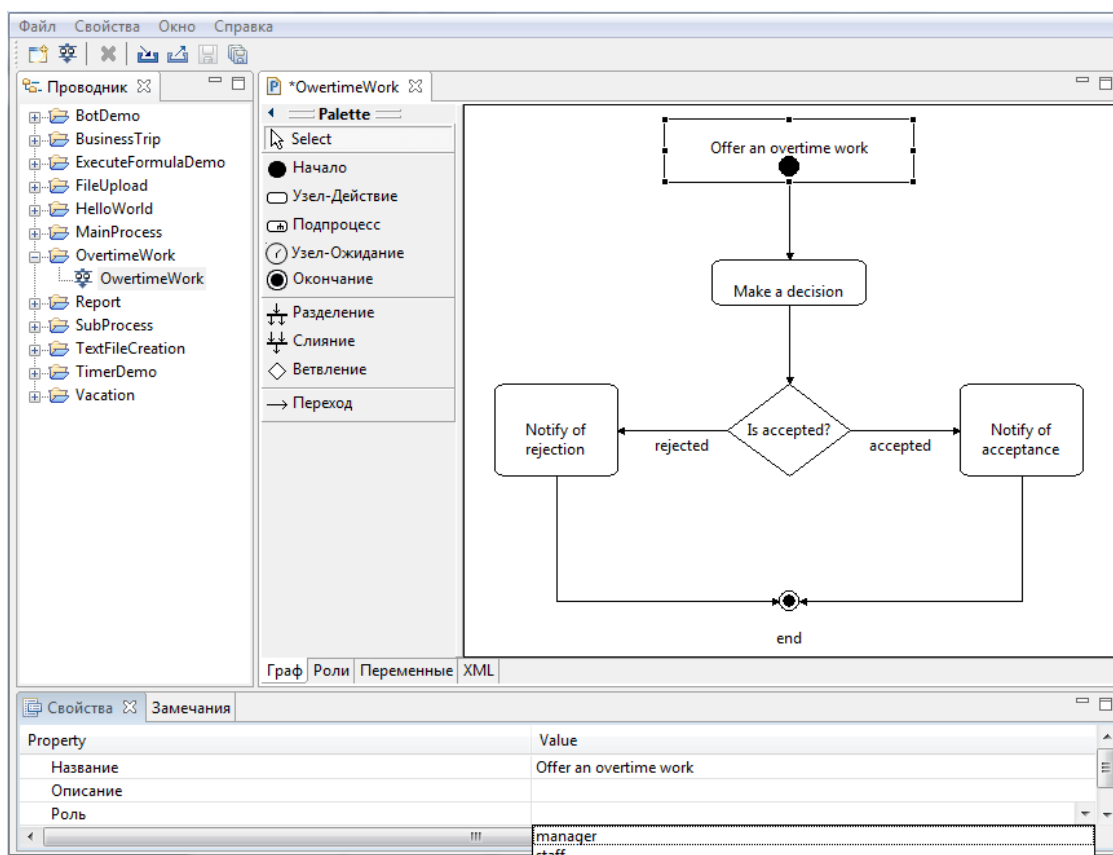
На появившейся странице кликните на кнопку «Создать».

В появившейся форме введите название роли «manager» и кликните на «ОК», т.к. данная роль-дорожка будет использована в стартовом состоянии – т.е. инициализирована пользователем, запустившим процесс. Поэтому роли-дорожке manager не требуется инициализатор.

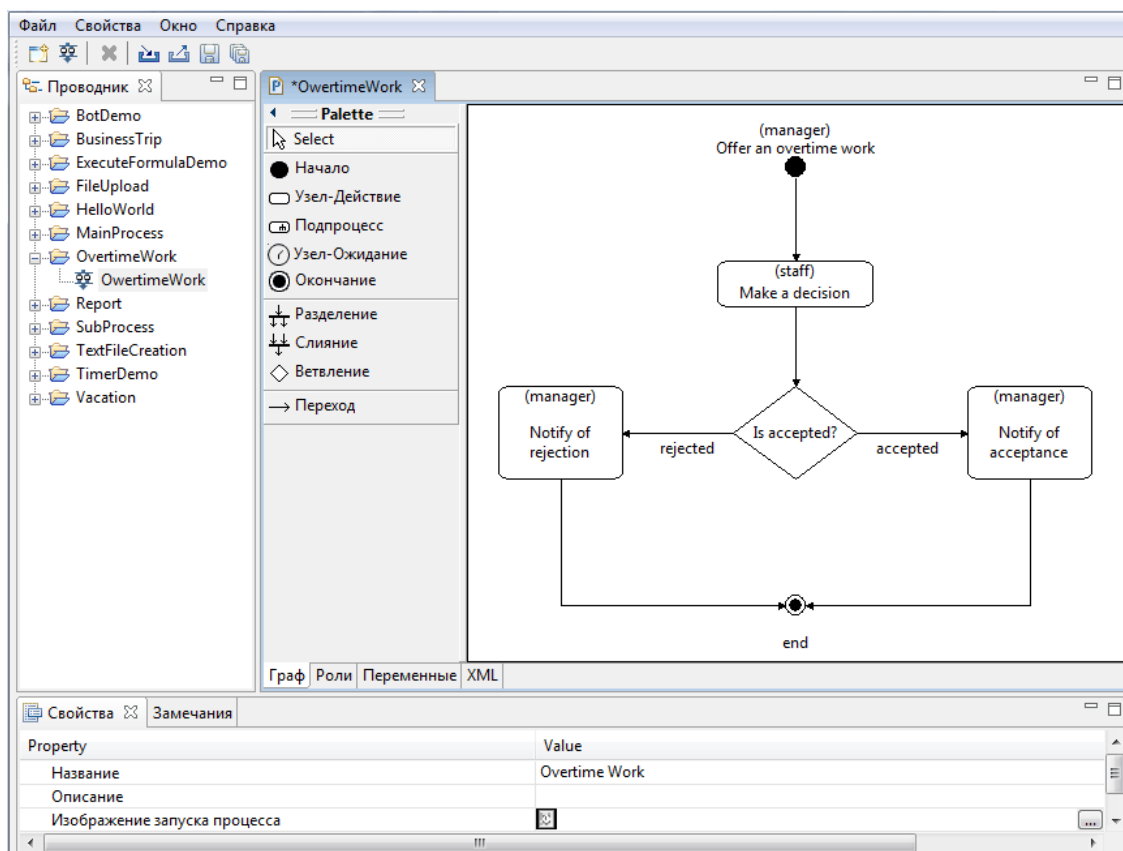
Аналогично добавьте роль «staff».

Замечание. Примеры задания инициализаторов будут отдельно рассмотрены далее в данном документе.

Откройте опять вкладку «Граф», выберите стартовый узел-действие “Offer an overtime work”. В свойствах кликните на правую часть поля Роль/Value. В появившемся списке выберите “manager”.



Аналогично задайте роли во всех остальных узлах-действиях.



Создание переменных

Описание и инициализация переменных

В бизнес-процессе используются следующие переменные:

Переменная	Тип	Описание
since	Дата-время	Дата-время начала сверхурочных
till	Дата-время	Дата-время окончания сверхурочных
reason	Строка	Причина
comment	Многострочный текст	Комментарий
staffPersonDecision	Логический	Решение сотрудника
staffPersonComment	Многострочный текст	Комментарий сотрудника

Переменные

- since
- till
- reason
- comment
- staff

должны быть проинициализированы значениями в стартовом узле-действии “Offer an overtime work”.

Несмотря на то, что Staff является ролью-дорожкой, ее можно инициализировать как обычную переменную.

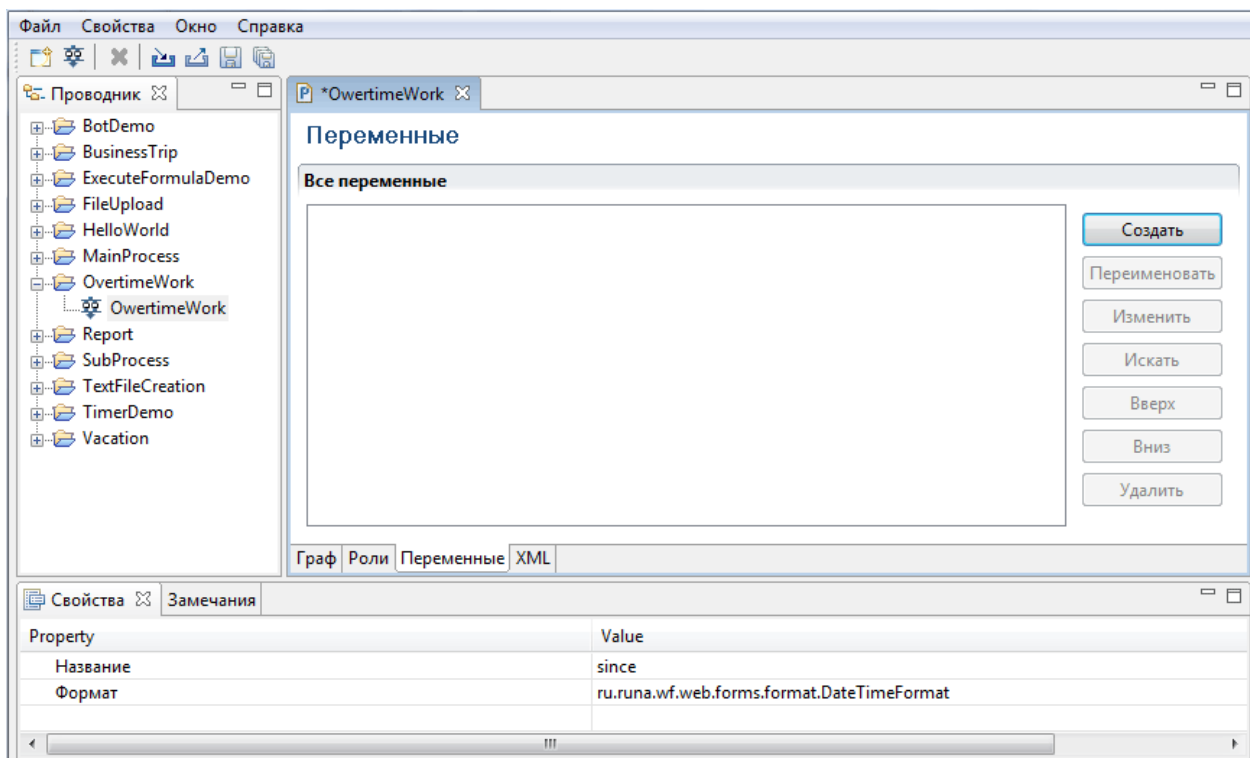
Переменные

- staffPersonDecision
- staffPersonComment

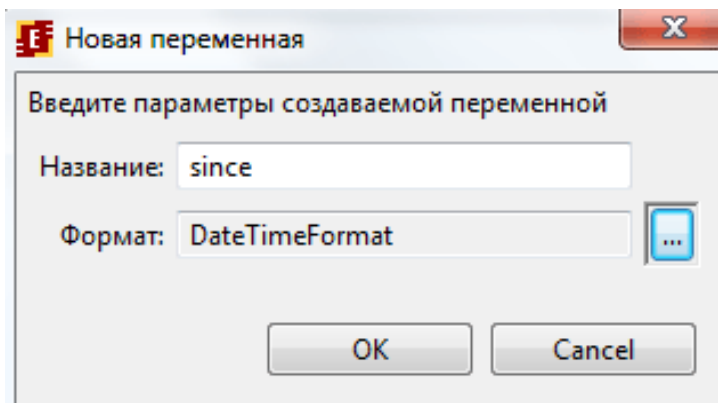
должны быть проинициализированы в узле “Make a decision”

Создание переменных

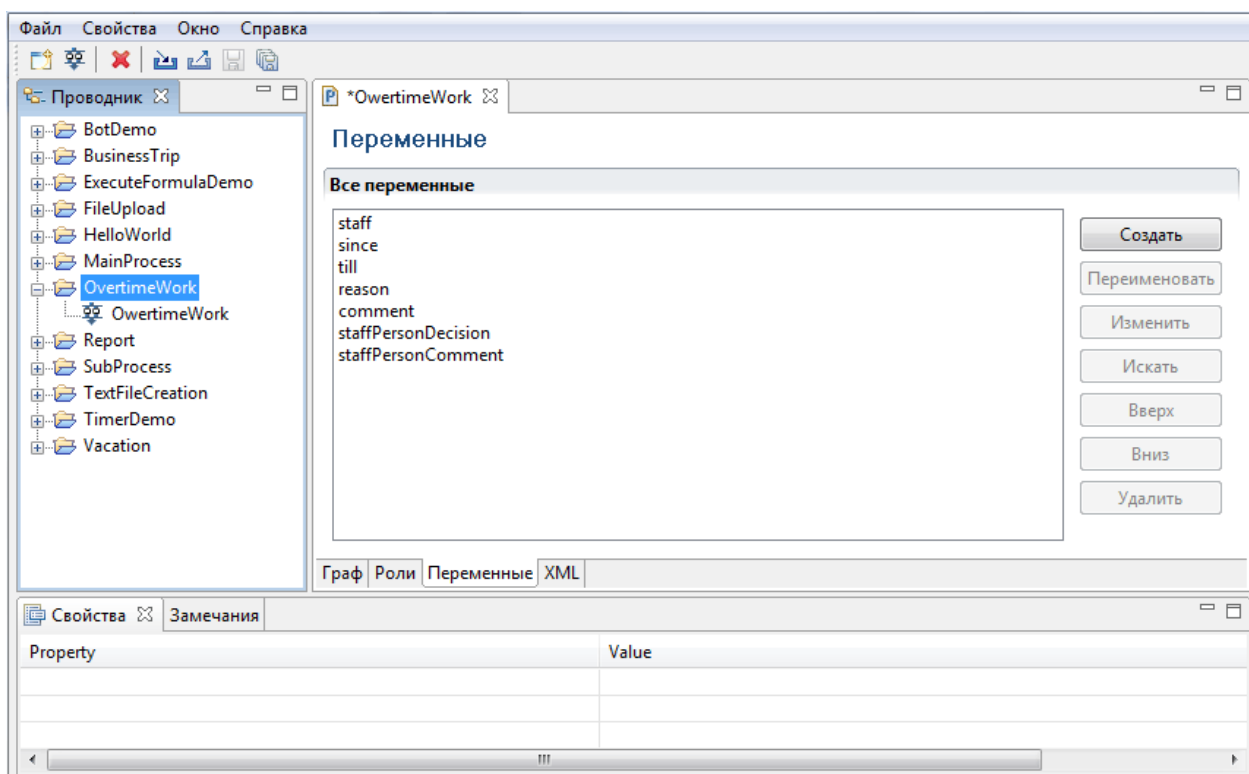
Кликните на вкладке «Переменные». Кликните на кнопке «Создать»



Введите название переменной – since, в качестве формата выберите DateTimeFormat.



Аналогично заведите все остальные переменные бизнес-процесса.



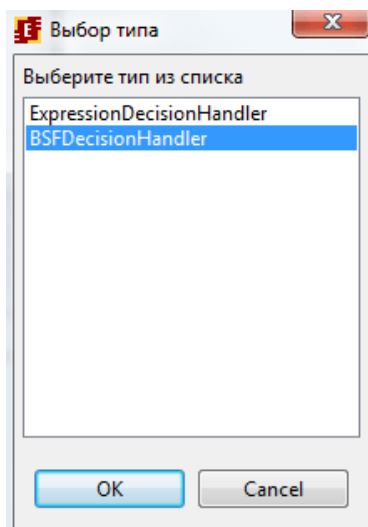
Задание формулы для элемента «Исключающий выбор»

Описание формулы

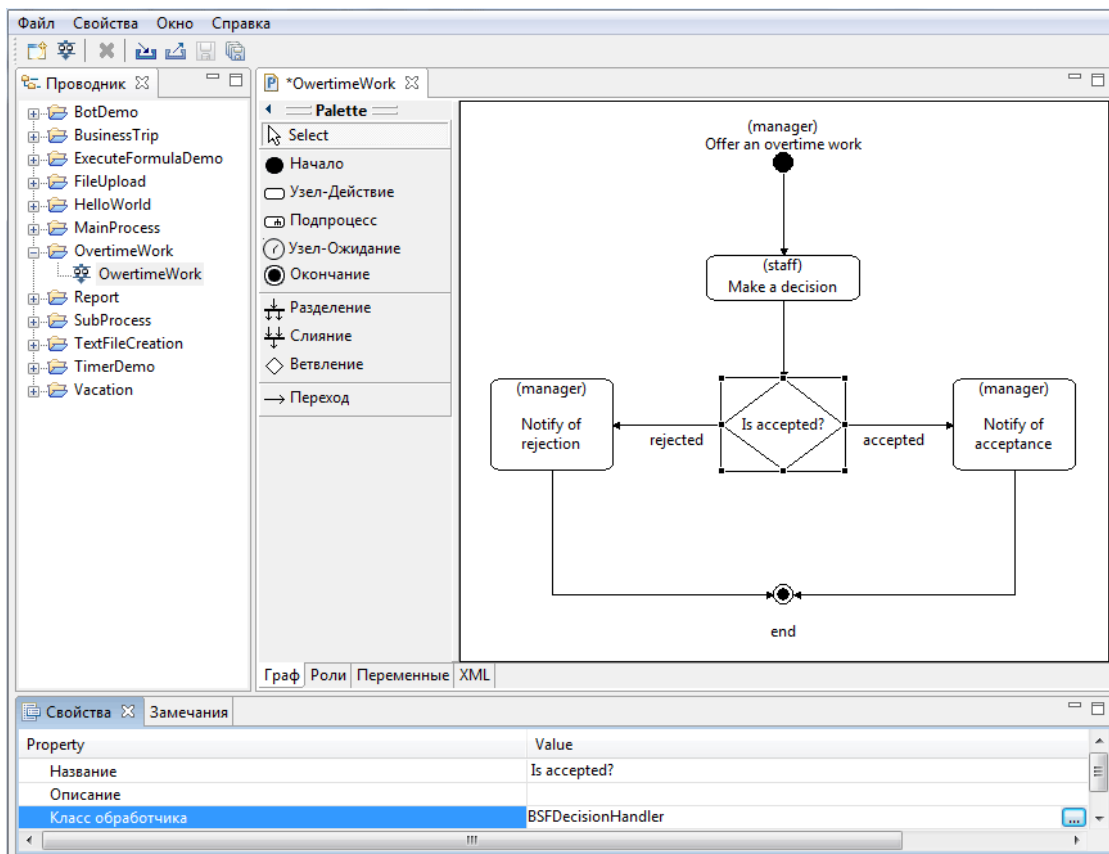
В случае, если значение переменной `staffPersonDecision` (решение сотрудника) - «истина», точка управления должна перейти в узел-действие «Notify of acceptance» (ознакомиться с сообщением о согласии). В противном случае точка управления должна перейти в узел-действие «Notify of rejection» (ознакомиться с сообщением об отклонении).

Создание формулы в редакторе RUNA GPD

Выберите закладку «Граф», кликните на узел исключающего выбора «Is accepted?». В свойствах узла выберите поле, находящееся на пересечении Класс-Value. Кликните на командную кнопку, появившуюся в правой части этого поля. Откроется форма выбора обработчика для решения:



Выберите `BSFDecisionHandler` и нажмите OK. Название обработчика появится в свойствах узла.

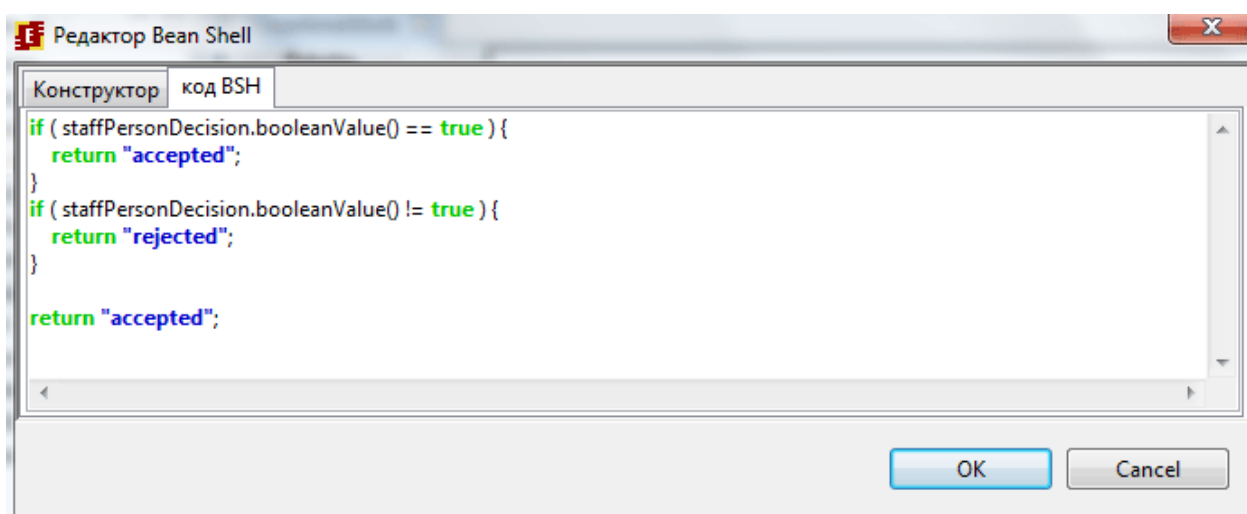


Правой кнопкой мыши кликните на элементе и выберите пункт «Изменить».

В появившейся форме для перехода accepted выберите переменную `staffPersonDecision`, установите операцию сравнения «равно» и значение «true». Для перехода rejected выберите переменную `staffPersonDecision`, установите операцию сравнения «не равно» и значение «true». Для «если ни одно из условий не выполнено» установите переход «rejected»:

Кликните «OK». Формула для узла исключаящего выбора задана.

Замечание. При желании можно посмотреть код формулы. Для этого в данной форме выберите закладку «Код BSH»:



Создание графических форм

Описание форм RunaWFE

Узлам-действиям, исполнителями заданий в которых являются сотрудники (не боты) должны быть поставлены в соответствие формы. Каждая форма записывается в отдельном файле. Можно использовать формы двух видов: HTML-страницы с расширением в виде FreeMarker тегов (или FTL-тегов) и HTML-страницы с расширением в виде дополнительного тега: <customtag>. При создании формы нужно определиться какими дополнительными тегами вы будете пользоваться в форме. Если вы используете <customtag>, то в той же самой форме уже нельзя использовать FreeMarker теги. Расширения используются для отображения специфических элементов и отображения значений переменных бизнес-процесса в формах.

Описание FreeMarker можно найти, например, по данной ссылке - <http://ru.wikipedia.org/wiki/FreeMarker>

HTML-страницы с расширением в виде дополнительного тега: <customtag> поддерживаются для обратной совместимости бизнес-процессов с новыми версиями RunaWFE.

У тега <customtag> есть следующие атрибуты:

- var – имя переменной бизнес-процесса
- delegation – имя Java класса, использующегося для работы с переменной

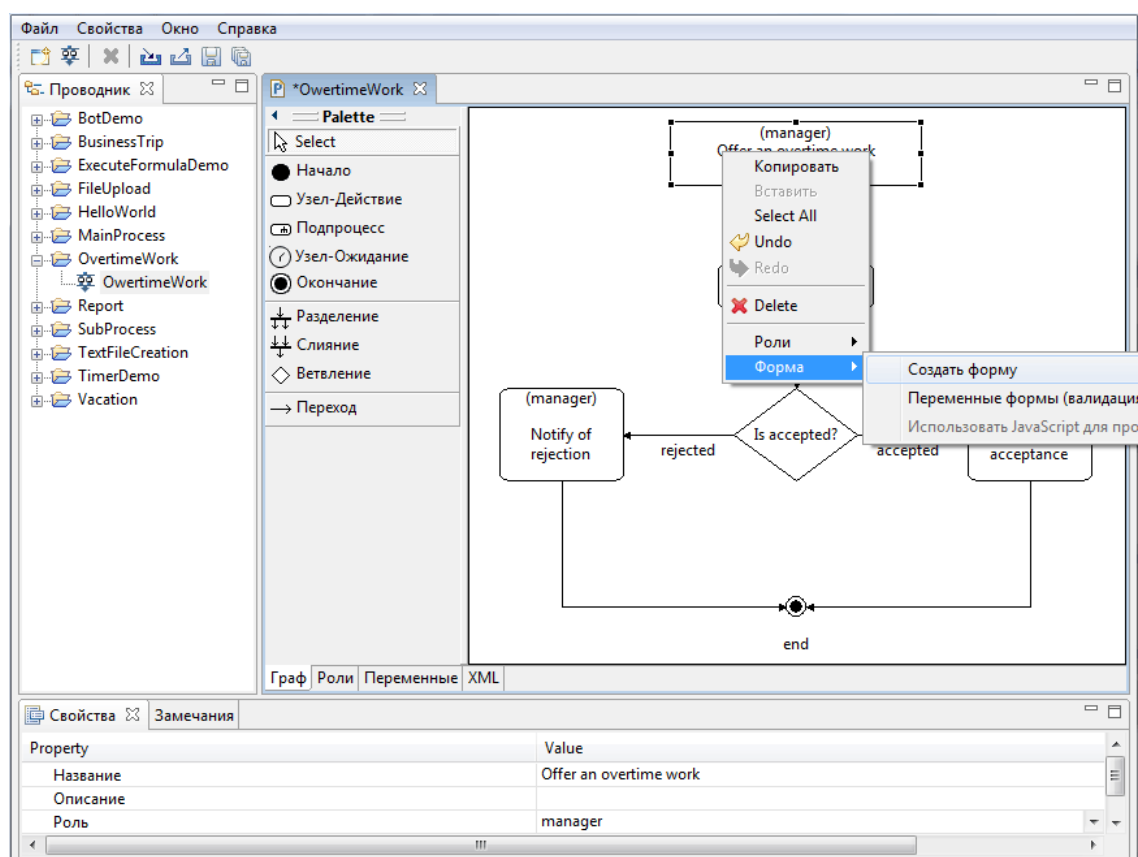
В процессе Overtime Work используются страницы с дополнительным тегом <customtag> со следующими классами:

В процессе Overtime Work используются следующие классы:

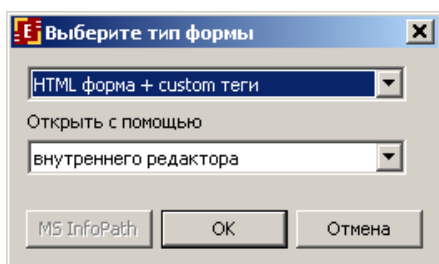
Class	Freemarker analog	Description
GroupMembersComboboxVarTag	Display group	Показывает список членов группы. Имя группы должно совпадать с именем некоторой переменной бизнес-процесса (и соответственно параметром «var» тега). Возвращает ID выбранного члена группы. В процессе Overtime Work используется группа (и, соответственно, переменная) «staff».
DateTimeInputVarTag	Input date-time	Создается поле для ввода даты-времени
ActorFullNameDisplayVarTag	Display actor	Показывает имя пользователя. Параметр «var» тега должен ссылаться на переменную, содержащую ID пользователя.
DateTimeValueDisplayVarTag	Вывод переменной	Выводит значение переменной типа дата-время (показывается в режиме «только чтение»).
VariableValueDisplayVarTag	Вывод переменной	Выводит значение переменной типа строка или число (показывается в режиме «только чтение»).

Создание графических форм при помощи конструктора форм

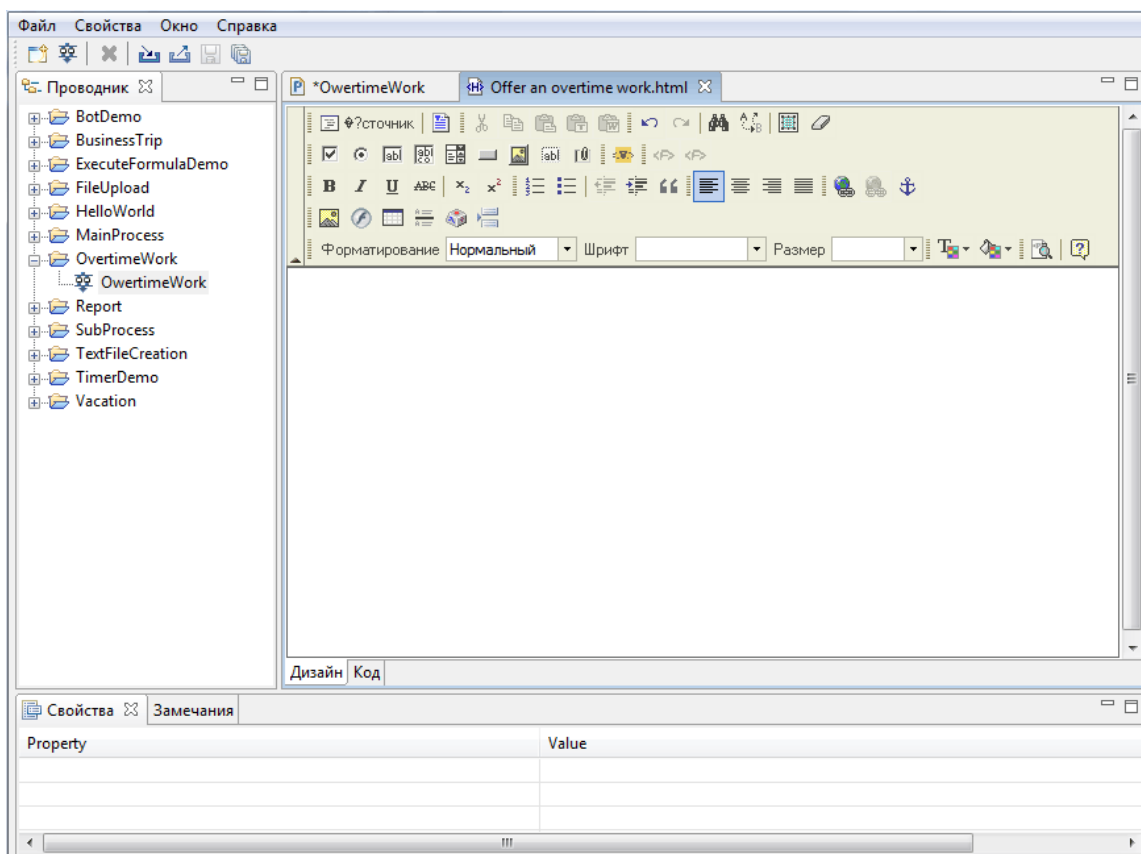
Кликните на стартовый узел «Offer an overtime work» графа бизнес-процесса правой кнопкой мыши и выберите команду «Форма» > «Создать форму» (левой кнопкой мыши):




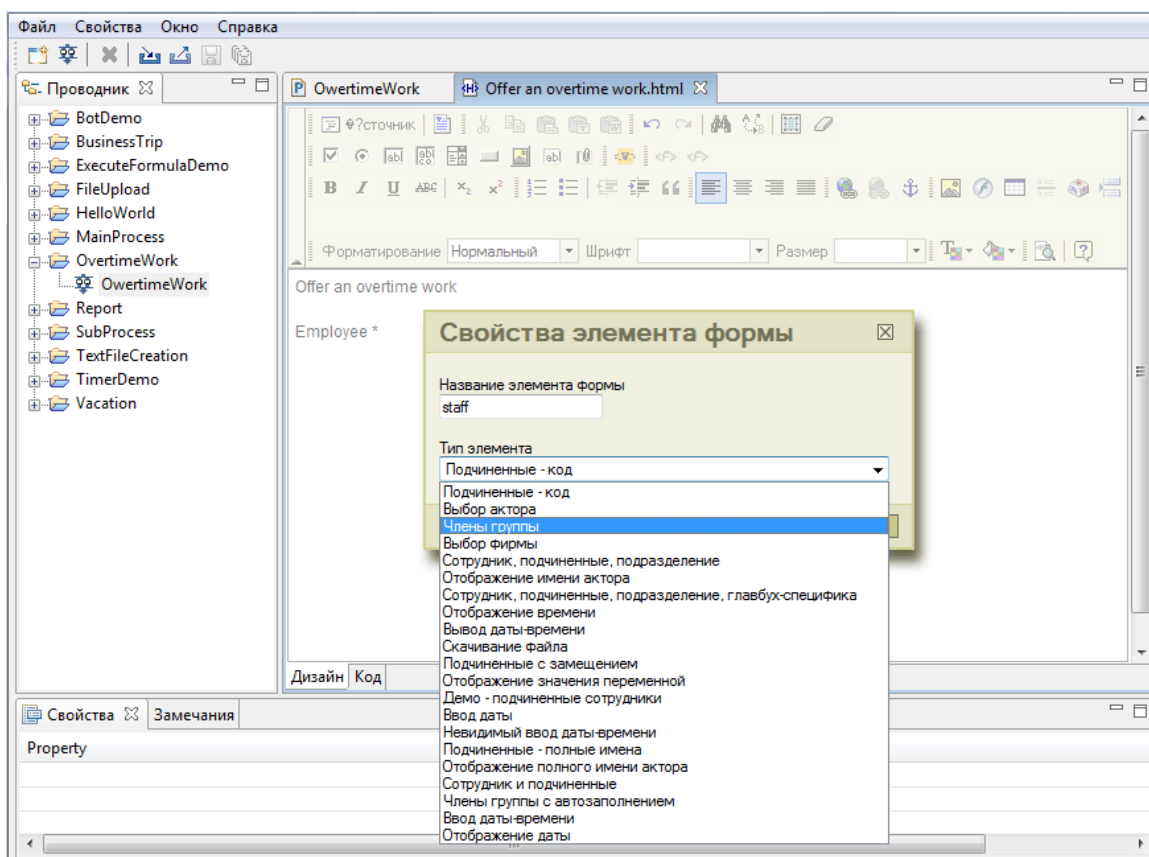
Выберите тип формы «HTML форма + custom теги»:



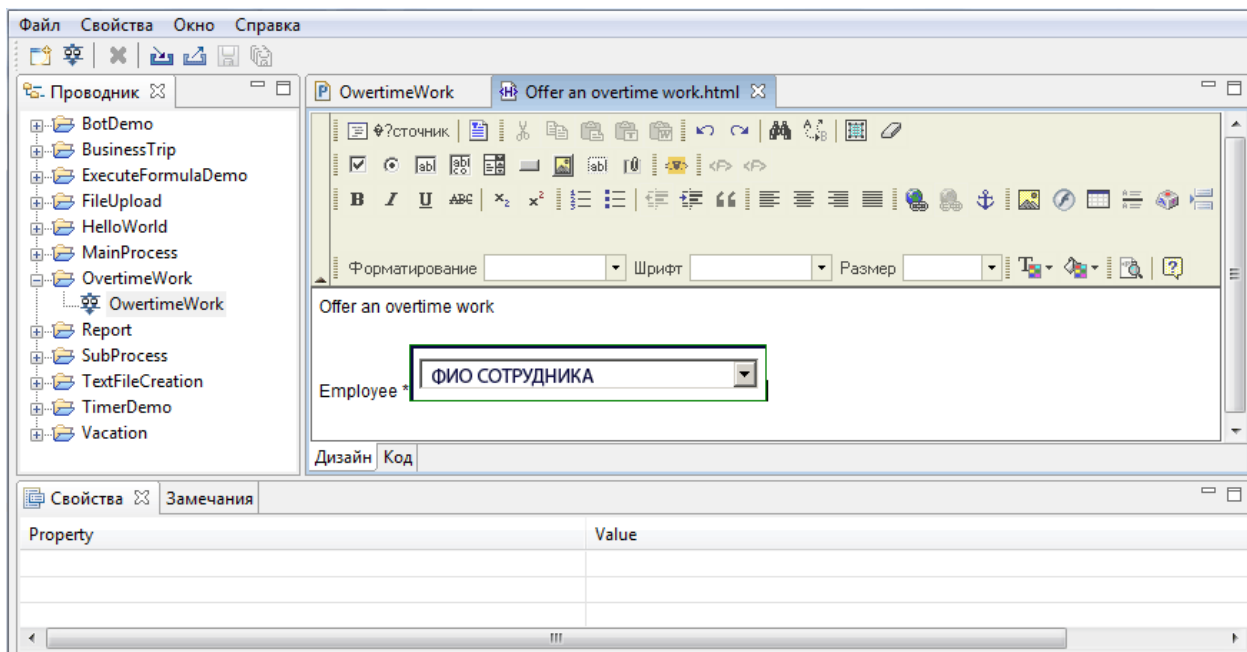
Появится следующий интерфейс:



Введите текст «**Offer an overtime work**», на следующей строке введите «Employee *» и кликните на элемент . В появившейся форме в качестве названия элемента формы введите «staff», в качестве элемента формы выберите «Члены группы»:

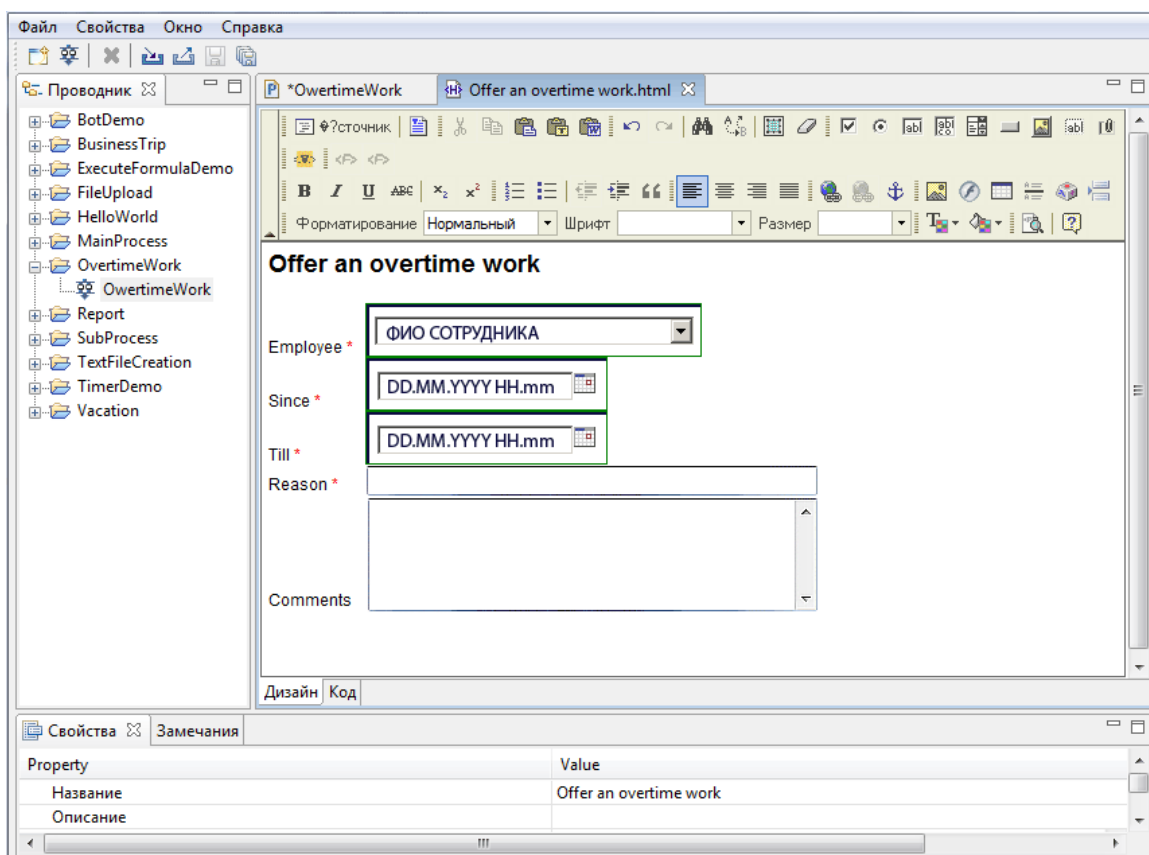


Кликните на «ОК». На форме появится элемент «ФИО СОТРУДНИКА»:

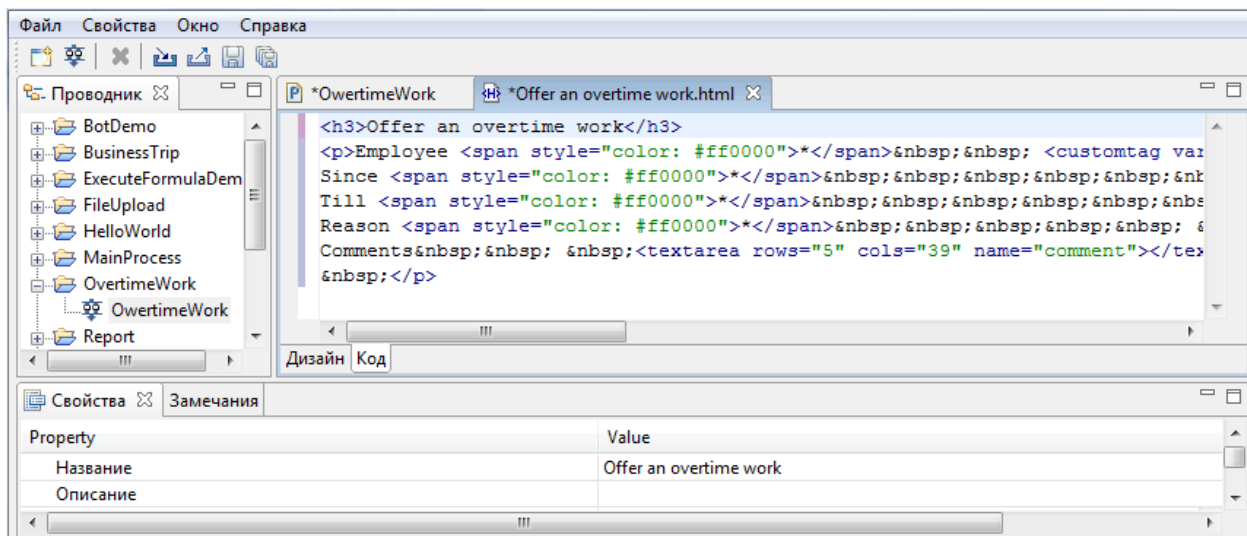


Аналогичным образом создайте элементы для ввода других переменных бизнес-процесса:

- since
- till
- reason
- comment



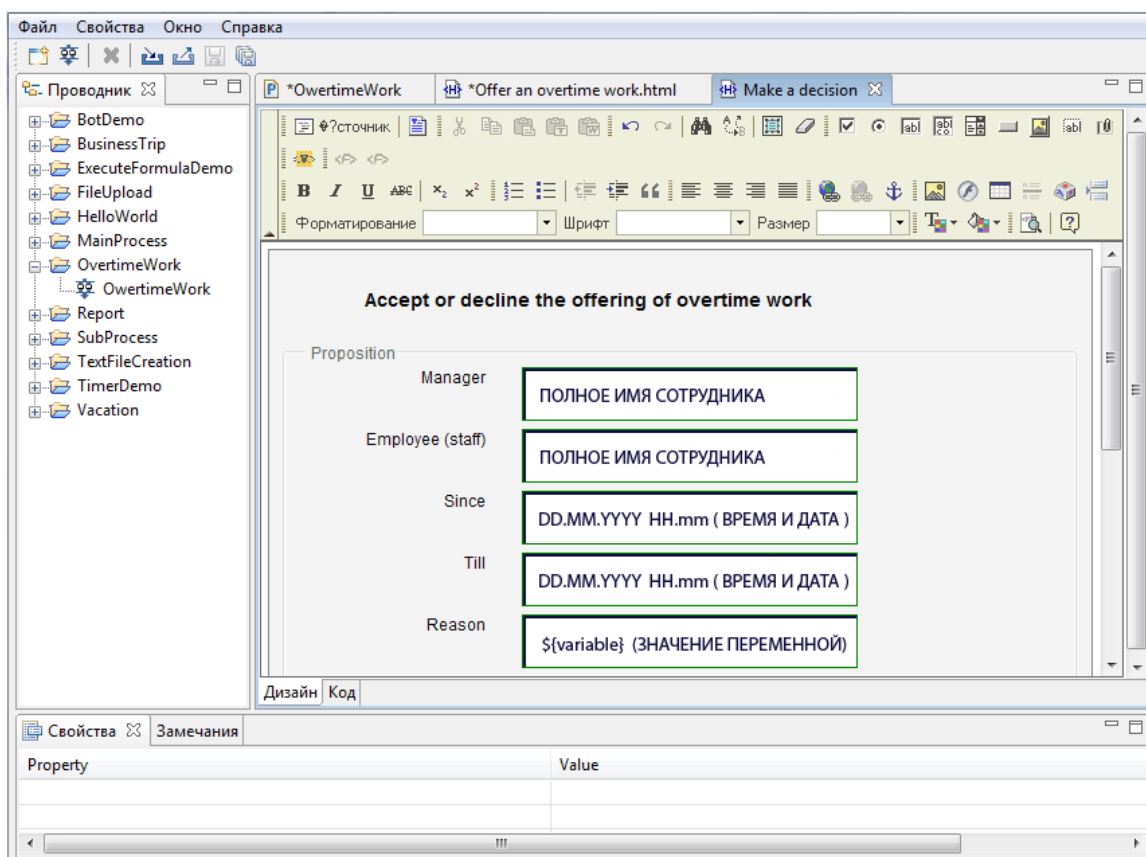
Замечание. Перейдя на вкладку «Код», можно увидеть страницу в виде HTML:



Аналогично создаются остальные формы для остальных узлов-действий

- Make a decision
- Notify of rejection
- Notify of acceptance

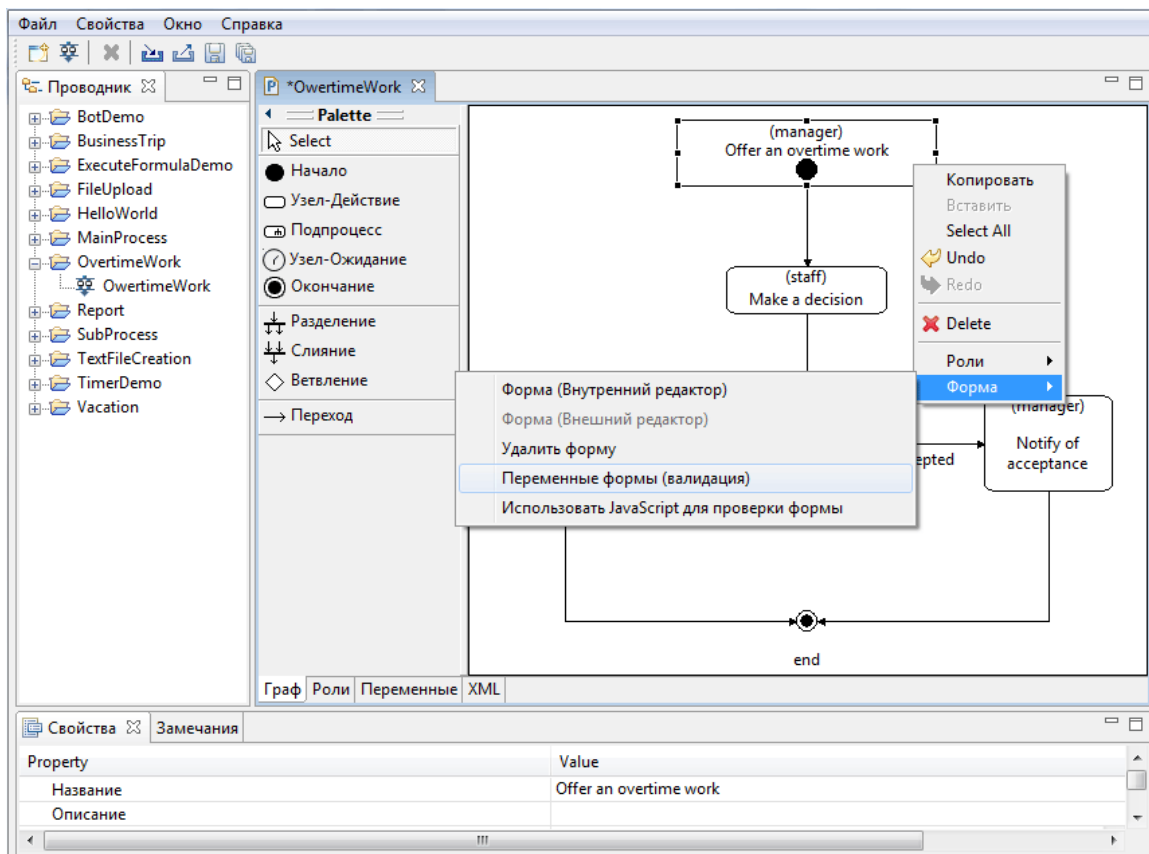
Форма для узла «Make a decision»:



Задание проверок значений, введенных в элементы форм (валидация)

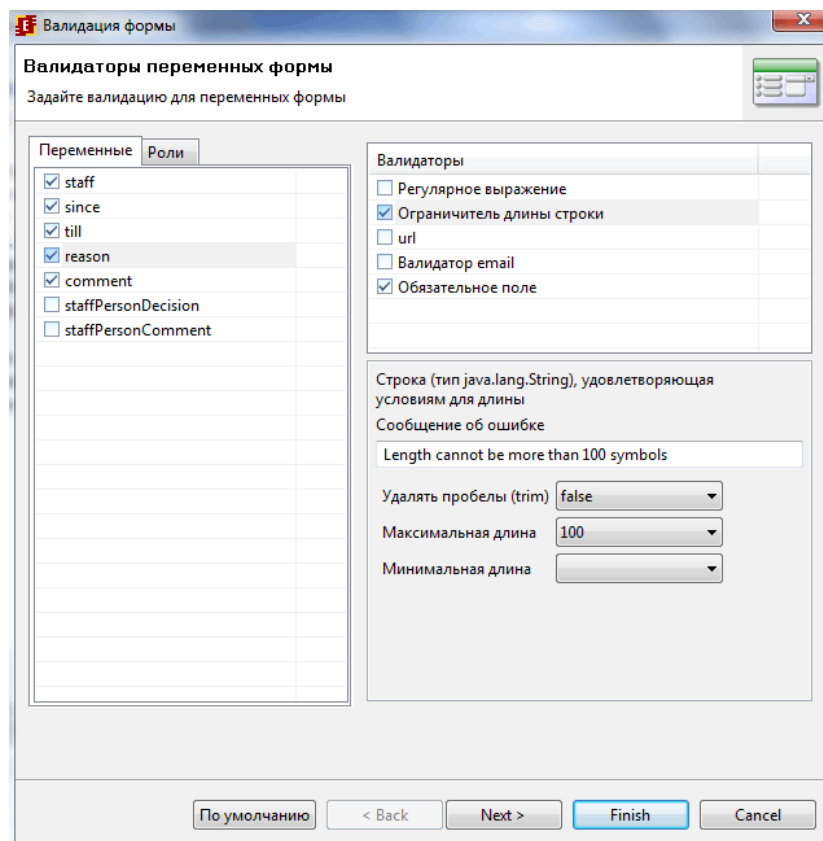
Элементы форм могут быть обязательными или не обязательными для ввода. Также могут быть правила, относящиеся к нескольким элементам форм. Например - «дата с» не может быть позже, чем «дата по». Кроме того, могут быть специфические для типов ограничения: для чисел — меньше или больше какого-то числа, для строк — ограничение на длину строки и т.д.

Для задания проверки значения полей формы «Offer an overtime work» кликните на узле правой кнопкой мыши, в появившейся форме выберите «Переменные формы (валидация)»:



В появившейся форме будут отображены все использующиеся в бизнес-процессе переменные. Все использующиеся в текущей форме переменные будут помечены галочкой.

Для выбранного поля в окне «Валидаторы» будут находиться доступные типы проверок поля. При клике на конкретную проверку в нижнем окне можно будет посмотреть и отредактировать параметры этой проверки:



Например, в данном случае для проверки «ограничитель длины строки» установлена максимальная длина 100 символов, в качестве сообщения об ошибке использовано «Длина не может превышать 100 символов»

Если требуется установить проверки на совместные значения нескольких полей, после установки всех проверок на одно поле кликните «Next».

В появившейся форме при помощи «мастера» можно установить совместные проверки значений полей. Например, что Дата Till не должна быть позже даты since:

Замечание. Если в правилах валидации (проверки формы) найдены ошибки - то процесс не экспортируется. Если в правилах валидации есть только замечания — то процесс экспортируется.

Типы ошибок при работе с формами:

- отсутствие файла валидации при наличии формы - error
- отсутствие переменной формы в переменных процесса - error
- отсутствие переменной формы в файле валидации - warning
- отсутствие переменной валидации в форме - warning

Замечание: на формы типа "Infopath" правила валидации не распространяются.

Создание файла-архива бизнес-процесса и загрузка его в систему

Последовательность действий полностью повторяет последовательность, описанную для процесса разработки бизнес-процесса «HelloWorldProcess».

Замечание. Для выполнения процесса «Overtime Work» необходимо создать в Workflow системе группы пользователей:

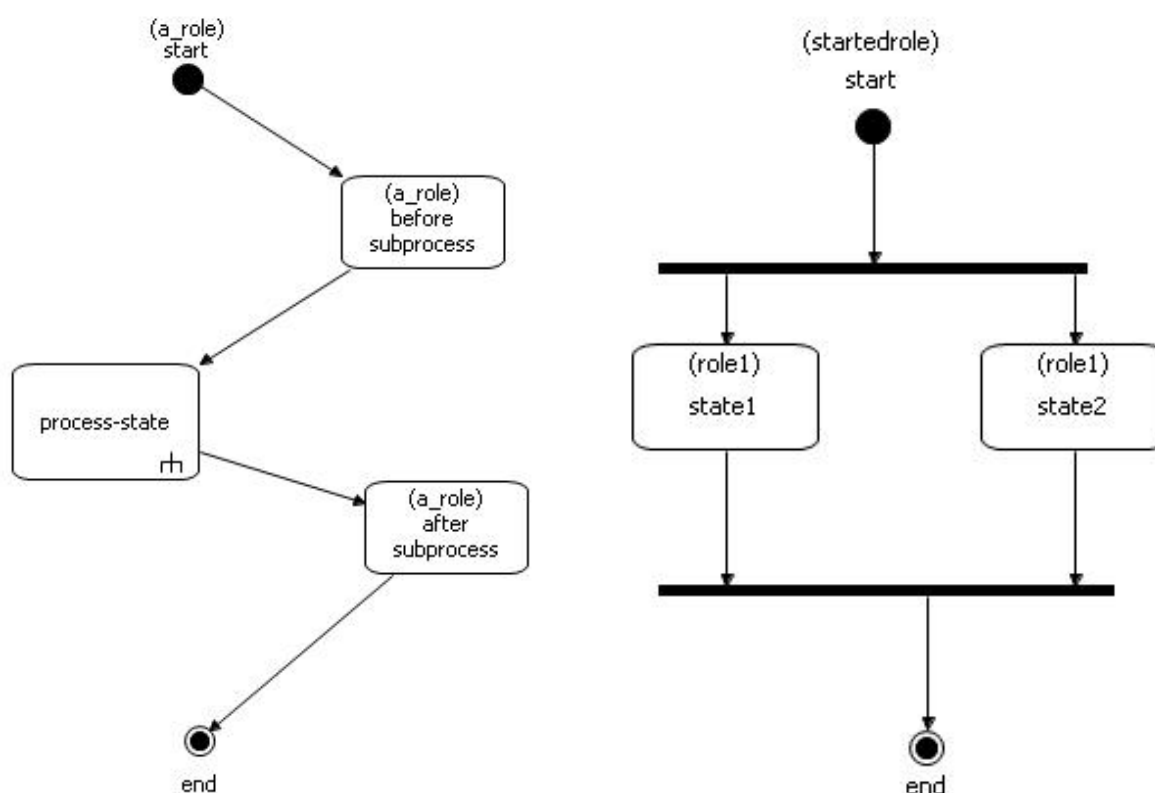
- manager
- staff
- all

завести конкретных пользователей этих групп и раздать соответствующие права (см. руководство пользователя системы RunaWFE).

Создание процесса с подпроцессом

Демо-процесс MainProcess и его подпроцесс SubProcess созданы для иллюстрации работы с подпроцессом.

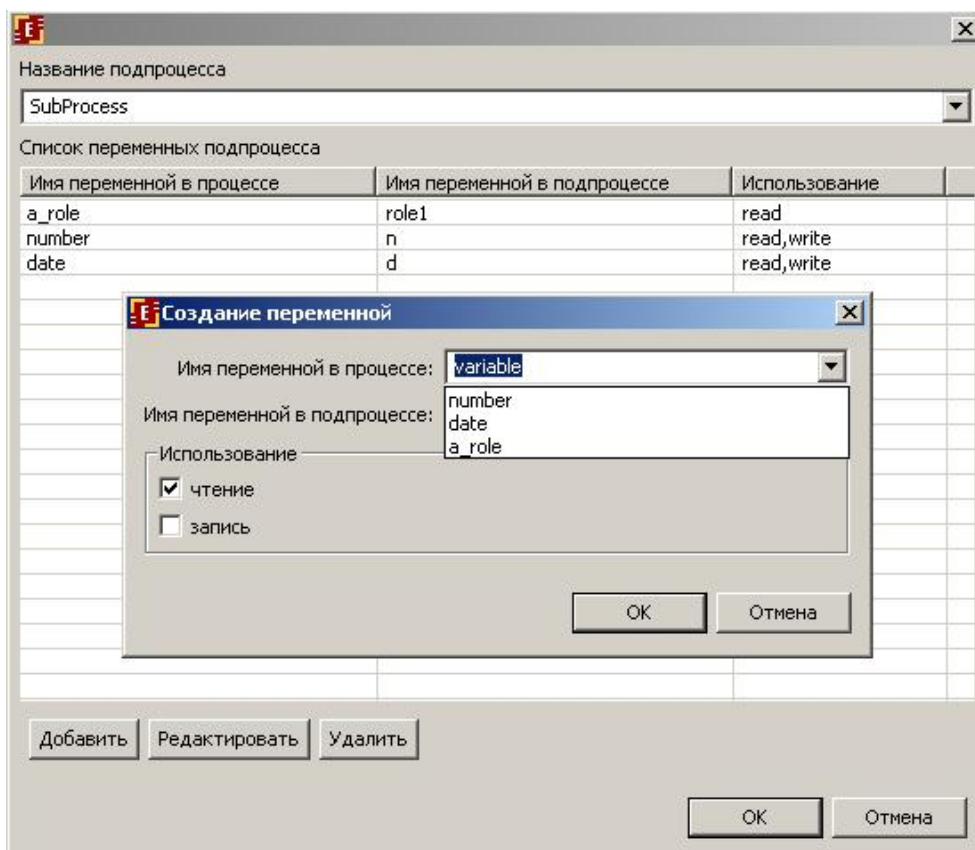
Графы процессов (MainProcess слева, SubProcess справа):



На графе основного процесса присутствует узел “process-state”, отмеченный специальным символом в правом нижнем углу прямоугольника узла. Это отображения узла-подпроцесса. Для создания такого узла на графе используйте элемент «подпроцесс» из палитры.

Для настройки связи между процессом и подпроцессом, сначала рекомендуется создать подпроцесс, чтобы иметь готовый список переменных подпроцесса. Сам подпроцесс ничем не отличается от обычного процесса. Это может быть как независимый процесс, который можно запускать сам по себе, так и процесс, который имеет смысл только в рамках родительского процесса. Один и тот же процесс может быть подпроцессом в нескольких разных родительских процессах.

Когда управление доходит до узла подпроцесса, то запускается новый экземпляр подпроцесса, но стартовая форма подпроцесса не показывается. Вместо этого в подпроцесс передаются значения переменных родительского процесса. Для настройки передачи переменных кликните правой кнопкой мышки на узле подпроцесса и выберите опцию «подпроцесс» из выпадающего меню. Откроется диалог настройки подпроцесса:



Сначала необходимо выбрать название подпроцесса из списка всех процессов из всех проектов, открытых в редакторе. Затем нужно задать соответствия между переменными родительского процесса и подпроцесса, а так же обозначить использование. В качестве передаваемых параметров могут быть и переменные и роли.

Если выбрано использование для чтения (read), то значение переменной родительского процесса будет записано в переменную подпроцесса. Если выбрано запись (write), то после завершения подпроцесса значение из переменной подпроцесса будет записано в переменную родительского процесса.

После того, как экземпляр процесса MainProcess прошел узел-подпроцесс, для перехода к свойствам экземпляра подпроцесса SubProcess можно кликнуть по узлу-подпроцесса левой кнопкой мышки. Для того, чтобы вернуться к свойствам экземпляра родительского процесса MainProcess из SubProcess, достаточно кликнуть на имя родительского процесса в списке свойств процесса.



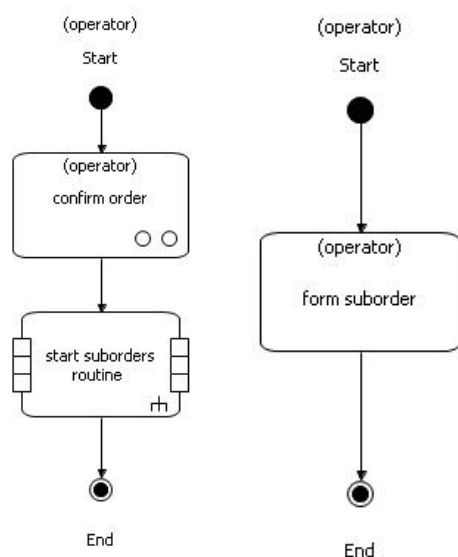
Menu	Process Instance												
Task List													
Start process													
Started processes													
Executors													
System													
SOURCEFORGE.NET													
Feedback email or forum													
	<table> <tr> <td>Name</td><td>SubProcess</td></tr> <tr> <td>Id</td><td>4530</td></tr> <tr> <td>Version</td><td>2</td></tr> <tr> <td>Started</td><td>24.08.2010 17:12</td></tr> <tr> <td>Ended</td><td>24.08.2010 17:12</td></tr> <tr> <td>Parent Process</td><td>MainProcess</td></tr> </table>	Name	SubProcess	Id	4530	Version	2	Started	24.08.2010 17:12	Ended	24.08.2010 17:12	Parent Process	MainProcess
Name	SubProcess												
Id	4530												
Version	2												
Started	24.08.2010 17:12												
Ended	24.08.2010 17:12												
Parent Process	MainProcess												

В одном процессе может быть несколько подпроцессов. В подпроцессе могут быть свои подпроцессы.

Создание процесса с мультидействием

Демо-процесс MultiInstanceProcess и его подпроцесс MultiInstanceSubProcess созданны для иллюстрации работы с мультидействием.

Графы процессов (MultiInstanceProcess слева, MultiInstanceSubProcess справа):

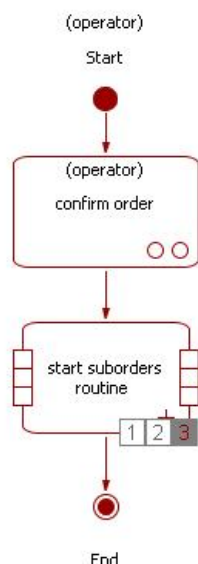


Когда процесс MultiInstanceProcess доходит до узла-мультидействия, то запускается несколько подпроцессов MultiInstanceSubProcess. Количество запускаемых подпроцессов определяется динамически по количеству элементов массива в переменной мультидействия. Переменная мультидействия может иметь любое имя, но ее тип должен быть StringArrayFormat.

Связь между родительским процессом MultiInstanceProcess и подпроцессом MultiInstanceSubProcess настраивается так же как и в случае обычного подпроцесса. Единственная разница — это обязательное присутствие в списке соответствий переменных переменной мультидействия. В соответствие переменной мультидействия родительского процесса (имеющей тип массив) ставится переменная подпроцесса (обычного типа, не массив) в которую будет записано значение элемента массива при создании подпроцесса. У каждого подпроцесса значение будет своё. Если при настройке выбран тип использования «write», то после завершения подпроцессов значения соответствующих переменных будут записаны в массив переменной мультидействия.

Process variable name	Subprocess variable name	Usage
operator	operator	read
addressArray	address	read, multiinstancelink

В данном примере addressArray - переменная мультидействия. В подпроцессе ей соответствует address. При запуске подпроцессов в узле «start suborder routine» будет запущено столько процессов, сколько адресов в массиве addressArray. В каждом процессе будет свое значение переменной address. Для перехода к свойствам экземпляра подпроцесса мультидействия кликните на квадратике с цифрой на графе экземпляра родительского процесса. Количество квадратиков совпадает с количеством подпроцессов мультидействия.



Возвращение к свойствам родительского экземпляра такое же, как и для обычных подпроцессов.

Работа с обработчиками (ActionHandler)

В системе RunaWFE есть возможность вызвать выполнение java-кода при наступлении определенных событий в бизнес-процессе, например, проход точки управления по определенному переходу. Java код должен быть реализован в методе execute() класса, реализующего интерфейс ActionHandler. Этот класс должен быть загружен в систему. (См. «Как добавить в редактор имена классов для инициализаторов ролей-дорожек, форматов переменных, обработчиков для узлов-выбора и элементов ActionHandler.»)

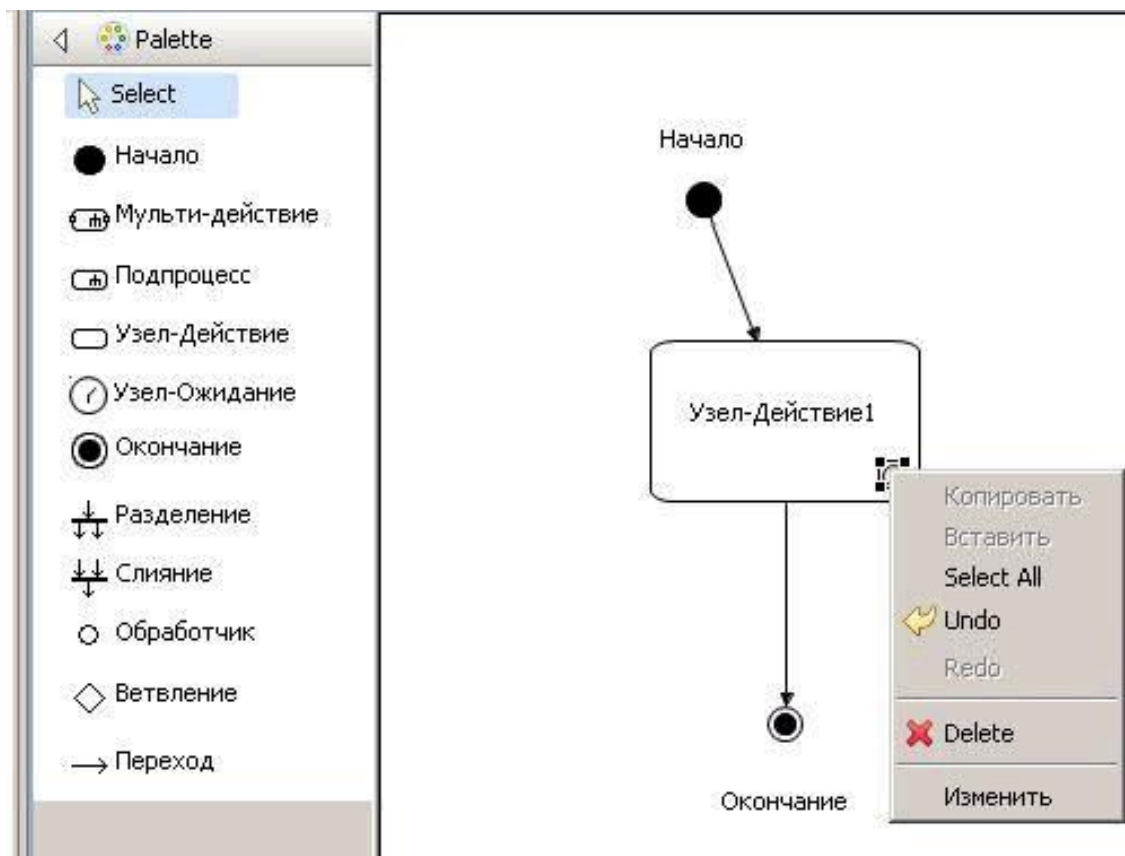
Для привязывания к переходу класса, реализующего интерфейс ActionHandler в редакторе бизнес-процесов надо проделать следующее:

1. Убедиться, что в меню «свойства» выбрана опция «показать обработчики».
2. Выбрать «обработчик» в палитре. К стрелке курсора добавится маленькая иконка.
3. Наведите стрелку на нужный переход и нажмите левую кнопку мышки. На графе на линии перехода появится маленький кружок, отображающий обработчик.

4. Выбрать в палитре элемент Select. Кликнуть на созданный обработчик. В свойствах созданного действия выбрать класс соответствующий ActionHandler-класс и указать (если необходимо) конфигурацию.

Как удалить обработчик

Если вы хотите удалить обработчик, выделите его на графе и нажмите delete.



BSHActionHandler

Данный action handler используется для пересчета значений переменных БП.

Конфигурацией является корректный код BeanShell (www.beanshell.org ^[5]),

который по синтаксису очень похож на Java и из которого можно делать вызов в Java.

Пример конфигураций:

```
My_date = new java.util.Date();
My_rnd = new java.util.Random(1000).nextInt();
My_time = java.lang.System.currentTimeMillis();
int n = Integer.parseInt(multNumber);
String[] array = new String[n];
for (int i=0; i < n; i++){
```

```
    array[i] = "book " + i;
```

```
};
```

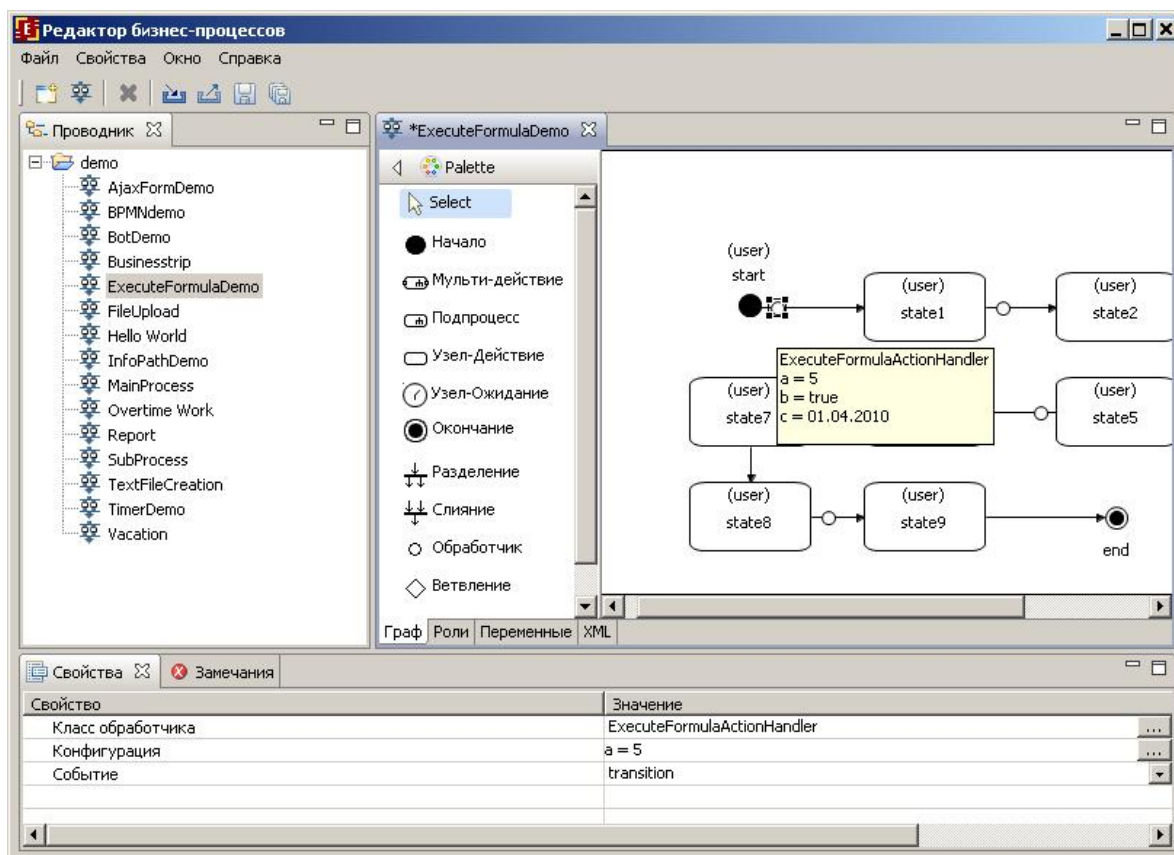
```
multArray = array;
```

Измененные в процессе исполнения скрипта переменные изменят свое значение в БП.

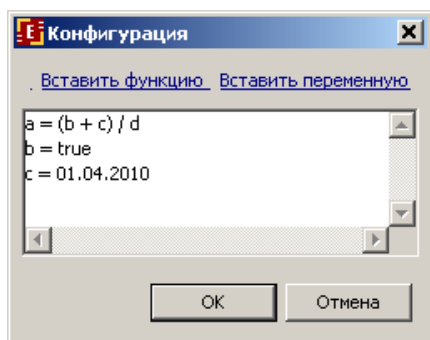
Замечание: будут изменены лишь переменные, которые до этого уже существовали в БП, создание новых не поддерживается.

ExecuteFormulaActionHandler

ExecuteFormulaActionHandler – универсальный ActionHandler, предназначенный для изменения значений переменных бизнес процесса. Для редактирования формул надо кликнуть на кнопке в строке «конфигурация» в свойствах обработчика:



Появится окно, в котором надо задать формулы вида : 'имя модифицируемой переменной' = выражение, например $a = (b + c) * d$



В выражении могут использоваться такие операции: +, -, *, /, <, <=, ==, !=, >, >=, &, |, ^.

Наибольший приоритет (4) у умножения и деления.

Приоритет 3 у операций сложения и вычитания.

Приоритет 2 у сравнений.

Наименьшим приоритетом обладают булевы операции «&» (and), «|» (or) и «^» (xor).

Если имя используемой в выражении переменной содержит пробелы, то его нужно заключить в одинарные кавычки.

Помимо переменных могут использоваться непосредственные значения типов Long, Double, Boolean, String, Date и функции.

Функции могут быть такими: date(дата), time(дата) и hours_round_up(число)

date(дата) округляет дату до целого числа дней, отбрасывая время.

time(дата) отбрасывает дни, оставляя только время.

hours_round_up(число) округляет время, заданное в минутах, до часов в верхнюю сторону.

round(число) или round(число, количество знаков после запятой) округляет число, оставляя фиксированное количество знаков после запятой.

round_up(число) или round_up(число, количество знаков после запятой) округляет число в верхнюю сторону.

round_down(число) или round_down(число, количество знаков после запятой) округляет число в нижнюю сторону.

number_to_string_ru(число) переводит число в строку на русском языке.

number_to_string_ru(число, п2, п3, п4, п5) переводит число с зависимым словом в строку на русском языке.

П2 — 'М' или 'F' — мужской/женский род зависимого слова.

П3, п4, п5 — зависимое слово в трех формах (например: день, дня, дней).

number_to_short_string_ru(число, п2, п3, п4, п5) выбирает соответствующую числу форму зависимого слова.

get_instance_id() - возвращает номер текущего экземпляра процесса

Например:

variable1 = 2

variable2 = 3.1415926535

variable3 = true

variable4 = «string»

variable5 = «\» (для использования символа кавычки требуется поставить перед ней '\')

variable6 = 01.09.2008

variable7 = '01.09.08 18:00' (здесь требуется заключить дату в одинарные кавычки, так как она содержит пробелы)

variable8 = round_up(2.564367, 3) (*variable8* = 2.565)

variable9 = number_to_string_ru(102, «F», «спичка», «спички», «спичек»)

(*variable9* = «сто две спички»)

"текущая дата" = current_date()

"время" = current_time()

"дата и время" = current_date_time()

Функция для работы с ФИО:

Функция FIO_case_ru(String fio, int caseNumber, String mode)fio - фамилия, имя и отчество в именительном падеже через пробел

caseNumber - номер падежа (от 1 до 6)

mode - строка форматирования

Символы F, N и P заменяются на фамилию, имя или отчество в соответствующем падеже.

Символы f, n и r заменяются на первую букву фамилии, имени или отчества. Остальные символы строки форматирования не меняются.

Примеры использования:

FIO_case_ru("Иванов Иван Иванович", 1, "F N P :)") Иванов Иван Иванович :)

FIO_case_ru("Иванова Ира Ивановна", 2, "F N p") Ивановой Иры И.

FIO_case_ru("Иванов Иван Иванович", 3, "F n P") Иванову И. Ивановичу

FIO_case_ru("Иванова Ира Ивановна", 4, "F n p") Иванову И. И.

FIO_case_ru("Иванов Иван Иванович", 5, "f N P") И. Иваным Ивановичем

FIO_case_ru("Иванова Ира Ивановна", 6, "o P n") о Ивановне И.

FIO_case_ru("Апырвапыркина Фыва Няковна", 5, "F N P") Апырвапыркиной Фывой Няковной

Если функция неправильно склоняет какие-либо имена и фамилии, можно добавить исключение в конфигурационный файл JBOSS_HOME/server/default/conf/ActionHandlers/nameCaseConf.xml

Формат файла:

```
<?xml version="1.0">
```

```
<nameCaseConf>
```

```
  <family value = "фамилия в именительном падеже">
```

```
    <rule case = "номер падежа">правильный вариант фамилии</rule>
```

```
    <rule case = "номер падежа">правильный вариант фамилии</rule>
```

```
    ...
```

```
  </family>
```

```
  <family value = "фамилия в именительном падеже">
```

```
    ...
```

```
  </family>
```

```
  <name value = "имя в именительном падеже">
```

```
    <rule case = "номер падежа">правильный вариант имени</rule>
```

```
    ...
```

```
  </name>
```

```
  <name value = "отчество в именительном падеже">
```

```
    <rule case = "номер падежа">правильный вариант отчества</rule>
```

```
    ...
```

```
  </name>
```

```
  ...
```

```
</nameCaseConf>
```

Пример файла конфигурации:

```
<?xml version="1.0" encoding="UTF-8"?> <nameCaseConf>
```

```
  <!-- <name value="Николай">
```

```
  </name> -->
```

```
  <family value="Кримза">
```

```
    <rule case="2">Кримзы</rule>
```

```
    <rule case="3">Кримзе</rule>
```



```

<rule case="4">Кримзу</rule>
<rule case="5">Кримзой</rule>
<rule case="6">Кримзе</rule>
</family>
<!-- <parent value="Петрович">
</parent> -->
</nameCaseConf>

```

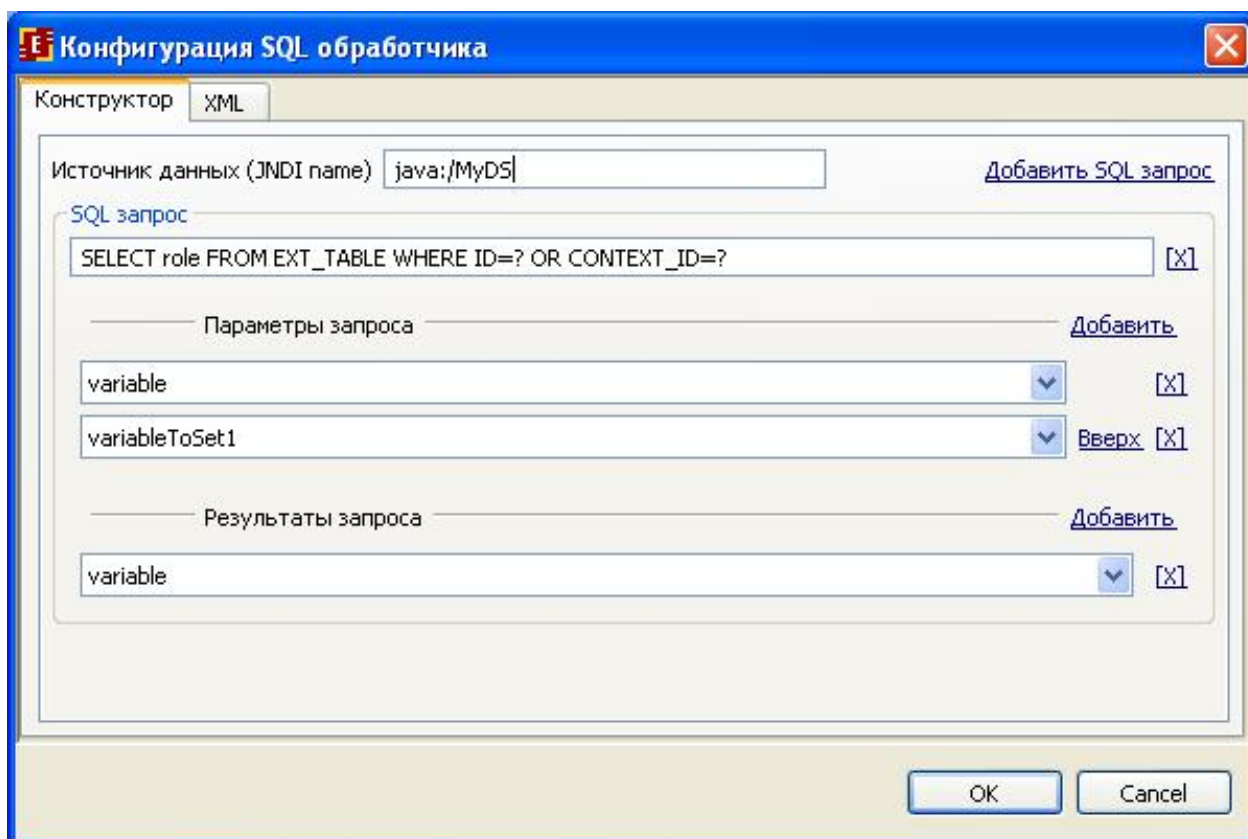
SQLActionHandler

Используется для взаимодействия с БД.

С помощью него можно выполнить запрос, выполняющий чтение из БД в переменные БП или наоборот, запись в БД значений переменных БП.

В одном обработчике возможно выполнить несколько запросов

Диалог конфигурации для данного обработчика выглядит следующим образом (чтобы он открылся — следует создать обработчик и выбрать класс обработчика):



В нем:

Источник данных (JNDI Name) — имя источника данных, зарегистрированного на сервере.

Для jboss, как правило нужно писать префикс java:/

SQL запрос — собственно сам запрос (в формате java.sql.PreparedStatement).

Его синтаксис включает текст запроса которым вместо знаками ? Отмечаются параметры, количество которых должно соответствовать числу параметров в секции Параметры запроса.

Если запрос типа SELECT, то он должен явно перечислять столбцы, значения которых будут присвоены переменным, перечисленным в секции Результаты запроса.

AssignSwimlaneActionHandler

Позволяет (пере)определить роль в любом месте процесса по заданной конфигурации. Значение инициализатора имеет такой же синтаксис как и на вкладке редактирования ролей.

SendEmailActionHandler

Позволяет послать по e-mail форму задания. Этот обработчик будет корректно работать только если размещен на узле задачи. Телом сообщения является форма задания. Конфигурация обработчика в редакторе содержит название файла на сервере в папке ...server/default/conf/ActionHandlers. Название надо указать без расширения .properties, которое должно быть у файла, в котором должны заданы параметры сообщения.

Пример файла настройки:

```
smtp.server=111.16.100.5
```

```
from=user@aaa.bbb.ru
```

```
#subject=
```

```
to=<customtag var="__currentTokenExecutor__" delegation="ru.runa.wf.web.html.vartag.UsersEmailVarTag" />
```

```
#smtp.user=
```

```
#smtp.password=
```

```
WFUser=Administrator
```

```
WFUserPass=wf
```

```
#authReq=false
```

```
smtp.SendNotification=false
```

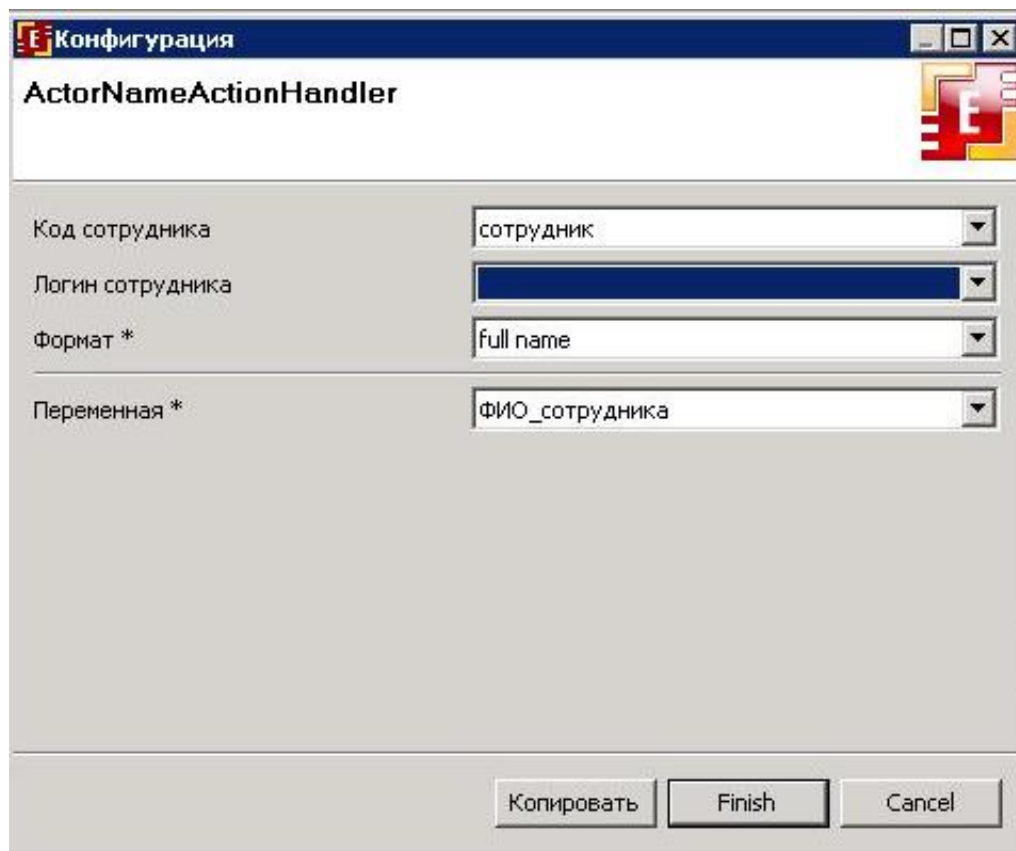
```
#content.type=text/html
```

```
content.type=text/plain
```

```
#email.template=/emailTemplates/emailTemplate.xml
```

ActorNameActionHandler

Позволяет добавить в переменные процесса логин или полное имя сотрудника по коду или по логину. Для настройки параметров обработчика используется форма.



Необходимо указать переменную с кодом сотрудника или переменную с логином сотрудника в качестве входящих данных для обработчика. Затем выбрать формат исходящих данных:

- name - логин сотрудника
- full name - полное имя сотрудника

И указать название переменной, в которую будет записан результат работы обработчика.

Как добавить в редактор имена классов для инициализаторов ролей-дорожек, форматов переменных, обработчиков для узлов-выбора и элементов ActionHandler.

Поместите .jar-файлы, содержащие соответствующие классы в подкаталог /plugins/org.jbpm.core_3.0.1/lib каталога, содержащего редактор бизнес-процессов. После этого перезапустите RunaGPD.

Замечание 1: если .jar-файл имеет зависимости на сторонние библиотеки, то эти библиотеки необходимо положить туда же.

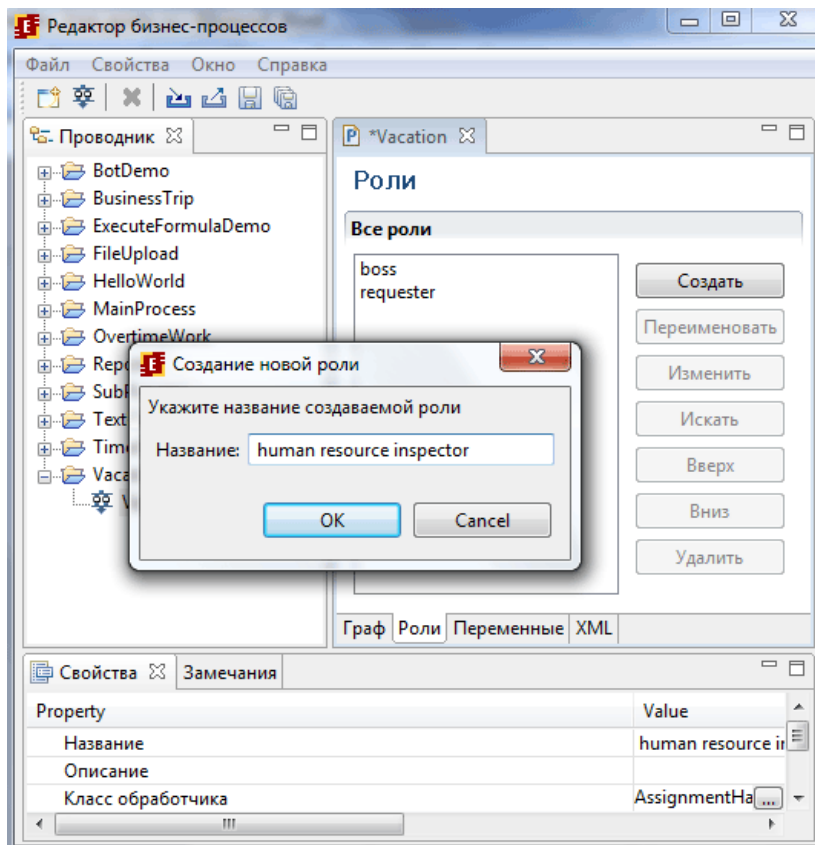
Замечание 2: иногда из-за кеша платформы eclipse новые элементы не появляются. Для этого при выключенном GPD необходимо удалить папку {GPD}/workspace/.metadata/.plugins/org.eclipse.jdt.core и запустить GPD снова.

Задание инициализаторов для ролей-дорожек.

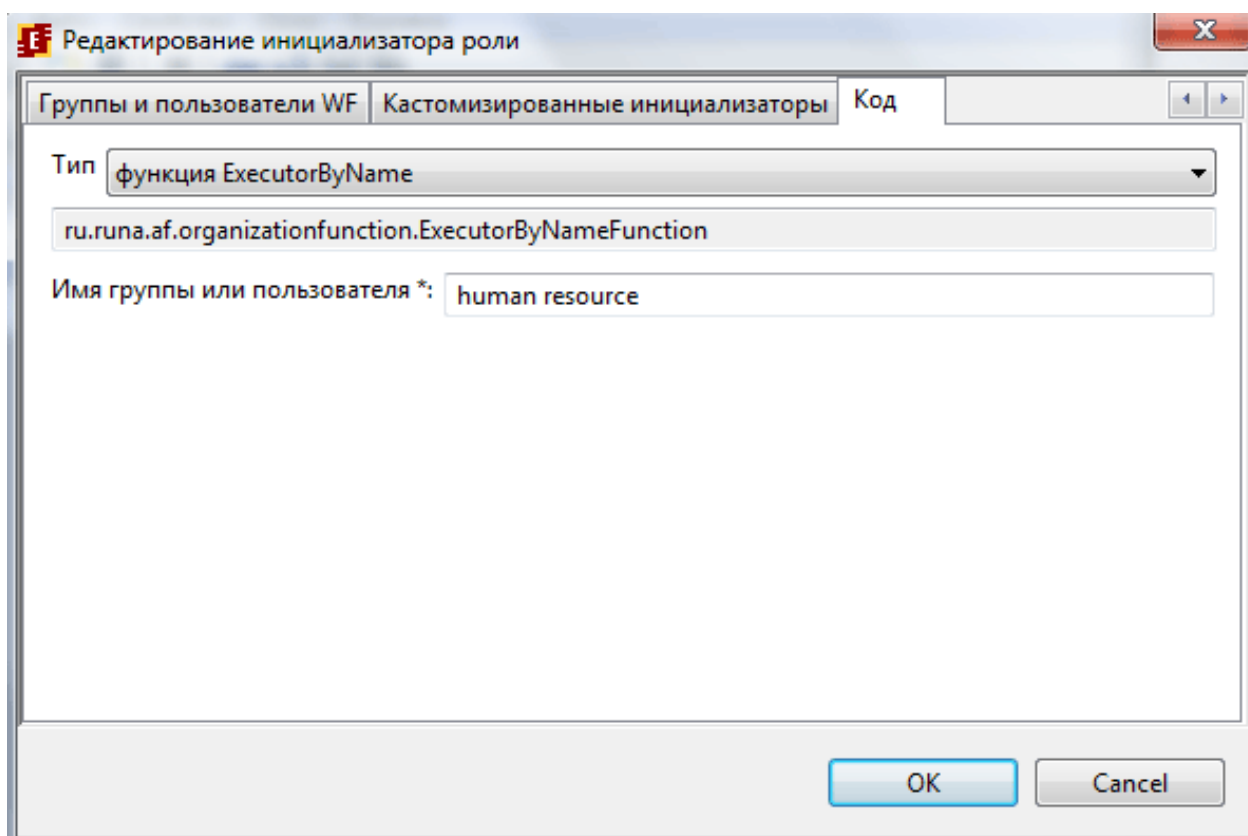
Первый пример

В качестве примера задания инициализатора при создании новой роли рассмотрим создание роли «human resource inspector» для демо-процесса «Vacation».

На закладке «Роли» кликните «Создать». В появившейся форме введите «human resource inspector» в качестве названия роли-дорожки. Кликните «ОК».



Кликните по роли, нажмите кнопку «Изменить». В появившемся окне на вкладке «Код» выберите инициализатор «функция ExecutorByName» и в качестве параметра введите имя группы, содержащей сотрудников – инспекторов кадровой службы.



Кликните «OK». Инициализатор новой роли при этом будет определен.

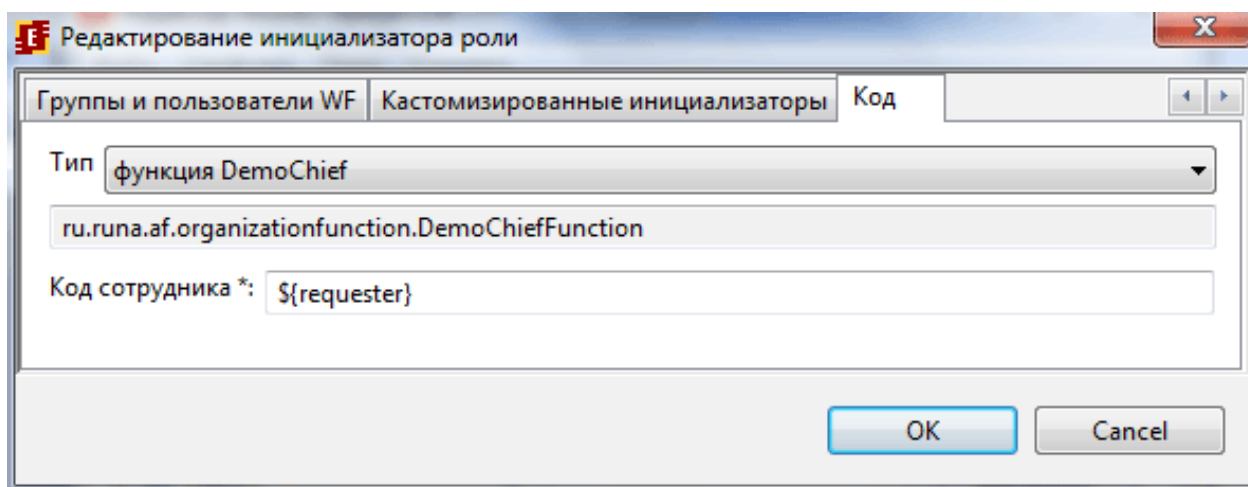
Второй пример

Приведем пример задания инициализатора роли-дорожки, параметром которого является значение переменной бизнес-процесса.

Рассмотрим создание роли «boss» (руководитель) для демо-процесса «Vacation».

На закладке «Роли» кликните «Создать». В появившейся форме введите «boss» в качестве названия роли-дорожки. Кликните «OK».

Кликните по роли, нажмите кнопку «Изменить». В появившемся окне на вкладке «Код» выберите инициализатор DemoChiefFunction (руководитель сотрудника в случае демо-процессов). В качестве параметра введите строку «\${requester}» - значение переменной requester.



Кликните «OK». Инициализатор новой роли при этом будет определен.

Замечание. Для задания в качестве параметра значения переменной надо ввести в окно ввода параметра выражение: \${имя_переменной}.

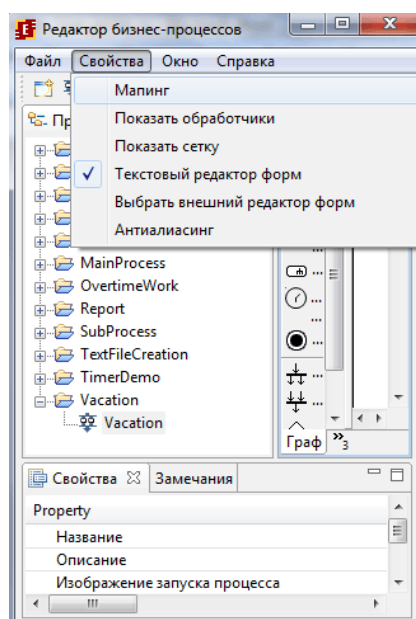
Замещение имен java-классов русскоязычными названиями.

В системе RunaWFE и редакторе процессов можно заменить англоязычные названия java-классов русскоязычными названиями. Для этого надо сделать следующее:

1. Загрузите java-классы в редактор:

Поместите .jar-файлы, содержащие соответствующие классы в подкаталог /plugins/org.jbpm.core_3.0.1/lib каталога, содержащего редактор бизнес-процессов. Запустите редактор бизнес-процессов.

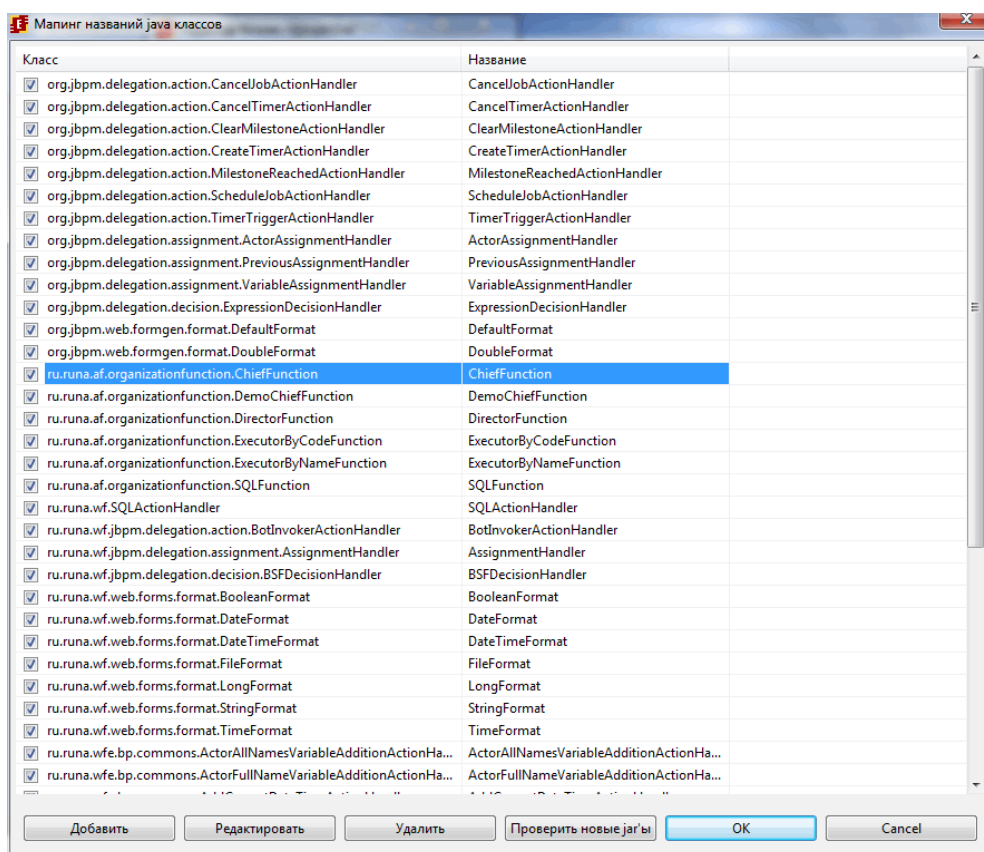
2. Запустите редактор, откройте пункт меню Свойства/Мапинг:



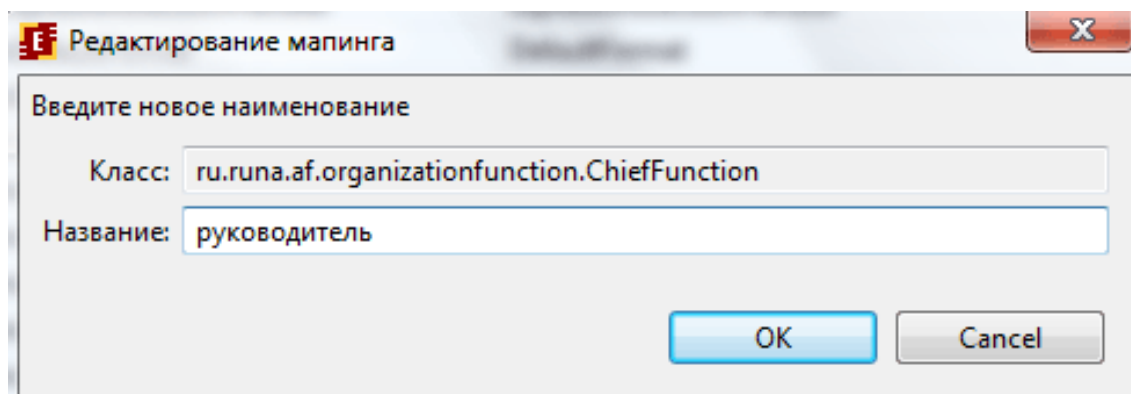
3. В появившейся форме кликните «Проверить новые jar'ы»

4. «Снимите» галочки у тех классов, которые не должны присутствовать в редакторе

5. Для замещения имени класса выделите строку и кликните «Редактировать»



6. В появившейся форме введите название для класса и кликните «OK»



1. Кликните «OK» на форме «Мэпинг названий java классов»

Замечание 1. Файлы, в которых хранится структура замещения, находятся в папке:

`${RunaGPD}\workspace\metadata\plugins\org.jbpm.ui`, рядом с `dialog_settings.xml`

Замечание 2. Файлы, из которых выбираются классы, находятся в папке:

`${RunaGPD}\plugins\org.jbpm.core_3.0.1\lib`

Примечания

[1] <http://sourceforge.net/projects/runawfe/files>

[2] <http://java.sun.com/j2se/1.5.0/download.jsp>

[3] <http://www.eclipse.org/downloads/>

[4] *Замечание: Для того, чтобы загрузить определение бизнес-процесса в систему в вас должны быть права «Загружать определение процесса» (могут быть даны в меню система).*

[5] <http://www.beanshell.org/>