

Лабораторная работа:

Особенности разработки диаграммы деятельности в среде IBM Rational Rose 2003


Диаграмма деятельности в среде IBM Rational Rose 2003, так же как и диаграмма состояний, может относиться к отдельному классу, операции класса, варианту использования, пакету или представлению. Для того чтобы построить диаграмму деятельности, ее вначале необходимо создать и активизировать.









Начать построение диаграммы деятельности для выбранного элемента модели или моделируемой системы в целом можно одним из следующих способов:

- Щелкнуть на кнопке с изображением диаграммы состояний на стандартной панели инструментов, после чего следует выбрать представление и тип разрабатываемой диаграммы - диаграмма деятельности.
- Выделить логическое представление (**Logical View**) или представление вариантов использования (**Use Case View**) в браузере проекта и выполнить операцию контекстного меню: **New** → **Activity Diagram** (Новая → Диаграмма деятельности).
- Раскрыть логическое представление (**Logical View**) в браузере проекта и выделить рассматриваемый класс, операцию класса, пакет, или раскрыть представление вариантов использования (**Use Case View**) и выбрать вариант использования, после чего выполнить операцию контекстного меню: **New** → **Activity Diagram** (Новая → Диаграмма деятельности).
- Выполнить операцию главного меню: **Browse** → **State Machine Diagram** (Обзор → Диаграмма состояний), после следует чего выбрать представление и тип разрабатываемой диаграммы – диаграмма деятельности.

В результате выполнения этих действий появляется новое окно с чистым рабочим листом диаграммы деятельности и специальная панель инструментов, содержащая кнопки с изображением графических элементов, необходимых для разработки диаграммы деятельности (табл. 10.1). Назначение отдельных кнопок панели можно узнать из всплывающих подсказок.

Таблица 10.1. Назначение кнопок специальной панели инструментов диаграммы деятельности

Графическое изображение	Всплывающая подсказка	Назначение кнопки
	Selection Tool	Превращает изображение курсора в форму стрелки для последующего выделения элементов

		на диаграмме
	<i>Text Box</i>	Добавляет на диаграмму текстовую область
	Note	Добавляет на диаграмму примечание
	<i>Anchor Note to Item</i>	Добавляет на диаграмму связь примечания с соответствующим графическим элементом диаграммы
	State	Добавляет на диаграмму состояние
	Activity	Добавляет на диаграмму <i>деятельность</i>
	Start State	Добавляет на диаграмму начальное состояние
	End State	Добавляет на диаграмму конечное состояние
	State Transition	Добавляет на диаграмму <i>переход</i>

	Transition to Self	Добавляет на диаграмму <i>рефлексивный переход</i>
	<i>Horizontal Synchronization</i>	Добавляет на диаграмму горизонтально расположенный символ синхронизации
	<i>Vertical Synchronization</i>	Добавляет на диаграмму вертикально расположенный символ синхронизации
	<i>Decision</i>	Добавляет на диаграмму <i>символ принятия решения</i> для альтернативных переходов
	Swimlane	Добавляет на диаграмму дорожку
	Object	Добавляет на <i>диаграмму объект</i> (по умолчанию отсутствует)
	Object Flow	Добавляет на диаграмму стрелку <i>потока объектов</i> (по умолчанию отсутствует)
	Business Activity	Добавляет на диаграмму бизнес-деятельность (по умолчанию отсутствует)

	Business Transaction	Добавляет на диаграмму бизнес-транзакцию (по умолчанию отсутствует)
---	----------------------	---

Как видно из этой таблицы, *по* умолчанию на панели инструментов отсутствуют некоторые графические элементы, а именно - кнопки с пиктограммами объекта и *потока объектов*. При необходимости их можно добавить на специальную панель диаграммы *деятельности* стандартным способом, который был описан ранее.

Для разрабатываемого проекта системы управления банкоматом *диаграмма деятельности* описывает последовательность действий клиента при использовании банкомата. Для удобства можно включить эту диаграмму в логическое *представление*, для чего необходимо в браузере проекта выделить логическое *представление* (**Logical View**) и выполнить операцию контекстного меню: **New** → **Activity Diagram**(Новая → *Диаграмма деятельности*). Продолжая разработку проекта *по* моделированию системы управления банкоматом, можно приступить к разработке новой диаграммы *деятельности*. С этой целью для диаграммы *деятельности* модели банкомата зададим имя **Диаграмма деятельности АТМ**, а в секцию ее документации введем текст "*Диаграмма деятельности описывает последовательность действий клиента при использовании банкомата*".

Добавление деятельности на диаграмму деятельности и редактирование ее свойств

Для добавления *деятельности* на диаграмму *деятельности* нужно с помощью левой кнопки мыши нажать кнопку с изображением пиктограммы *деятельности* на специальной панели инструментов, отпустить левую кнопку мыши и щелкнуть левой кнопкой мыши на свободном месте *рабочего листа* диаграммы. На диаграмме появится изображение *деятельности* с маркерами изменения его геометрических размеров и предложенным средой именем *по* умолчанию, которое разработчику следует изменить. Добавить *деятельность* на диаграмму можно также с помощью *операции* главного меню: **Tools** → **Create** → **Activity** или с помощью *операции* контекстного меню: **New** → **Activity**, предварительно выделив диаграмму *деятельности* в браузере проекта.

В результате этих действий на диаграмме появится изображение *деятельности* с именем **NewActivity**, предложенное программой *по* умолчанию. Начиная построение диаграммы *деятельности* модели банкомата, для первой добавленной *деятельности* зададим имя Вставить карточку (рис. 10.1).

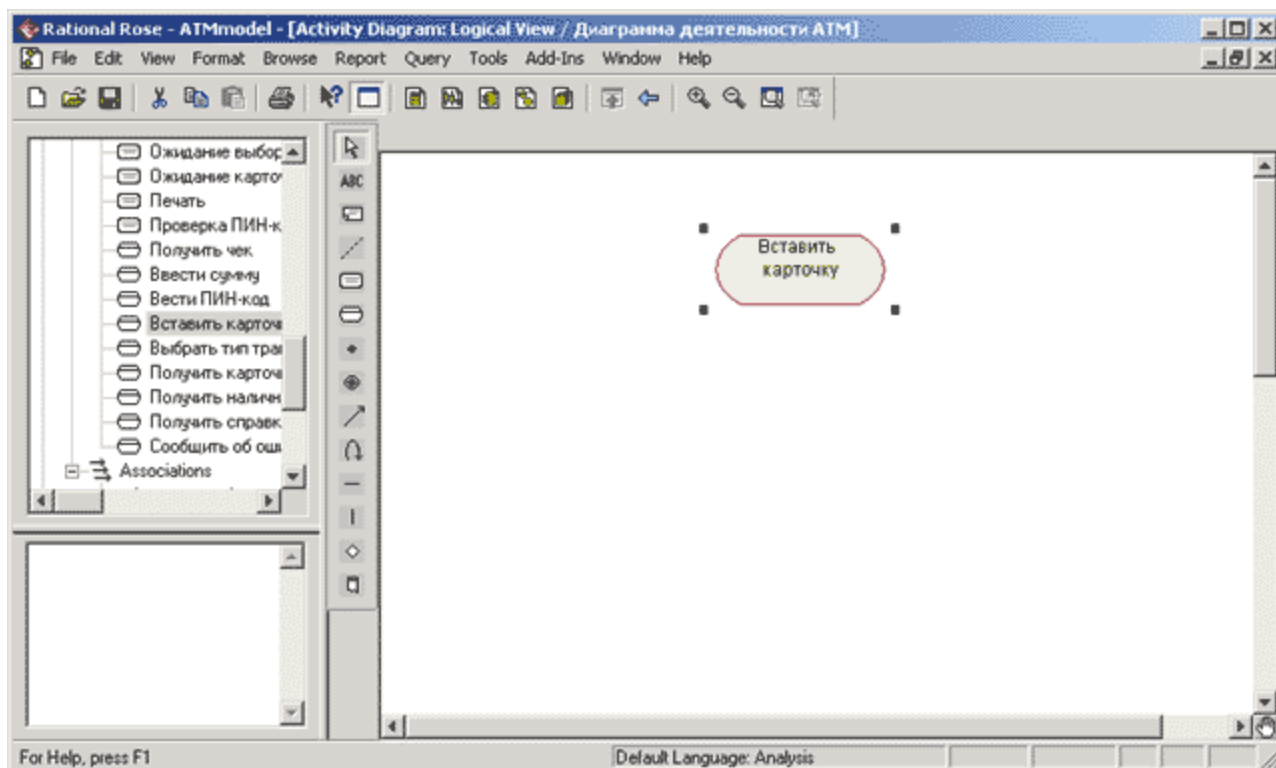


Рис. 10.1. Диаграмма деятельности после добавления на нее деятельности Вставить карточку

После добавления *деятельности* на диаграмму *деятельности* можно открыть *диалоговое окно* спецификации ее свойств и определить дополнительные свойства *деятельности*, доступные на соответствующих вкладках (рис. 10.2).

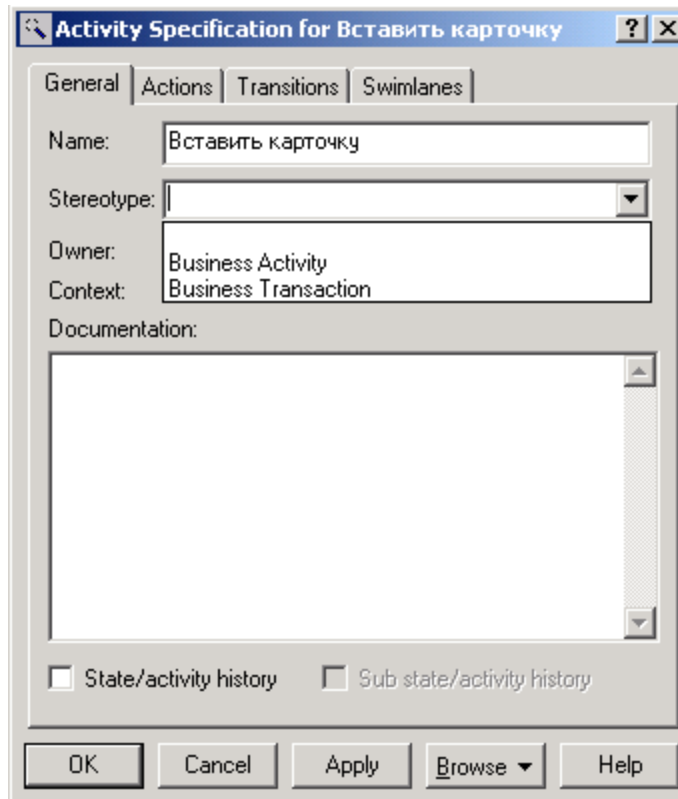


Рис. 10.2. Диалоговое окно спецификации свойств деятельности

При этом для *деятельности* становятся доступными для выбора два стереотипа: **Business Activity** (Бизнес-деятельность) и **Business Transaction** (Бизнес-транзакция), которые имеют собственное графическое изображение (см. табл. 10.1). На вкладке **Transitions** (Переходы) окна спецификации свойств *деятельности* можно определять и редактировать *переходы*, которые входят и выходят из рассматриваемой *деятельности*. Последняя вкладка **Swimlanes** (Дорожки) служит для спецификации дорожки, на которую помещается рассматриваемая *деятельность*.

Хотя программа IBM Rational Rose 2003 позволяет определить свойства *деятельности*, доступные на вкладке **Actions** (Действия), следует помнить, что внутренние действия являются свойствами общего понятия состояния, а внутренняя *деятельность* служит именем собственно *деятельности*, помещаемой на диаграмму *деятельности*. Поэтому для *деятельности* во избежание недоразумений лучше оставить эту вкладку пустой.

Добавление перехода и редактирование его свойств

Добавление *перехода* на диаграмму *деятельности* полностью аналогично диаграмме состояний. А именно, для добавления *перехода* между двумя *деятельностями* нужно с помощью левой кнопки мыши нажать кнопку с изображением *перехода* на специальной панели инструментов, отпустить левую кнопку мыши, щелкнуть левой кнопкой мыши на изображении исходной *деятельности* на диаграмме и отпустить ее на изображении

целевой *деятельности*. В результате этих действий на диаграмме появится изображение *перехода*, соединяющего две выбранных *деятельности*. Если в качестве одной из *деятельностей* является символ ветвления или соединения, то порядок добавления *перехода* сохраняется прежним.

Следует заметить, что при наличии в проекте законченной *диаграммы состояний* попытка добавить начальное состояние на диаграмму *деятельности* с помощью кнопки специальной панели инструментов окажется безуспешной. В этом случае *программа IBM Rational Rose 2003* фиксирует наличие в модели начального состояния и не позволит добавить его с помощью соответствующей кнопки на разрабатываемые *диаграммы состояний* или *деятельности*. Решить данную проблему можно посредством перетаскивания с помощью мыши начального состояния из браузера проекта на любую из вновь разрабатываемых диаграмм.

После добавления *перехода* на диаграмму *деятельности* становятся доступными для редактирования его свойства в специальном диалоговом окне (рис. 10.3), которое можно открыть *по* двойному щелчку левой кнопкой мыши на изображении *перехода*.

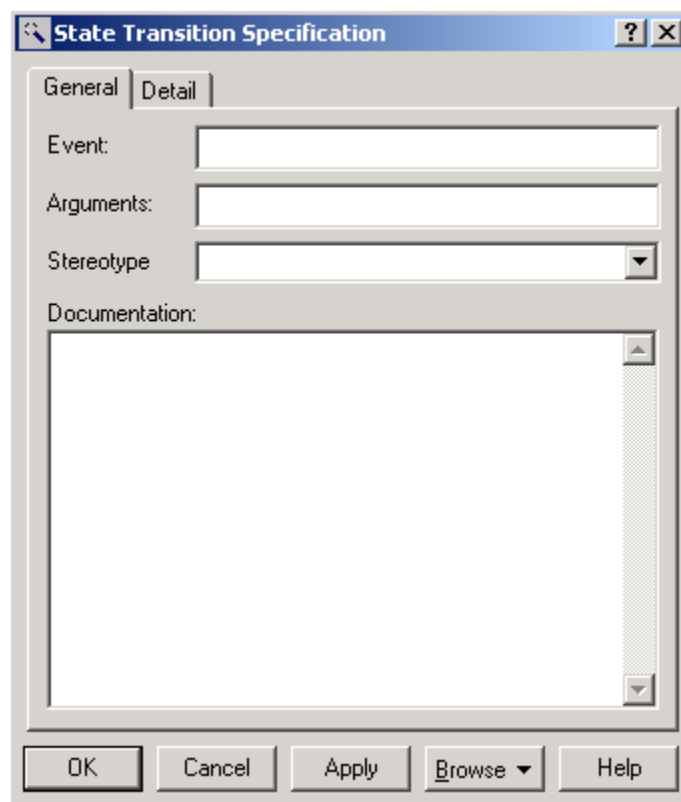


Рис. 10.3. Диалоговое окно спецификации свойств перехода

При спецификации свойств *переходов* следует помнить, что все *переходы* на диаграмме *деятельности* является нетриггерными, т.е. не имеют имен событий. *По* этой причине поле ввода с именем **Event** (Событие) для всех *переходов* должно оставаться пустым. Но

все *переходы*, выходящие из *символов ветвления (решения)*, должны иметь *сторожевые условия*, которые специфицируются на вкладке **Detail**(Подробно) диалогового окна спецификации свойств *перехода*.

Окончательное построение диаграммы деятельности модели банкомата

Для завершения построения диаграммы *деятельности* рассматриваемого примера следует описанным выше способом добавить оставшиеся *деятельности* и *переходы*. С этой целью следует выполнить следующие действия:

1. Добавить *деятельности* с именами: Ввести ПИН-код, Выбрать тип транзакции, Ввести сумму, Получить справку о состоянии счета, Получить наличные, Получить чек, Получить карточку и финальное состояние.
2. Добавить *символы ветвления (решения)*, расположив их между *деятельностями* с именами: Ввести ПИН-код и Выбрать тип транзакции, Выбрать тип транзакции и Ввести сумму, Ввести сумму и Получить справку о состоянии счета, Получить наличные и Получить чек, Получить чек и Получить карточку. При этом последний *символ решения* будет использоваться в качестве символа соединения.
3. Добавить *переход*, направленный от *деятельности* Ввести ПИН-код к *символу решения*.
4. Добавить *переход* со *сторожевым условием*: [ПИН-код верный], направленный от *символа решения* к *деятельности* Выбрать тип транзакции. Для задания *сторожевого условия* данного *перехода* следует ввести текст ПИН-код верный в поле ввода **Guard Condition** (*Сторожевое условие*) на вкладке **Detail** (Подробно) окна спецификации свойств данного *перехода* (рис. 10.4). При этом текст *сторожевого условия* следует вводить без скобок.

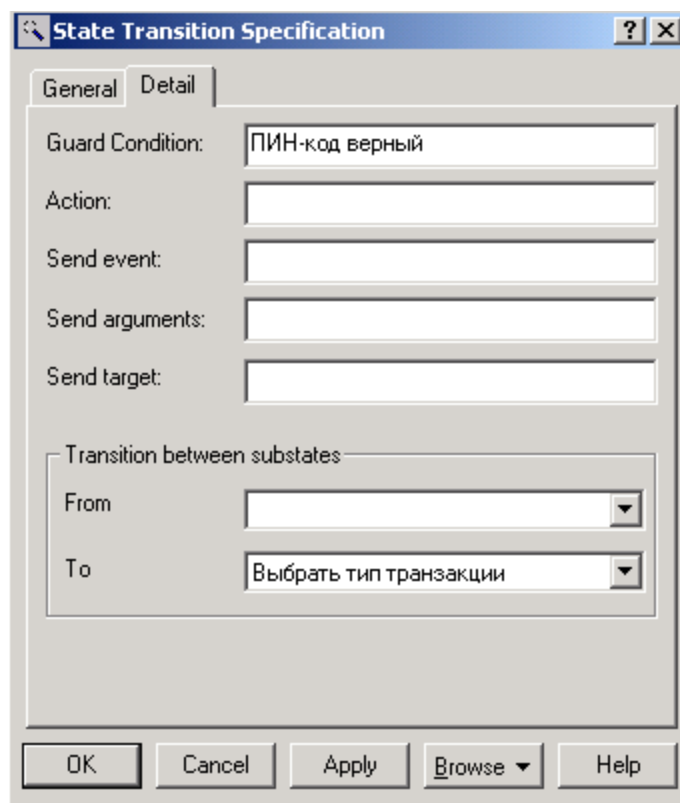


Рис. 10.4. Диалоговое окно спецификации свойств перехода при задании сторожевого условия

Для продолжения построения диаграммы *деятельности* следует выполнить следующие действия:

5. Добавить *переход* со *сторожевым условием*: [ПИН-код неверный], направленный от символа решения к символу соединения.
6. Добавить *переход*, направленный от деятельности **Выбрать тип транзакции** к символу решения.
7. Добавить *переход* со *сторожевым условием*: [выбор снятия суммы], направленный от символа решения к деятельности **Ввести сумму**.
8. Добавить *переход* со *сторожевым условием*: [выбор получения справки], направленный от символа решения к деятельности **Получить справку о состоянии счета**.
9. Добавить *переход*, направленный от деятельности **Ввести сумму** к символу решения.
10. Добавить *переход* со *сторожевым условием*: [сумма не превышает кредит], направленный от символа решения к деятельности **Получить наличные**.
11. Добавить *переход* со *сторожевым условием*: [сумма превышает кредит], направленный от символа решения к символу соединения.
12. Добавить *переход*, направленный от деятельности **Получить наличные** к символу решения.

13. Добавить *переход* со *сторожевым условием*: [выбрана печать чека], направленный от символа решения к деятельности **Получить чек**.
14. Добавить *переход* со *сторожевым условием*: [печать чека не выбрана], направленный от символа решения к символу соединения.
15. Добавить *переход*, направленный от деятельности **Получить чек** к символу соединения.
16. Добавить *переход*, направленный от деятельности **Получить справку о состоянии счета** к символу соединения.
17. Добавить *переход*, направленный от символа соединения к деятельности **Получить карточку**.
18. Добавить *переход*, направленный от деятельности **Получить карточку** к финальному состоянию.

Построенная таким образом *диаграмма деятельности* будет иметь следующий вид (рис. 10.5).

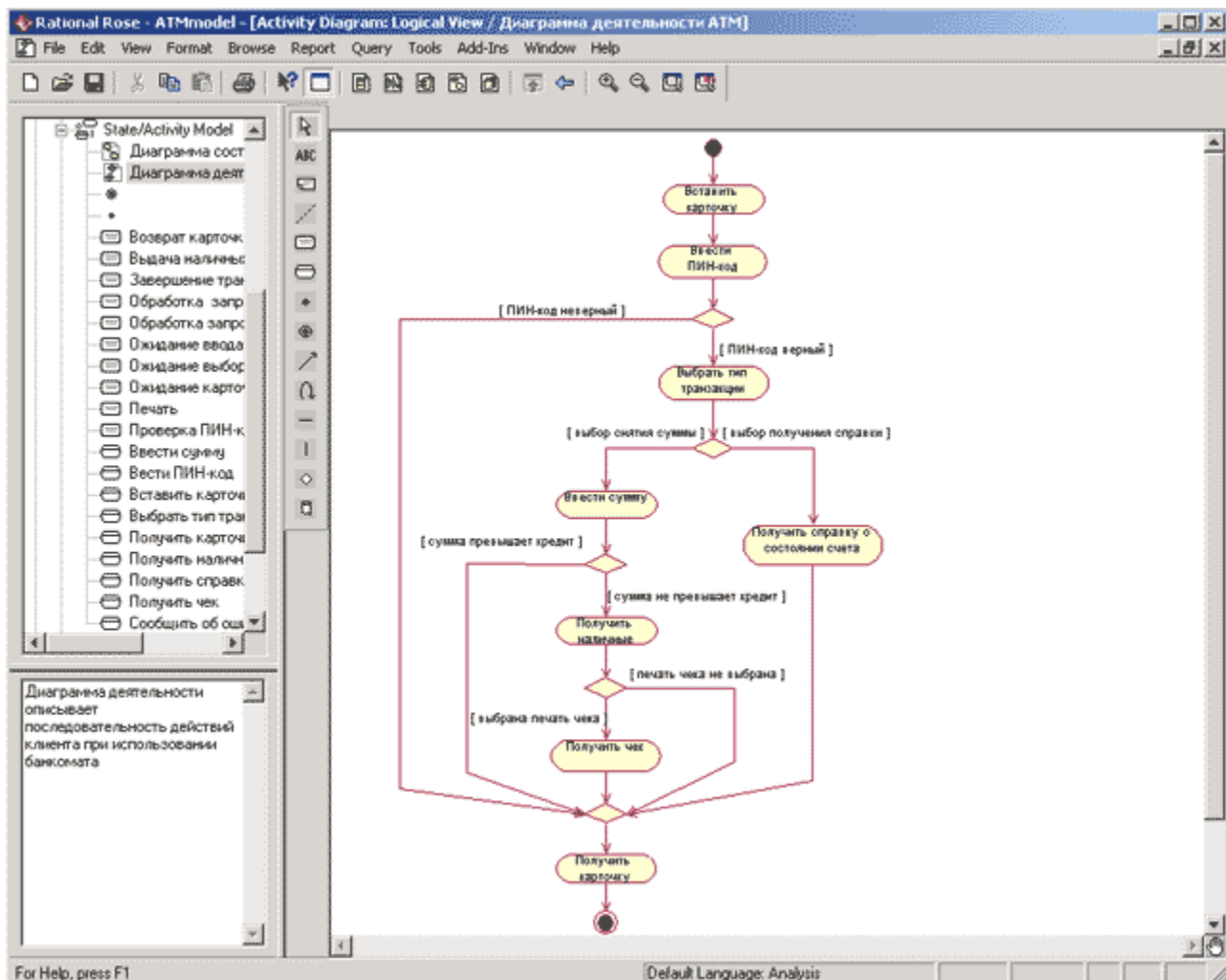


Рис. 10.5. Окончательный вид диаграммы деятельности для модели банкомата

Следует заметить, что в разрабатываемой модели *диаграмма деятельности* не описывает ситуацию блокирования карточки при трижды неверно введенном ПИН-коде. Дополнить данную *диаграмму деятельности*, которая учитывает данное условие в форме проверки отдельного условия, предлагается читателям самостоятельно в качестве упражнения.

Следует помнить, что в среде *IBM Rational Rose 2003* *диаграмма деятельности* не является необходимой для генерации программного кода. Поэтому разработку диаграмм этого типа, особенно в условиях дефицита времени, отпущенного на выполнение проекта, иногда опускают. В то же время следует отметить, что в проектах реинжиниринга и документирования бизнес-процессов *диаграмма деятельности* является основным средством визуализации бизнес-процессов в контексте языка *UML*.

Особенности проектов по моделированию бизнес-процессов в среде IBM Rational Rose 2003

Продолжая рассмотрение особенностей разработки диаграмм деятельности, следует отметить, что *программа IBM Rational Rose 2003* может быть успешно использована для выполнения проектов по моделированию *бизнес-процессов*. Наиболее подходящим типом диаграмм для визуального представления схем выполнения *бизнес-процессов* являются диаграммы деятельности, на которых дополнительно размещаются так называемые *дорожки* (**Swimlane**). Назначение *дорожек* состоит в том, чтобы указать *зоны ответственности* за выполнения отдельных деятельностей в рамках моделируемого *бизнес-процесса*. В качестве имен *дорожек* используются либо названия подразделений (департаментов) рассматриваемой компании, либо названия отдельных должностей сотрудников тех или иных подразделений.

Проекты по моделированию *бизнес-процессов* могут выполняться либо с целью реорганизации или *реинжиниринга* компании, либо с целью собственно документирования *бизнес-процессов*. Особенности данных проектов заключаются в том, что в обоих случаях необходимо построить модели *бизнес-процессов* некоторой существующей компании. Чтобы акцентировать внимание на подобных проектах, их часто называют проектами типа "As is" ("Как есть"). Соответственно проекты по разработке новых продуктов или моделей новых систем называют проектами типа "To be" ("Как должно быть").

В данном контексте рассматриваемый ранее проект по разработке системы управления банкоматом следует отнести к проектам типа "Как есть", поскольку при построении диаграмм предполагалась известная существующая технология использования банкоматов для обслуживания клиентов. С другой стороны, если бы стояла цель разработки новой модели банкомата с некоторой дополнительной функциональностью или, например, разработки нового Интернет-магазина, то подобные проекты можно было бы отнести к проектам типа "Как должно быть". Именно этот тип проектов служит базовым для принятой в курсе лекций последовательности разработки *канонических диаграмм* в нотации *UML*, начиная от представления *диаграмм вариантов использования* и заканчивая диаграммами физического представления.

Выполнение проектов типа "Как есть" по моделированию *бизнес-процессов* в большинстве случаев начинают с построения диаграмм деятельности, которые служат для графического представления схем выполнения *бизнес-процессов* и документооборота рассматриваемой компании. После этого, исходя из требований проекта, разрабатывается модель *диаграммы вариантов использования* и выполняется реорганизация *бизнес-процессов*. Наконец, в случае необходимости разработки или внедрения *корпоративной информационной системы*, строятся *диаграмма классов*, *диаграммы взаимодействия* и компонентов, которые служат основой для программной реализации соответствующего проекта.

Таким образом, первый этап выполнения проектов типа "Как есть" связан с построением моделей существующих *бизнес-процессов* компании в форме диаграмм деятельности. В качестве примера проекта этого типа в данной лекции рассматривается модель *бизнес-процесса* по оптовой продаже товаров со склада торговой компании. Хотя данный пример имеет упрощенный характер, он позволяет наглядно представить основные особенности моделирования *бизнес-процессов* в нотации языка *UML* с использованием средства *IBM Rational Rose 2003*.

Для вновь разрабатываемого проекта по моделированию *бизнес-процессов* торговой компании в среде *IBM Rational Rose 2003* создадим новый проект с именем: МодельБП. В качестве первой диаграммы проекта будет служить *диаграмма* деятельности, которая описывает отдельный *бизнес-процесс* в виде последовательности выполнения действий подразделениями компании при оптовой продаже товаров клиентам. Для удобства можно включить эту диаграмму в логическое *представление*, для чего необходимо в браузере проекта выделить логическое *представление* (**Logical View**) и выполнить операцию контекстного меню: **New** → **Activity Diagram** (Новая → *Диаграмма деятельности*).

Добавление дорожек на диаграмму деятельности

Для представления модели *бизнес-процесса* в форме диаграммы деятельности первоначально необходимо добавить на нее *дорожки*. Для добавления *дорожки* на диаграмму деятельности нужно с помощью левой кнопки мыши нажать кнопку с изображением пиктограммы *дорожки* на специальной панели инструментов, отпустить левую кнопку мыши и щелкнуть левой кнопкой мыши на свободном месте *рабочего листа* диаграммы. Добавить *дорожку* на диаграмму можно также с помощью *операции* главного меню: **Tools** → **Create** → **Swimlane** или с помощью *операции* контекстного меню: **New** → **Swimlane**, предварительно выделив диаграмму деятельности в браузере проекта.

В результате этих действий на диаграмме в области диаграммы появится изображение *дорожки* с вертикальной линией и именем *дорожки* **NewSwimlane** в верхней части, предложенное программой по умолчанию. Для задания имени *дорожки* следует открыть *диалоговое окно* спецификации ее свойств и ввести ее имя в *поле ввода* **Name** (рис. 11.1).

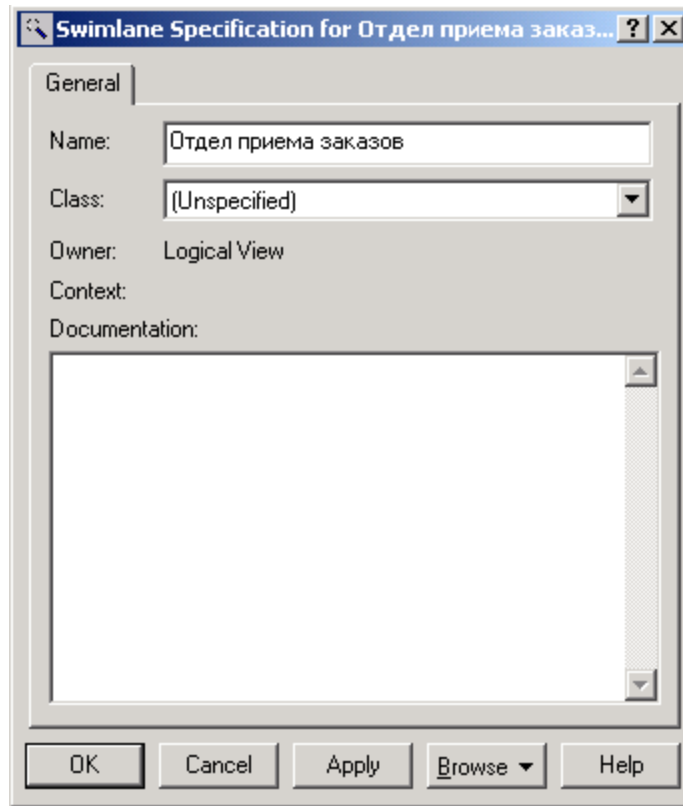


Рис. 11.1. Диалоговое окно спецификации свойств дорожки

Начиная практическую разработку модели *бизнес-процесса* оптовой продажи товаров со склада компании, последовательно добавим на диаграмму деятельности *дорожки* с именами отдельных подразделений компании: **Отдел приема заказов**, **Бухгалтерия**, **Склад** и **Отдел доставки** (рис. 11.2).

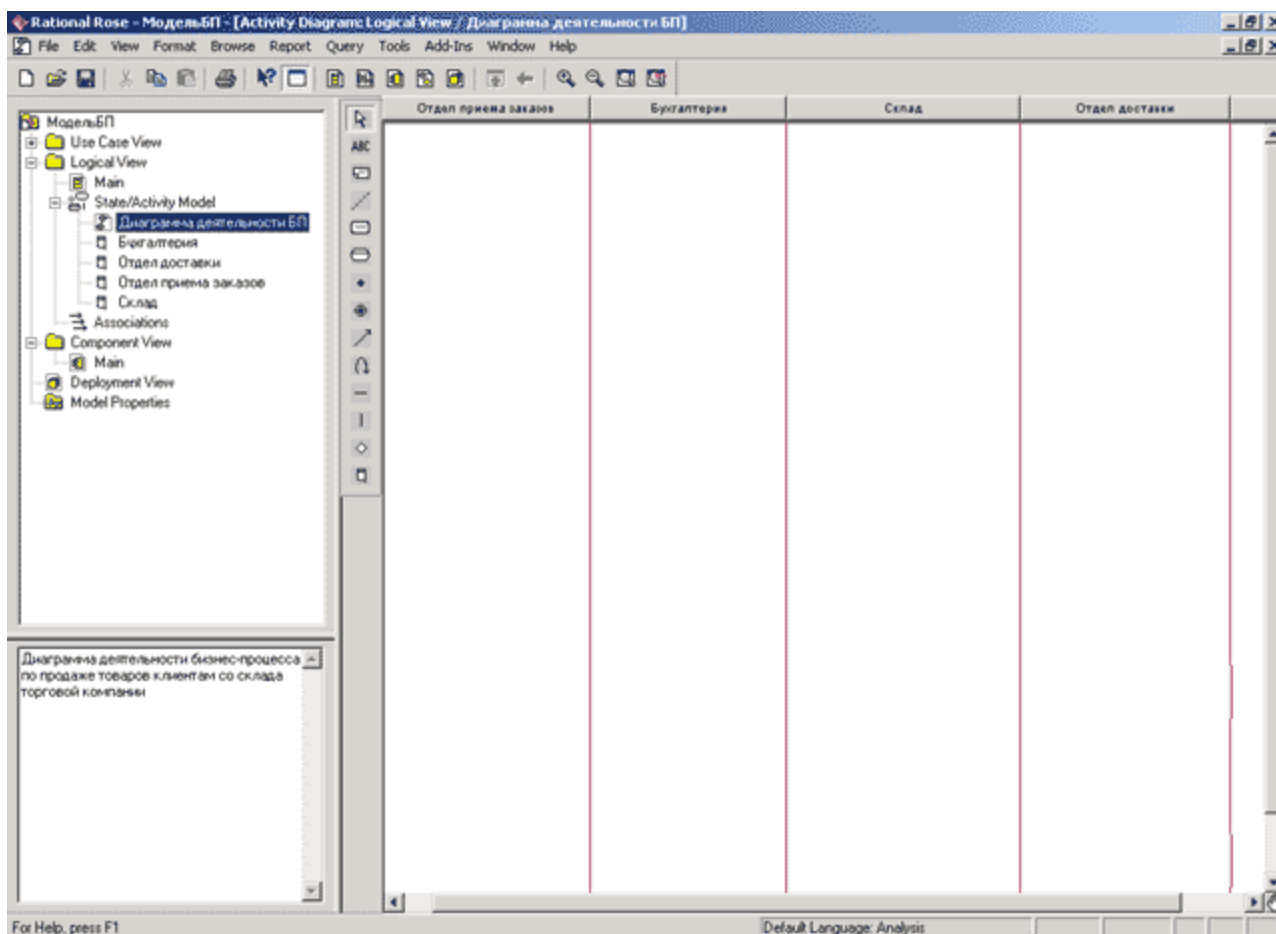


Рис. 11.2. Диаграмма деятельности после добавления на нее дорожек

После добавления *дорожек* на *диаграмму состояний* можно перейти к добавлению деятельности и переходов. В качестве первой деятельности добавим *деятельность* с именем **Принять заказ по факсу**, которую разместим в первой дорожке с именем **Отдел приема заказов**. Этот факт будет означать, что *деятельность Принять заказ по факсу* выполняется в **Отделе приема заказов** или, другими словами, сотрудники этого отдела несут ответственность за выполнение данной деятельности.

Деятельности **Принять заказ по факсу** должно предшествовать начальное состояние, которое также следует добавить в эту же *дорожку* и соединить переходом с этой деятельностью. После добавления начального состояния и перехода *диаграмма деятельности* будет иметь следующий вид (рис. 11.3).

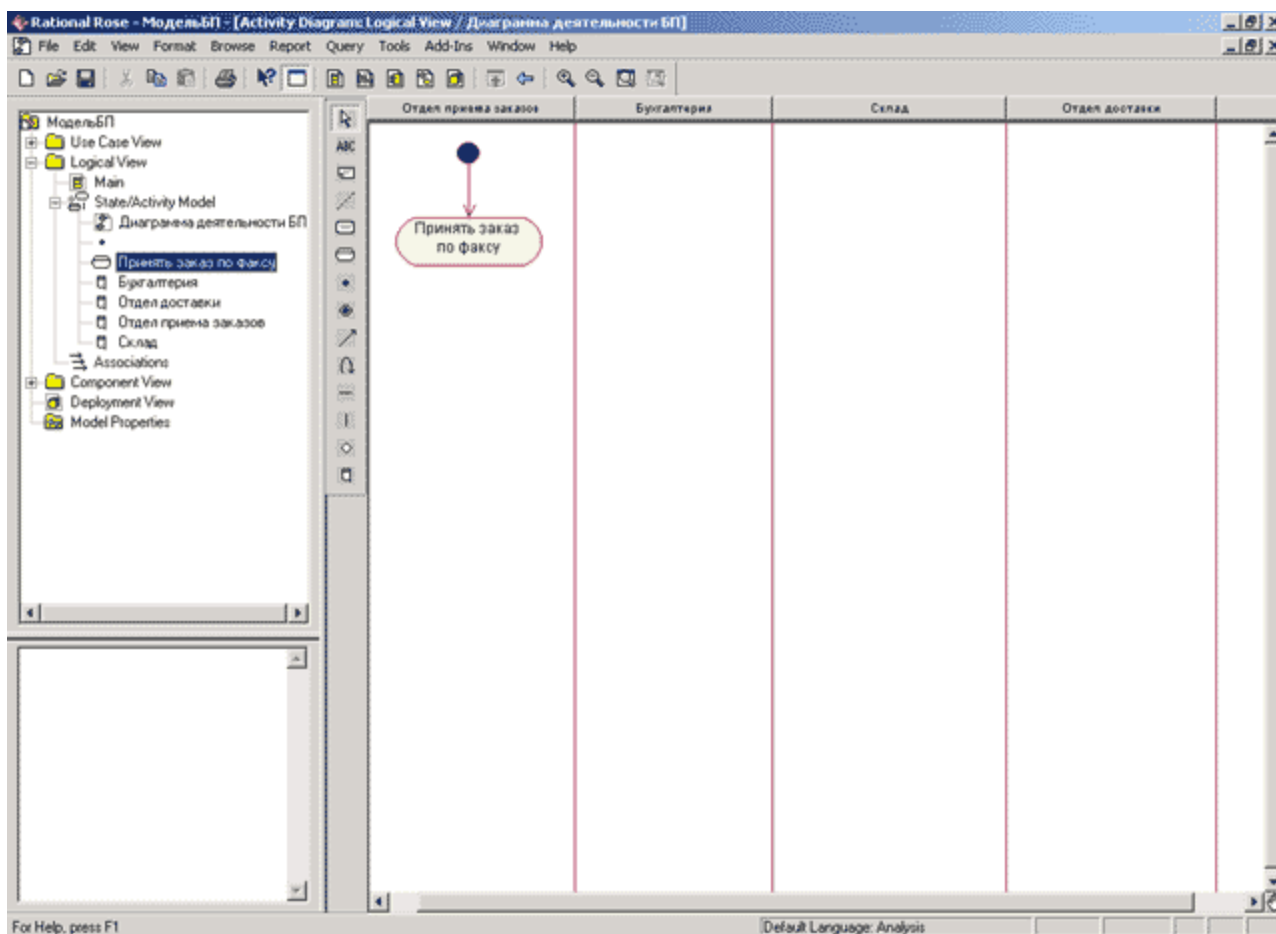


Рис. 11.3. Диаграмма деятельности после добавления на нее перехода из изначального состояния в деятельность Принять заказ по факсу

Построение диаграммы деятельности с дорожками для модели бизнес-процесса

Для построения диаграммы деятельности с *дорожками* для рассматриваемой модели бизнес-процесса следует добавить оставшиеся деятельности и переходы. С этой целью следует выполнить следующие действия:

1. Добавить деятельность с именем: **Заказать товар на складе** в *дорожку* **Отдел приема заказов**.
2. Добавить деятельности с именами: **Выставить счет к оплате** и **Получить оплату за товар** в *дорожку* **Бухгалтерия**.
3. Добавить деятельности с именами: **Подобрать товар** и **Подготовить товар к отправке** в *дорожку* **Склад**.
4. Добавить деятельность с именем: **Отправить товар клиенту** в *дорожку* **Отдел доставки**.

5. Добавить символ горизонтальной синхронизации в дорожки **Отдел приема заказов** и **Склад**. Следует заметить, что первый символ будет использован для разделения параллельных потоков деятельности, а второй - для слияния этих потоков.
6. Добавить переход, направленный от деятельности **Принять заказ по факсу** к деятельности **Заказать товар на складе**.
7. Добавить переход, направленный от деятельности **Заказать товар на складе** к символу горизонтальной синхронизации.
8. Добавить переход, направленный от символа горизонтальной синхронизации к деятельности **Выставить счет к оплате**.
9. Добавить переход, направленный от символа горизонтальной синхронизации к деятельности **Подобрать товар**.
10. Добавить переход, направленный от деятельности **Выставить счет к оплате** к деятельности **Получить оплату за товар**.
11. Добавить переход, направленный от деятельности **Подобрать товар** к деятельности **Подготовить товар к отправке**.
12. Добавить переход, направленный от деятельности **Получить оплату за товар** к символу горизонтальной синхронизации.
13. Добавить переход, направленный от деятельности **Подготовить товар к отправке** к символу горизонтальной синхронизации.
14. Добавить переход, направленный от символа горизонтальной синхронизации к деятельности **Отправить товар клиенту**.
15. Добавить переход, направленный от деятельности **Отправить товар клиенту** к финальному состоянию.

Построенная таким образом *диаграмма* деятельности с *дорожками* будет иметь следующий вид (рис. 11.4).

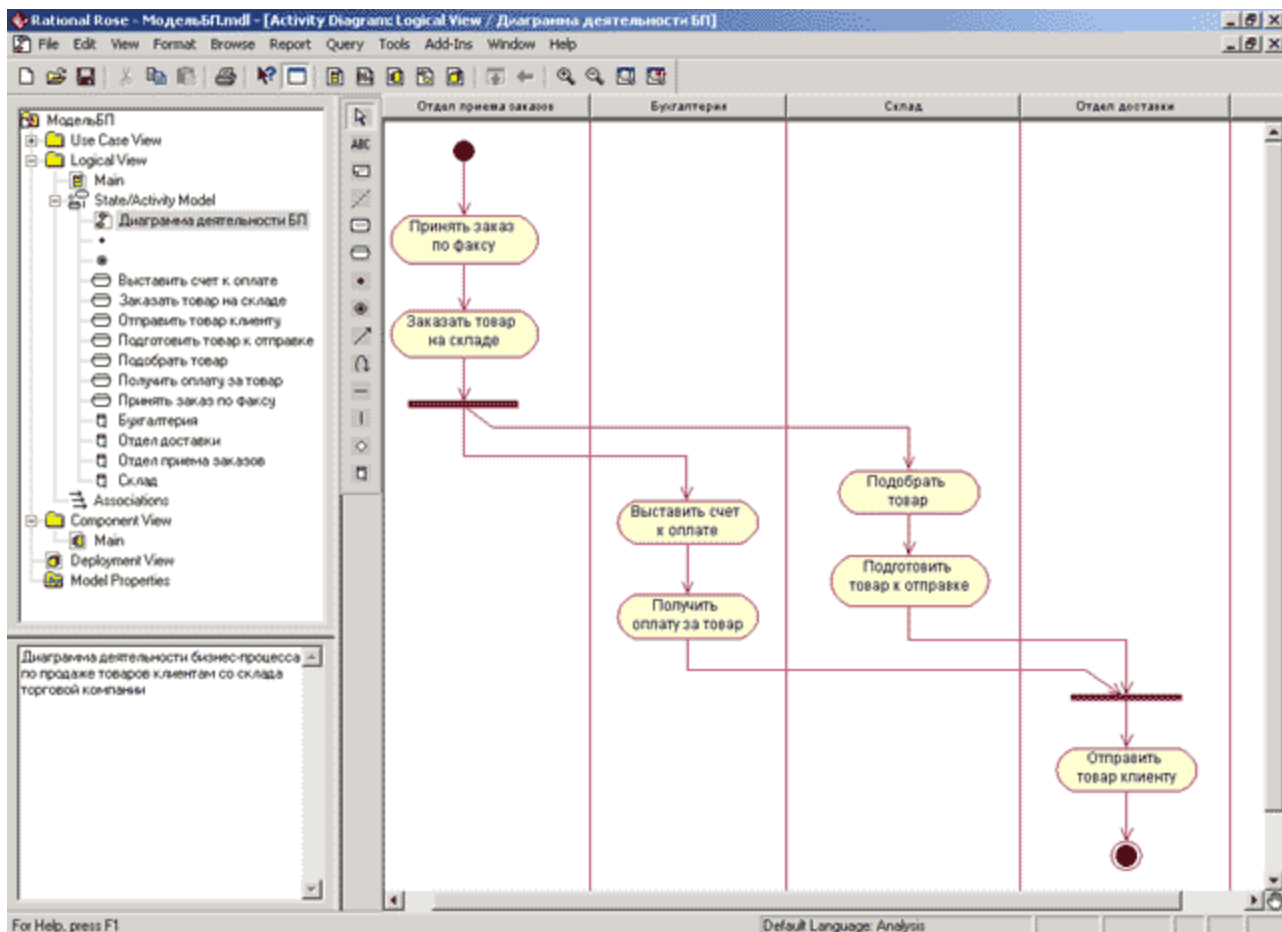


Рис. 11.4. Диаграмма деятельности с дорожками для модели бизнес-процесса

Следует заметить, что в разрабатываемой модели *диаграмма* деятельности не описывает ситуацию, когда заказанного клиентом товара не окажется на складе. Дополнить данную диаграмму, которая учитывает данное условие в форме проверки отдельного условия, предлагается читателям самостоятельно в качестве упражнения.

Построение диаграммы деятельности с дорожками и потоком объектов

Для построения диаграммы деятельности с *дорожками* и *потоком объектов* для рассматриваемой модели *бизнес-процесса* следует добавить на диаграмму *объекты* и стрелки *потоков объектов*. *Объекты* на диаграмме деятельности могут обозначать отдельные документы, которые необходимы для выполнения моделируемого *бизнес-процесса*. Соответственно *поток объектов* служит моделью документооборота рассматриваемой компании. Для добавления на диаграмму *объекта* следует воспользоваться соответствующей кнопкой на специальной панели инструментов. При этом данную кнопку предварительно следует на нее добавить, поскольку по умолчанию на панели она отсутствует.

В качестве первого *объекта* добавим на диаграмму деятельности *объект* с именем заказ, для которого зададим состояние: **получен**. Для задания состояния добавленного *объекта* следует открыть *диалоговое окно* свойств данного *объекта*, во вложенном списке **State** (Состояние) выбрать нужное состояние или задать новое (рис. 11.5). При этом будет открыто дополнительное окно свойств состояния, в которое можно занести всю информацию по данному состоянию.

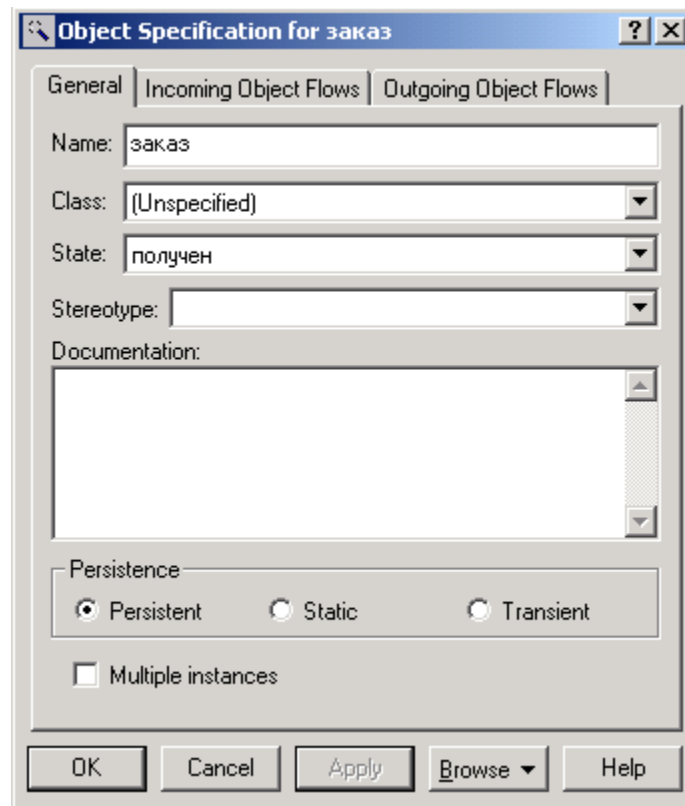


Рис. 11.5. Диалоговое окно спецификации свойств объекта

Для завершения построения диаграммы деятельности рассматриваемого примера следует описанным выше способом добавить оставшиеся *объекты* и стрелки *потоков объектов*. С этой целью следует выполнить следующие действия:

1. Добавить стрелку *потока объектов*, направленную от деятельности **Принять заказ по факсу** к объекту заказ в состоянии получен.
2. Добавить стрелку *потока объектов*, направленную от *объекта* заказ в состоянии получен к деятельности **Заказать товар на складе**.
3. Добавим объект с именем заказ, для которого зададим состояние: **оформлен**. Следует заметить, что для добавления на диаграмму деятельности уже существующего в модели *объекта* его следует просто перетащить из браузера проекта на диаграмму и задать ему новое состояние.
4. Добавить стрелку *потока объектов*, направленную от деятельности **Заказать товар на складе** к объекту заказ в состоянии **оформлен**.

5. Добавить стрелку *потока объектов*, направленную от *объекта* заказ в состоянии **оформлен** к деятельности **Выставить счет к оплате**.
6. Добавим объект с именем **счет**, для которого зададим состояние: **выставлен**.
7. Добавить стрелку *потока объектов*, направленную от деятельности **Выставить счет к оплате** к объекту **счет** в состоянии **выставлен**.
8. Добавить стрелку *потока объектов*, направленную от *объекта* **счет** в состоянии **выставлен** к деятельности **Получить оплату за товар**.
9. Добавим объект с именем **счет**, для которого зададим состояние: **оплачен**.
10. Добавить стрелку *потока объектов*, направленную от деятельности **Получить оплату за товар** к объекту **счет** в состоянии **оплачен**.
11. Добавить стрелку *потока объектов*, направленную от *объекта* **счет** в состоянии **оплачен** к деятельности **Отправить товар клиенту**.
12. Добавим объект с именем **накладная**, для которого зададим состояние: **выписана**.
13. Добавить стрелку *потока объектов*, направленную от деятельности **Заказать товар на складе** к объекту **накладная** в состоянии **выписана**.
14. Добавить стрелку *потока объектов*, направленную от *объекта* **накладная** в состоянии **выписана** к деятельности **Подобрать товар**.
15. Добавим объект с именем **накладная**, для которого зададим состояние: **оформлена**.
16. Добавить стрелку *потока объектов*, направленную от деятельности **Подготовить товар к отправке** к объекту **накладная** в состоянии **оформлена**.
17. Добавить стрелку *потока объектов*, направленную от *объекта* **накладная** в состоянии **оформлена** к деятельности **Отправить товар клиенту**.

Построенная таким образом *диаграмма* деятельности с *дорожками* и *потоком объектов* будет иметь следующий вид (рис. 11.6).

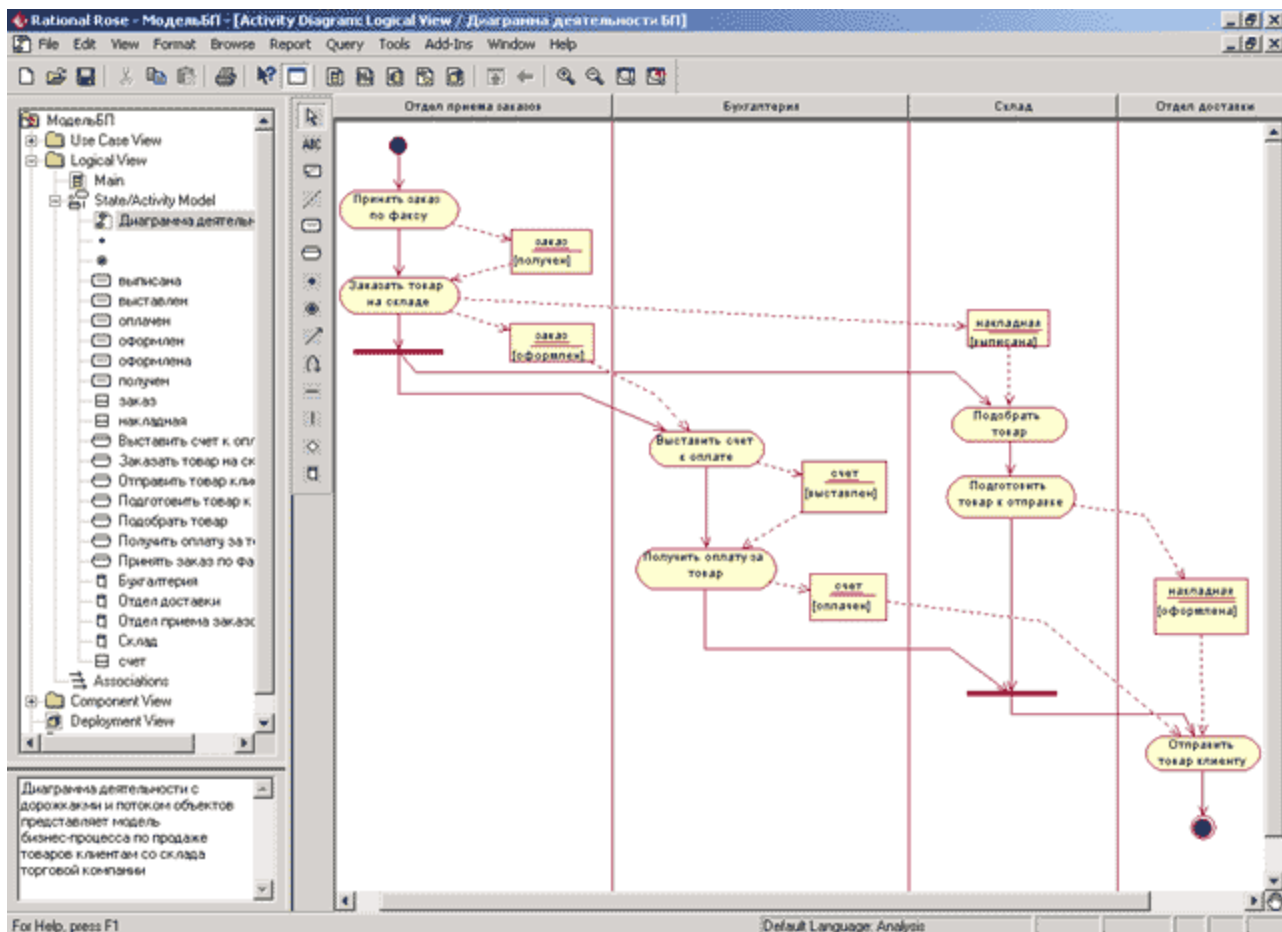


Рис. 11.6. Окончательный вид диаграммы деятельности для модели бизнес-процесса

Для большей наглядности представления данной модели можно задать для всех деятельностей стереотип **Business Activity** (Бизнес-деятельность), который будет означать в данном контексте *деятельность*, выполняемую в рамках некоторого *бизнес-процесса*. Напомним, что изменить стереотип деятельности можно с помощью выбора нужного варианта стереотипа в окне спецификации свойств деятельности. Соответствующий вариант изображения диаграммы деятельности представлен на рис. 11.7.

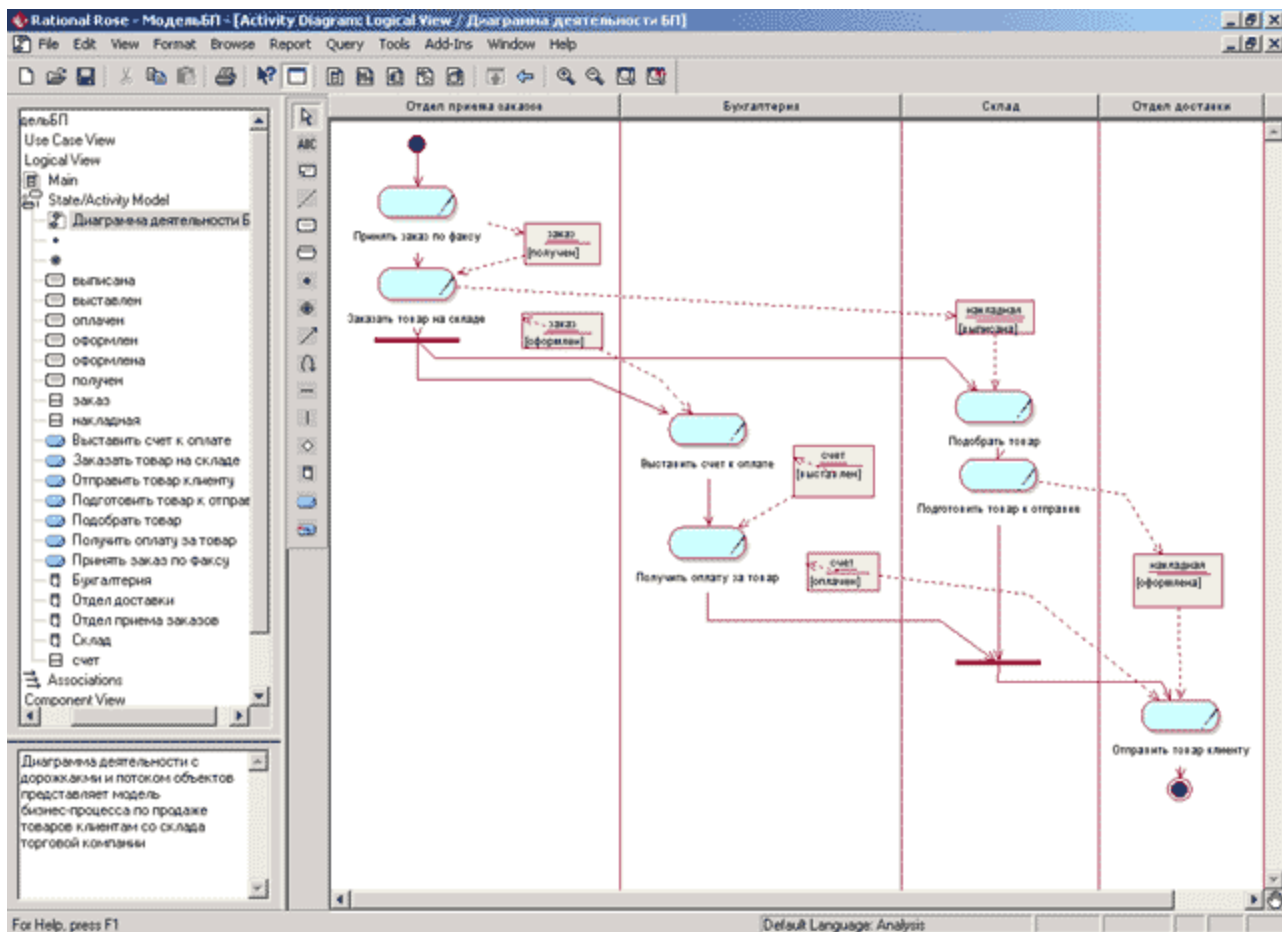

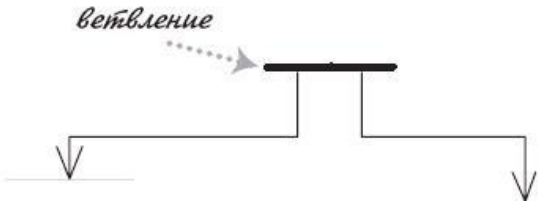
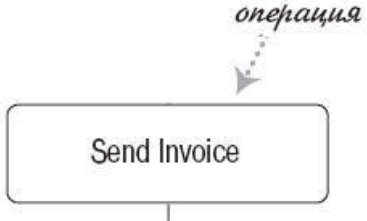
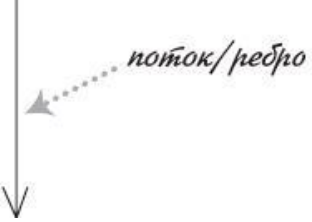
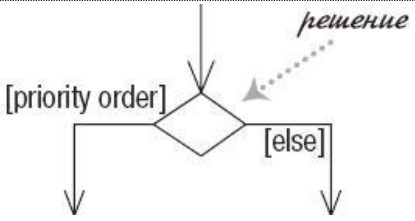


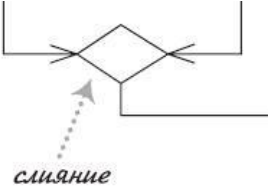
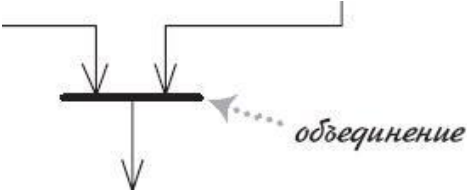


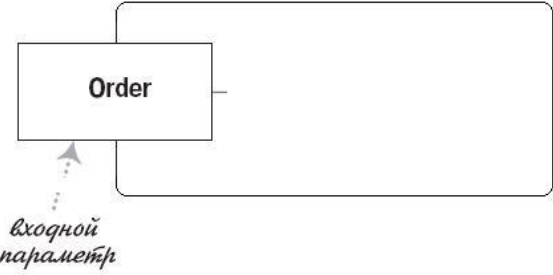
Рис. 11.7. Окончательный вид диаграммы деятельности для модели бизнес-процесса со стереотипами деятельности

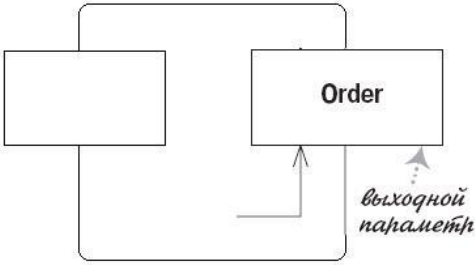
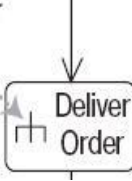
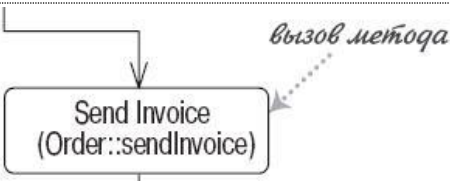
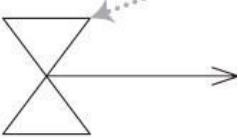
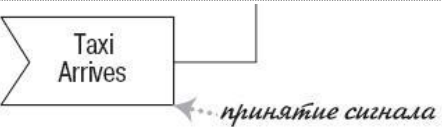
Следует заметить, что в разрабатываемой модели *диаграмма* деятельности не описывает ситуацию, когда клиент отказался от оплаты товара после выставления ему счета. Дополнить данную диаграмму деятельности, которая учитывает данное условие, предлагается читателям самостоятельно в качестве упражнения.

Хотя в среде *IBM Rational Rose 2003* *диаграмма* деятельности не является необходимой для генерации программного кода, диаграммы данного типа имеют большое значение для документирования *бизнес-процессов* и их последующей сертификации по международному стандарту *ISO 9000*. Поэтому разработка диаграмм этого типа занимает центральное место при выполнении проектов по реинжинирингу и оптимизации *бизнес-процессов* с использованием нотации *UML*.

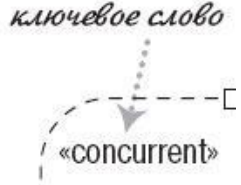



Терминология

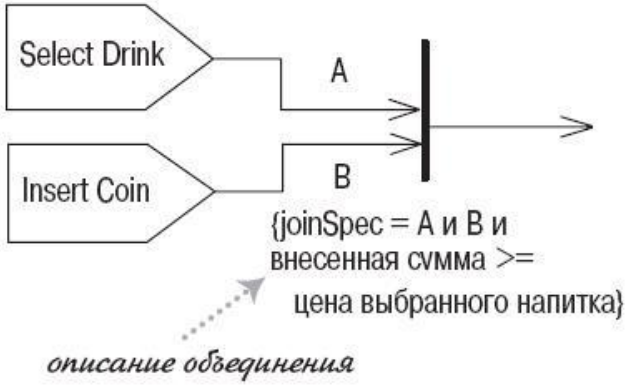
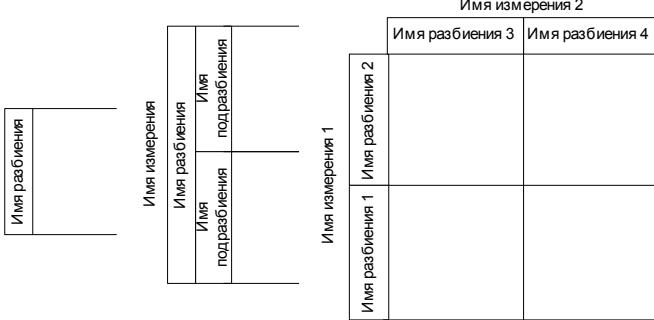
Термин	Изображение	Описание
Начальный узел		Старт диаграммы
Ветвление		Ветвление (fork) , имеет один входной поток и несколько выходных параллельных потоков.
Операция		
Поток / ребро		В UML 2 параллельно употребляются термины поток (flow) и ребро (edge) для обозначения связи между двумя операциями. Самый простой вид ребра – это обычная стрелка между двумя операциями.
Решение		Каждый раз при достижении решения (decision) выбирается только один из выходных потоков, поэтому защиты должны быть взаимно исключающими.

Слияние		Слияние (merge) обозначает завершение условного поведения, которое было начато решением.
Объединение		При наличии параллелизма необходима синхронизация. Мы не закрываем заказ, пока он не оплачен и не доставлен. Это показывается с помощью объединения (join)
Конец деятельности		Окончание диаграммы
Имя деятельности		
Входной параметр		

Выходной параметр		
"Грбли"	<p>«грбли» означают диаграмму вложенной деятельности</p> 	Операции могут быть реализованы или как вложенные деятельности или как методы классов. Вложенную деятельность можно обозначить с помощью символа «грблей».
Вызов метода		
Временной сигнал	<p>временной сигнал</p> 	Временной сигнал (time signal) приходит по прошествии времени. Такие сигналы могут означать конец месяца в отчетном периоде или приходиться каждую секунду в контроллере реального времени.
Принятие сигнала		

Посылка сигнала		
Прием сигнала		
Разъем		
Узел объекта		
Контакт		<p>Процедуры, как и методы, могут иметь параметры. Показывать на диаграмме деятельности информацию о параметрах не обязательно, но при желании можно отобразить параметры с помощью контактов (pins)</p>
Выражение преобразования		<p>Если требуется нарисовать точную диаграмму деятельности, то необходимо обеспечить соответствие выходных параметров</p>

		<p>одной процедуры входным параметрам другой. Если они не совпадают, то можно указать преобразование (transformation) для перехода от одной процедуры к другой.</p>
Ключевое слово		
Выходной список контактов		
Область расширения		<p>Область расширения (expansion region) отмечает область диаграммы деятельности, где операции выполняются один раз для каждого элемента коллекции.</p>
Окончание потока		<p>Окончание потока (flow final) означает завершение конкретного потока без завершения всей активности.</p>

<p>Описание объединения</p>	 <p><i>описание объединения</i></p>	<p>Описание объединения (join specification) – это логическое выражение, присоединенное к объединению.</p>
<p>Разбиение деятельности</p>		<p>Разбиение деятельности (activity partitions). Простое <u>одномерное разбиение</u> – <u>плавательные дорожки (swim lanes)</u>, UML 1. В UML 2 сетка может быть двумерной. Кроме того, можно взять каждое измерение и разделить строчки на столбцы, создавая тем самым иерархическую структуру.</p>