

## **9. Разработка диаграммы состояний и редактирование свойств ее элементов**

Особенности разработки диаграммы состояний в среде IBM Rational Rose 2003. Добавление состояния на диаграмму состояний и редактирование его свойств. Добавление перехода и редактирование его свойств. Пример диаграммы состояний для модели банкомата.





### **Особенности разработки диаграммы состояний в среде IBM Rational Rose 2003**









Переходя к рассмотрению диаграммы *состояний*, следует отметить, что в среде IBM Rational Rose 2003 этот тип диаграмм может относиться к отдельному классу, операции класса, варианту использования, пакету или представлению. Общие рекомендации по построению диаграммы *состояний* были рассмотрены в лекциях 9 и 10 курса «Основы объектно-ориентированного моделирования в нотации UML». Для того чтобы построить диаграмму *состояний*, ее вначале необходимо создать и активизировать.

Начать построение диаграммы *состояний* 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 - новая диаграмма *состояний*.
- Выделить логическое представление (**Logical View**) или представление вариантов использования (**Use Case View**) в браузере проекта и выполнить операцию контекстного меню: **New**→**Statechart Diagram** (Новая→Диаграмма *состояний*).
- 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**→**Statechart Diagram** (Новая→Диаграмма *состояний*).
- Выполнить операцию главного меню: **Browse**→**State Machine Diagram** (Обзор→Диаграмма *состояний*), после чего следует выбрать представление и тип разрабатываемой диаграммы.

В результате выполнения этих действий появляется новое окно с чистым рабочим листом диаграммы *состояний* и специальная панель инструментов, содержащая кнопки с изображением графических элементов модели, необходимых для разработки диаграммы *состояний* (табл. 9.1). Назначение отдельных кнопок панели можно узнать из всплывающих подсказок.

<b>Таблица 9.1. Назначение кнопок специальной панели инструментов диаграммы состояний</b>		
<b>Графическое изображение</b>	<b>Всплывающая подсказка</b>	<b>Назначение кнопки</b>
	Selection Tool	Превращает изображение курсора в форму стрелки для последующего выделения элементов на диаграмме
	Text Box	Добавляет на диаграмму текстовую область
	Note	Добавляет на диаграмму примечание
	Anchor Note to Item	Добавляет на диаграмму связь примечания с

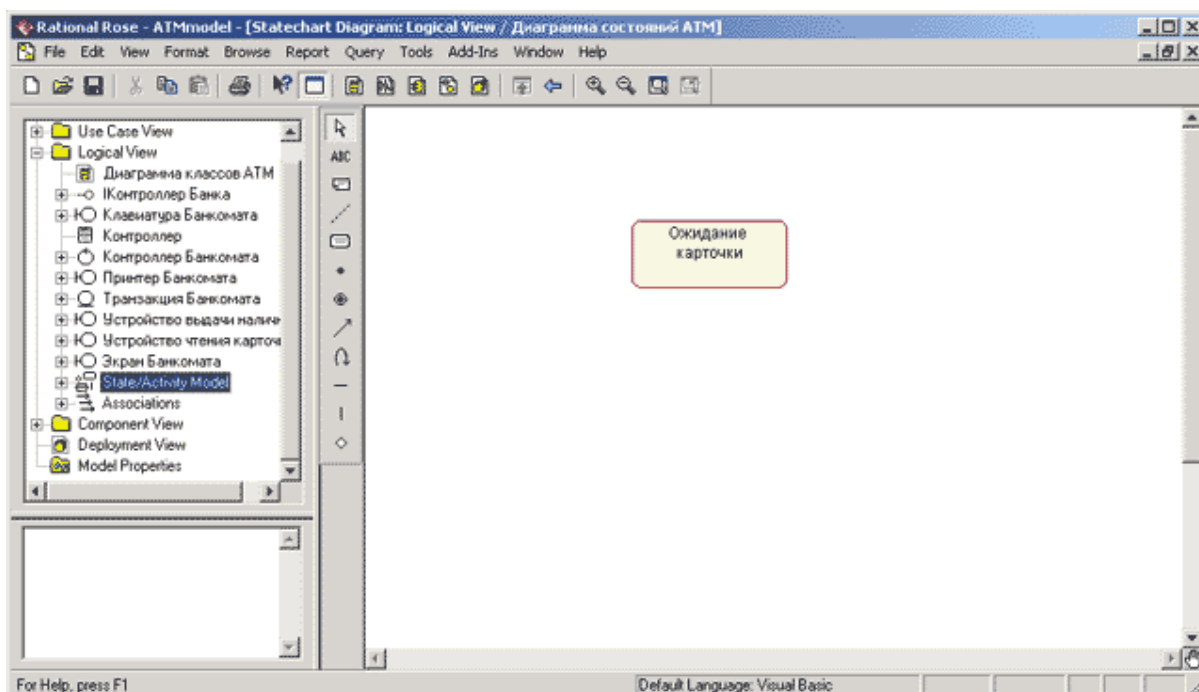
		соответствующим графическим элементом диаграммы
	State	Добавляет на диаграмму <i>состояние</i>
	Start State	Добавляет на диаграмму начальное <i>состояние</i>
	End State	Добавляет на диаграмму конечное <i>состояние</i>
	State Transition	Добавляет на диаграмму <i>переход</i>
	Transition to Self	Добавляет на диаграмму рефлексивный <i>переход</i>
	Horizontal Synchronization	Добавляет на диаграмму горизонтально расположенный символ синхронизации (по умолчанию отсутствует)
	Vertical Synchronization	Добавляет на диаграмму вертикально расположенный символ синхронизации (по умолчанию отсутствует)
	Decision	Добавляет на диаграмму символ принятия решения для альтернативных <i>переходов</i> (по умолчанию отсутствует)

По умолчанию на специальной панели инструментов могут отсутствовать кнопки с тремя последними графическими элементами из таблицы 9.1. При необходимости их можно добавить на специальную панель диаграммы *состояний* аналогично способу, рассмотренному ранее в лекции 3 (рис. 3.1). Продолжая разработку проекта по моделированию системы управления банкоматом, можно приступить к разработке новой диаграммы *состояний*. С этой целью для диаграммы *состояний* модели банкомата зададим имя *Диаграмма состояний АТМ*, а в секцию ее документации введем текст «Диаграмма *состояний* описывает конечный автомат банкомата».

## Добавление состояния на диаграмму состояний и редактирование его свойств

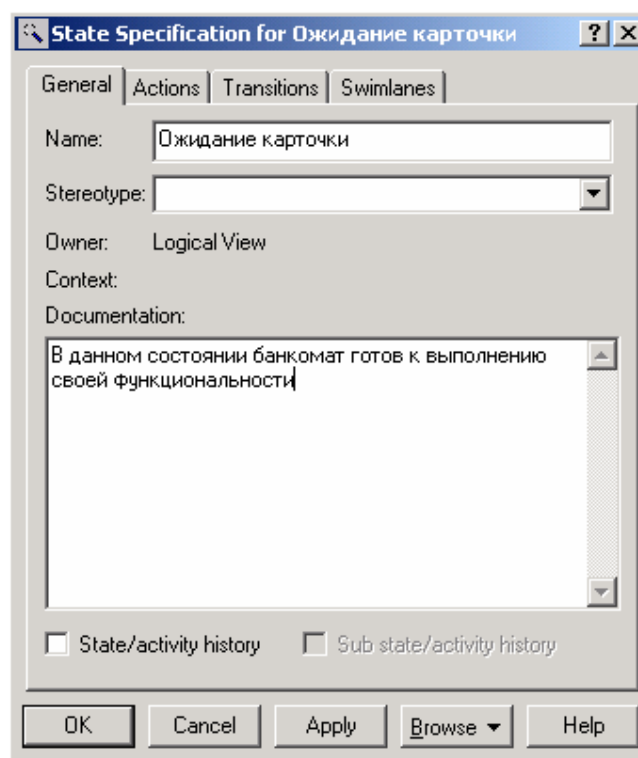
Для добавления *состояния* на диаграмму *состояний* необходимо с помощью левой кнопки мыши нажать кнопку с изображением пиктограммы *состояния* на специальной панели инструментов, отпустить левую кнопку мыши и щелкнуть левой кнопкой мыши на свободном месте рабочего листа диаграммы. На диаграмме появится изображение *состояния* с маркерами изменения его геометрических размеров и предложенным средой именем по умолчанию, которое разработчику следует изменить.

Для диаграммы *состояний* модели банкомата в качестве имени первого добавленного *состояния* изменим предложенное программой по умолчанию имя *NewState* на *Ожидание карточки* (рис. 9.1). Задать имя *состояния* можно либо непосредственно при добавлении нового *состояния* на диаграмму *состояний*, либо открыв окно спецификации свойств нового *состояния*.



**Рис. 9.1.** Диаграмма состояний после добавления на нее состояния Ожидание карточки

Для добавленного *состояния* можно открыть диалоговое окно его свойств двойным щелчком левой кнопкой мыши на изображении этого элемента на диаграмме. В этом случае активизируется диалоговое окно со специальными вкладками, в поля которых можно занести всю информацию по данному *состоянию* (рис. 9.2).



**Рис. 9.2.** Диалоговое окно спецификации свойств состояния

При необходимости в диалоговом окне спецификации свойств выбранного *состояния* можно задать вложенное *историческое состояние*. Для этого следует выставить отметку у свойства **State/activity history** (*Историческое состояние/деятельность*) и нажать кнопку

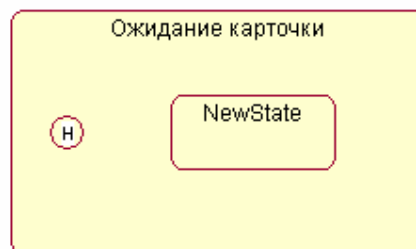
**Apply.** В результате внутри исходного *состояния* появится вложенное *историческое состояние* (рис. 9.3, а).



**Рис. 9.3.** Добавление вложенного исторического состояния (а) и состояния глубокой истории (б) для состояния Ожидание карточки

Чтобы обычное *историческое состояние* превратить в *состояние* глубокой истории, следует дополнительно выставить отметку у свойства **Sub state/activity history** (Историческое под-состояние/деятельность), которое становится доступным для редактирования после выбора первого свойства, и нажать кнопку **Apply**. В результате внутри исходного *состояния* появится вложенное *состояние* глубокой истории (рис. 9.3, б).

Чтобы обычное *состояние* превратить в *композит*, следует при добавлении нового *состояния* поместить его внутри границы того *состояния*, которое необходимо сделать *композитным*. В результате внутри исходного *состояния* появится новое вложенное *состояние* с именем **NewState**, которое при перемещении композита в области диаграммы *состояний* всегда будет находиться внутри своего композита (рис. 9.4).



**Рис. 9.4.** Превращение состояния Ожидание карточки в композитное состояние

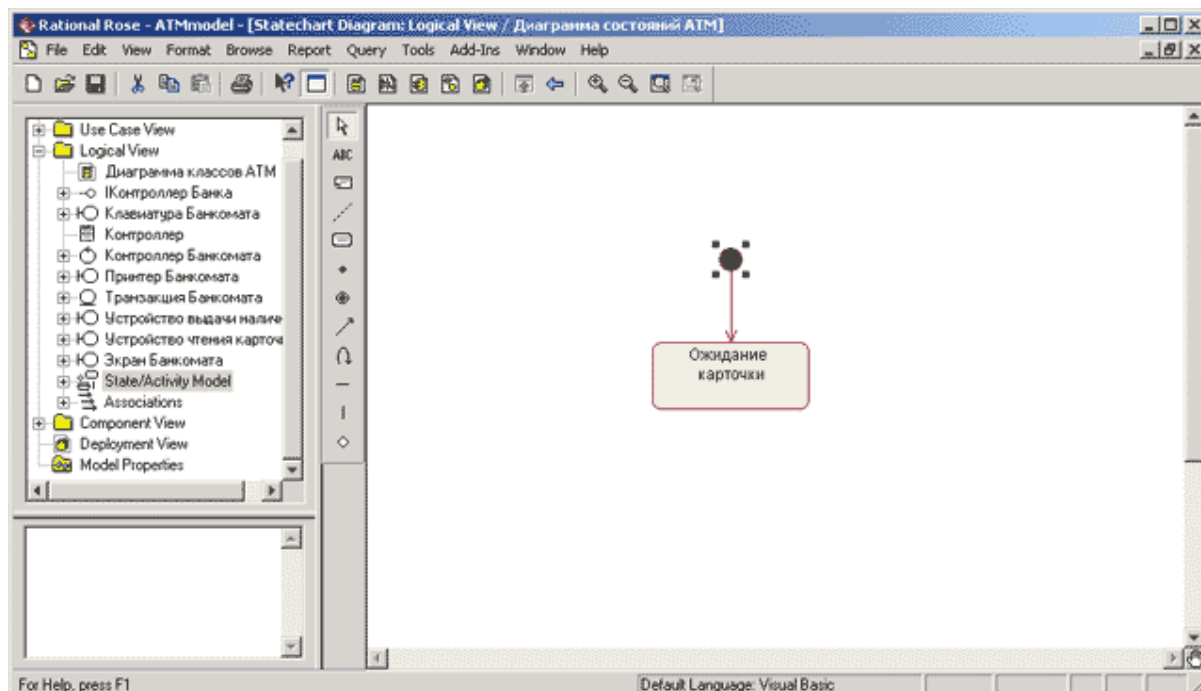
Рассмотренные выше действия приведены только с целью иллюстрации особенностей спецификации исторических и вложенных подсостояний и не относятся к разрабатываемой модели банкомата.

Дополнительно можно определить следующие свойства *состояний*: задать текстовый стереотип *состояния*, определить внутренние действия на входе и выходе, а также внутреннюю деятельность. Эти свойства доступны для редактирования на вкладке **General** (Общие) и **Actions** (Действия). На вкладке **Transitions** (Переходы) можно определять и редактировать *переходы*, которые входят и выходят из рассматриваемого состояния. Последняя вкладка **Swimlanes** (Дорожки) служит для спецификации дорожек, которые, в контексте языка UML, определяются для диаграммы деятельности.

## Добавление перехода и редактирование его свойств

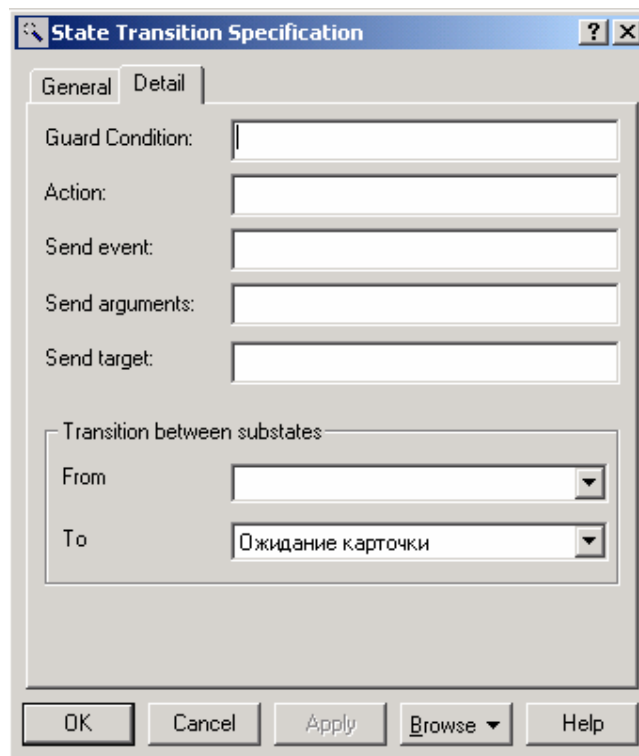
Для добавления *перехода* между двумя *состояниями* нужно с помощью левой кнопки мыши нажать кнопку с изображением *перехода* на специальной панели инструментов, отпустить левую кнопку мыши, щелкнуть левой кнопкой мыши на изображении исходного состояния на диаграмме и отпустить ее на изображении целевого состояния. В результате

этих действий на диаграмме появится изображение *перехода*, соединяющего два выбранных состояния. Продолжая разработку модели системы управления банкоматом, добавим на диаграмму *состояний* начальное состояние (**Start State**) и соединим его *переходом* с состоянием *Ожидание карточки* (рис. 9.5).



**Рис. 9.5.** Диаграмма состояний после добавления на нее перехода из начального состояния в состояние Ожидание карточки

После добавления *перехода* на диаграмму *состояний* можно открыть диалоговое окно его свойств и специфицировать дополнительные свойства, доступные на соответствующих вкладках (рис. 9.6). Следует обратить внимание на две первые строки вкладки **Detail** (Подробно), которые представляются наиболее важными из свойств *перехода*. Первое поле ввода **Guard Condition** служит для задания **сторожевого условия**, которое определяет правило срабатывания соответствующего *перехода*. Во втором поле ввода **Action** можно специфицировать *действие*, которое происходит при срабатывании *перехода* до того, как моделируемая система попадет в целевое состояние.



**Рис. 9.6.** Диалоговое окно спецификации свойств перехода, открытое на вкладке Detail (Подробно)

При необходимости можно определить сообщение о событии, происходящем при срабатывании *перехода*, а также визуализировать вложенность *состояний* и подключить историю отдельных *состояний*.

## Окончательное построение диаграммы состояний модели банкомата

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

1. Добавить *состояния* с именами: Ожидание ввода ПИН-кода, Проверка ПИН-кода, Ожидание выбора клиента, Обработка запроса на снятие наличных, Обработка запроса на получение справки, Выдача наличных, Печать, Возврат карточки, Завершение транзакции и финальное *состояние*.
2. Добавить *переход*: карточка вставлена, направленный от *состояния* Ожидание карточки к *состоянию* Ожидание ввода ПИН-кода.
3. Добавить *переход*: ПИН-код введен, направленный от *состояния* Ожидание ввода ПИН-кода к *состоянию* Проверка ПИН-кода.
4. Добавить *переход*: отмена транзакции, направленный от *состояния* Ожидание ввода ПИН-кода к *состоянию* Возврат карточки.
5. Добавить *переход* со сторожевым условием: [ПИН-код верный], направленный от *состояния* Проверка ПИН-кода к *состоянию* Ожидание выбора клиента.
6. Добавить *переход* со сторожевым условием: [ПИН-код неверный], направленный от *состояния* Проверка ПИН-кода к *состоянию* Ожидание ввода ПИН-кода.
7. Добавить *переход*: три неудачи с действием на переходе конфискация карточки, направленный от *состояния* Проверка ПИН-кода к *состоянию* Завершение транзакции. Для задания действия на данном переходе следует ввести текст

конфискация карточки в поле ввода **Action** (Действие) на вкладке **Detail** (Подробно) окна спецификации свойств данного *перехода* (рис. 9.7).

The image shows a dialog box titled "State Transition Specification" with two tabs: "General" and "Detail". The "Detail" tab is active. It contains several input fields: "Guard Condition:", "Action:" (containing "конфискация карточки"), "Send event:", "Send arguments:", and "Send target:". Below these is a section "Transition between substates" with "From" and "To" dropdown menus. "From" is set to "Проверка ПИН-кода" and "To" is set to "Завершение транзакции". At the bottom are buttons for "OK", "Cancel", "Apply", "Browse", and "Help".

**Рис. 9.7.** Диалоговое окно спецификации свойств перехода три неудачи при задании действия на переходе

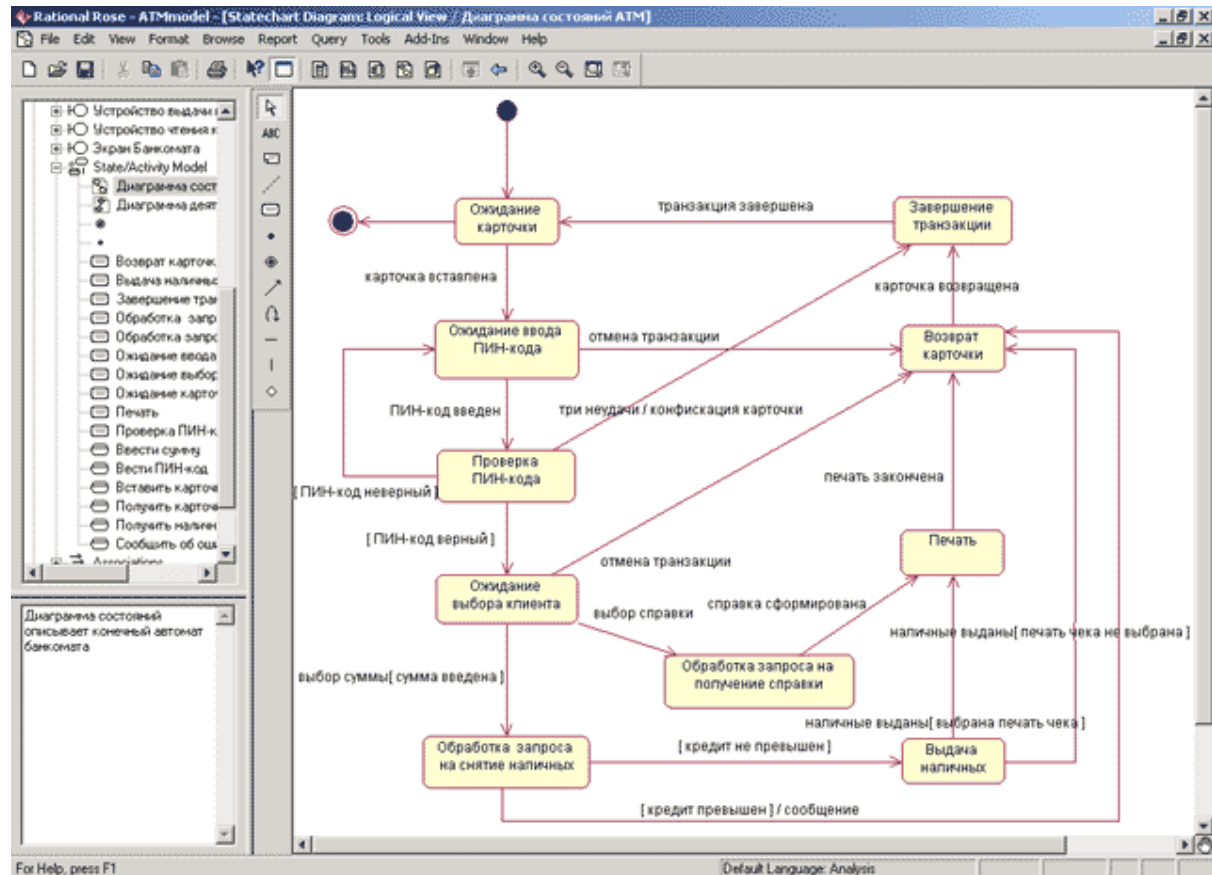
Для продолжения построения диаграммы *состояний* следует выполнить следующие действия:

8. Добавить *переход*: выбор суммы со *сторожевым условием*: [сумма введена], направленный от *состояния* Ожидание выбора клиента к *состоянию* Обработка запроса на снятие наличных.
9. Добавить *переход*: выбор справки, направленный от *состояния* Ожидание выбора клиента к *состоянию* Обработка запроса на получение справки.
10. Добавить *переход*: отмена транзакции, направленный от *состояния* Ожидание выбора клиента к *состоянию* Возврат карточки.
11. Добавить *переход* со *сторожевым условием*: [кредит не превышен], направленный от *состояния* Обработка запроса на снятие наличных к *состоянию* Выдача наличных.
12. Добавить *переход* со *сторожевым условием*: [кредит превышен] с действием на *переходе* сообщение, направленный от *состояния* Обработка запроса на снятие наличных к *состоянию* Возврат карточки.
13. Добавить *переход*: наличные выданы со *сторожевым условием*: [выбрана печать чека], направленный от *состояния* Выдача наличных к *состоянию* Печать.
14. Добавить *переход*: наличные выданы со *сторожевым условием*: [печать чека не выбрана], направленный от *состояния* Выдача наличных к *состоянию* Возврат карточки.
15. Добавить *переход*: справка сформирована, направленный от *состояния* Обработка запроса на получение справки к *состоянию* Печать.
16. Добавить *переход*: печать закончена, направленный от *состояния* Печать к *состоянию* Возврат карточки.



17. Добавить *переход*: *карточка возвращена*, направленный от *состояния Возврат карточки* к *состоянию Завершение транзакции*.
18. Добавить *переход*: *транзакция завершена*, направленный от *состояния Завершение транзакции* к *состоянию Ожидание карточки*.
19. Добавить *переход*, направленный от *состояния Ожидание карточки* к *финальному состоянию*.

Диаграмма *состояний* для рассматриваемой модели банкомата будет иметь следующий вид (рис. 9.8).



**Рис. 9.8.** Окончательный вид диаграммы состояний для моделирования поведения банкомата

Следует заметить, что в разрабатываемой модели диаграмма *состояний* является единственной и описывает поведение системы управления банкоматом в целом. Главное достоинство данной диаграммы *состояний* - возможность моделировать условный характер реализации всех вариантов использования в форме изменения отдельных *состояний* разрабатываемой системы. В то же время в среде IBM Rational Rose 2003 данная диаграмма не является необходимой для генерации программного кода. Поэтому в случае дублирования информации, представленной на диаграммах кооперации и последовательности, разработку диаграммы *состояний*, особенно в условиях дефицита времени, отпущенного на выполнение проекта, иногда опускают.