

Министерство образования и науки РФ

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ
(ТУСУР)**

Кафедра автоматизации обработки информации (АОИ)

УТВЕРЖДАЮ

Зав. каф. АОИ, профессор

Ю.П. Ехлаков

" ____ " _____ 2011 г.

**СОЗДАНИЕ МОДЕЛИ БИЗНЕС-ПРОЦЕССА
С ПОМОЩЬЮ ИНСТРУМЕНТАЛЬНОГО СРЕДСТВА
«RATIONAL ROSE»**

**Методические указания
для выполнения лабораторных работ
по дисциплине "Реинжиниринг бизнес-процессов"**

для студентов специальности 080504 –
Государственное и муниципальное управление

Разработчик:

профессор каф. АОИ, д.т.н.

_____ М.П. Силич

Томск 2011

СОДЕРЖАНИЕ

Введение	3
Лабораторная работа №1 «Знакомство с UML и Rational Rose. Создание диаграммы вариантов использования»	4
1.1. Выполнение работы на примере «Продажа продукта»	4
1.2. Самостоятельная работа	14
1.3. Выполнение индивидуального задания	14
Лабораторная работа №2 «Создание диаграммы деятельности»	15
2.1. Выполнение работы на примере «Продажа продукта»	15
2.2. Самостоятельная работа	19
2.3. Выполнение индивидуального задания	19
Лабораторная работа №3 «Создание диаграмм взаимодействия объектов»	20
3.1. Выполнение работы на примере «Продажа продукта»	20
3.2. Самостоятельная работа	27
3.3. Выполнение индивидуального задания	27
Лабораторная работа №4 «Создание диаграммы классов»	28
4.1. Выполнение работы на примере «Продажа продукта»	28
4.2. Самостоятельная работа	33
4.3. Выполнение индивидуального задания	33
Приложение. Варианты индивидуального задания	34

ВВЕДЕНИЕ

Лабораторные работы по дисциплине имеют целью получение навыков самостоятельного моделирования и анализа бизнес-процессов с помощью современных инструментальных средств.

Лабораторные работы выполняются в соответствии с методическими рекомендациями. Выполнение каждой работы включает в себя три этапа:

1. Изучение методического материала по теме работы и выполнение заданий, описанных в методических рекомендациях.
2. Самостоятельная работа по подготовке к выполнению индивидуального задания.
3. Выполнение индивидуального задания и защита работы.

Первый и третий этап выполняются в компьютерном классе во время, отведенное по расписанию для проведения лабораторных работ, второй этап (самостоятельная работа) выполняется в свободное время между аудиторными занятиями.

Самостоятельная работа состоит в сборе информации о конкретном бизнес-процессе, выбранном студентом, и составлении текстового описания. Список бизнес-процессов, из которого студент выбирает процесс для выполнения индивидуального задания, приведен в приложении.

Сбор информации о бизнес-процессе осуществляется на основе изучения литературных источников (книг, статей в журналах, в сборниках), публикаций в Интернете, законодательных, нормативно-правовых актов, знаний и опыта коллег и знакомых. Источники информации выбираются студентом самостоятельно.

Защита лабораторной работы выполняется путем демонстрации созданной модели и собеседования.

ЛАБОРАТОРНАЯ РАБОТА №1 ЗНАКОМСТВО С UML И RATIONAL ROSE. СОЗДАНИЕ ДИАГРАММЫ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ

1.1. Выполнение работы на примере «Продажа продукта»

Цель работы: Ознакомиться с основными возможностями языка моделирования UML и основами работы с CASE-средством Rational Rose. Создать диаграмму вариантов использования для бизнес-процесса «Продажа продукта» с помощью инструментального средства Rational Rose.

Порядок выполнения работы

1. Знакомство с основами языка моделирования UML

Унифицированный язык моделирования UML (Unified Modeling Language) предназначен для описания, визуализации и документирования бизнес-систем на базе объектно-ориентированного подхода с целью последующего использования моделей бизнес-процессов для реализации их в виде программного обеспечения.

Бурное развитие объектно-ориентированных языков программирования, сопровождающееся возрастанием сложности прикладных программ и стоимости их разработки, вызвало потребность в создании объектно-ориентированного языка для формирования предварительной модели предметной области, для которой разрабатывается программа. Такая модель необходима заказчиком, программистам и менеджерам проекта по созданию информационной системы для того, чтобы они могли выработать общий взгляд на цели и функции системы. И хотя модели предметной области, формируемые с помощью UML, предназначены, прежде всего, для последующей реализации в виде программного обеспечения, они имеют и самостоятельную ценность, т.к. позволяют наглядно отобразить функции и процессы бизнес-системы, объекты, участвующие в бизнес-процессах, их отношения, а также динамику выполнения процессов.

Начало работ над созданием унифицированного объектно-ориентированного языка моделирования относится к середине 1990-х годов. К тому времени уже было разработано более 50 различных языков объектно-ориентированного моделирования. Авторы наиболее распространенных языков – Г. Буч, Д. Румбах и А. Джекобсон, – собравшись «под крылом» компании Rational Software Corporation, начали работу над унифицированным методом. Ими был создан ряд вер-

сий унифицированного метода, который они назвали Unified Modeling Language (UML). В настоящее время большинством производителей информационных систем и такими комитетами по стандартам, как ANSI и OMG, язык UML был признан в качестве стандарта.

В технологии реинжиниринга бизнес-процессов, пожалуй, впервые UML стали применять не только и не столько для создания информационных систем (ИС), сколько для анализа и перепроектирования бизнеса. Вместо моделей процессов, реализуемых информационной системой, строятся модели бизнес-процессов, даже если они и не будут подвергнуты автоматизации, вместо объектов ИС (программных объектов) в моделях отражаются объекты бизнеса (исполнители, продукция, услуги и т.д.), вместо окружения ИС (пользователей ИС) моделируется окружение бизнеса (поставщики, партнеры, клиенты).

В рамках языка UML все представления о модели сложной системы фиксируются в виде специальных графических конструкций (схем, графов), получивших название диаграмм. Предполагается, что никакая единственная модель не может с достаточной степенью адекватности описывать различные аспекты сложной системы. Таким образом, модель сложной системы состоит из некоторого числа диаграмм, каждая из которых отражает некоторый аспект поведения или структуры системы. В языке UML определены следующие виды диаграмм:

- диаграмма вариантов использования (Use case diagram);
- диаграмма состояний (State diagram);
- диаграмма деятельности (Activity diagram);
- диаграмма последовательности (Sequence diagram);
- диаграмма кооперации (Collaboration diagram);
- диаграмма классов (Class diagram);
- диаграмма компонентов (Component diagram);
- диаграмма развертывания (Deployment diagram).

Диаграмма вариантов использования представляет собой наиболее общую концептуальную модель системы, которая является исходной для построения всех остальных диаграмм. Представление вариантов использования детализируется с помощью диаграмм состояний, деятельности, последовательности и кооперации.

Диаграммы классов используются для представления логической структуры информационной системы, диаграммы компонентов и диаграммы развертывания – для представления физических компонентов информационной системы.

2. Знакомство с основными возможностями Rational Rose

Для автоматизации проектирования и создания сложных информационных систем используются так называемые CASE-средства. Термин CASE расшифровывается как Computer-Aided Software Engineering – компьютерная поддержка проектирования программного обеспечения. Современные CASE-средства поддерживают весь жизненный цикл создания информационной системы – от создания графических моделей автоматизируемой предметной области (моделей бизнес-процессов) до проектирования баз данных и автоматического генерирования программного кода. Однако в настоящее время акцент стал смещаться с проектирования компонент ИС на анализ автоматизируемой предметной области, на моделирование сложных систем широкого назначения. Неслучайно аббревиатура CASE все чаще стала расшифровываться как Computer Aided System Engineering – компьютерная поддержка проектирования систем

Среди CASE-средств, ориентированных на объектно-ориентированные методы, наиболее популярным является средство Rational Rose компании Rational Software Corporation. Rational Rose позволяет строить объектные модели проектируемой системы в различных нотациях (UML, нотация Буча, метод ОМТ Румбаха) и генерировать на основе полученной модели приложения на языках программирования C++, Visual Basic, Power Builder, Java, Ada, Smalltalk и др. Rational Rose содержит все диаграммы UML. Модель, формируемая с помощью Rose, детально описывает, что система содержит и как функционирует.

3. Начало работы с Rational Rose

В рамках Rational Rose существуют различные программные инструментарии. Базовым средством, используемым для построения диаграмм, является Rational Rose Enterprise Edition. После запуска этой программы открывается главное окно, показанное на рис. 1.1.

В верхней части окна находится меню и стандартная панель инструментов (Tool Bar).

В левой части главного окна находится окно браузера (Browser), предназначенное для быстрого доступа к диаграммам. Любой элемент, который разработчик добавляет в модель, сразу отображается в этом окне. Оно позволяет легко перемещаться по дереву диаграмм, буксировать диаграммы мышкой и изменять структуру модели по своему усмотрению.

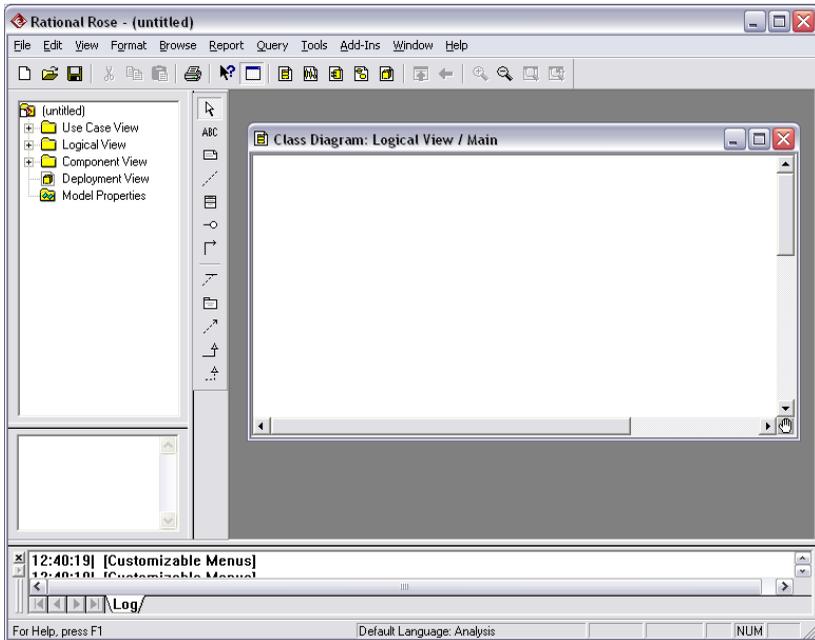


Рис. 1.1. Главное окно Rational Rose

В правой части главного окна находятся окна диаграмм (Diagram), открытых в текущий момент. Обычно это поле называется рабочим столом Rational Rose. Внизу рабочего стола находится окно протокола (Log). В нем Rational Rose постоянно фиксирует все действия, произведенные над диаграммами.

Между окном браузера и окном диаграммы находится панель инструментов текущей диаграммы (специальная панель), вид которой зависит от типа выбранной диаграммы.

В нижней левой части главного окна, под окном браузера находится окно документации (Documentation). В этом окне можно записывать самую различную информацию о выделенном в текущий момент элементе диаграммы.

При запуске программы кроме главного окна на экране появится диалоговое окно Create New Model. В нем будет отражен список «каркасных» моделей, которые можно взять за основу при создании новой модели. Поскольку мы будем строить модель "с нуля", нажмите кнопку Cancel в этом окне. Автоматически будет создан новый проект.

В окне браузера в виде дерева будет отражена структура проекта, включающая три представления:

представление вариантов использования (Use case view),
логическое представление (Logical View),
представление компонент (Component view).

Все эти представления отражены в виде папок (пакетов). Каждое из них, в свою очередь, содержит пустые диаграммы, созданные по умолчанию. Вы можете выбрать любую папку, «раскрыть» ее (щелчком мыши на квадратике с '+'), активизировать уже имеющиеся элементы (визуализировать в окне диаграммы), добавить новые диаграммы (через всплывающее меню, активизируемое щелчком правой кнопки мыши).

По окончании сеанса работы над проектом выполненную работу необходимо сохранить в файле проекта с расширением mdl. Это можно сделать через меню File→Save. В дальнейшем в начале нового сеанса можно открыть этот проект для последующей модификации через меню File→Open.

4. Основные понятия диаграммы вариантов использования (Use case diagram)

Представление вариантов использования показывает, как система должна выглядеть «извне», т.е. оно отражает функции системы и ее взаимодействие с внешним окружением. Основное внимание здесь уделяется представлению высокого уровня, отображающему, что система должна делать, а не как она будет делать это. Представление вариантов использования является исходным концептуальным представлением системы для последующей детализации.

Пример диаграммы вариантов использования приведен на рис. 1.2. Основными элементами диаграммы являются варианты использования (use case), которые отображаются в виде эллипса, и акторы (actor), которые отображаются в виде фигуры «человечка».

Вариантом использования или прецедентом называется законченная совокупность действий моделируемой системы, начинающаяся при получении стимула извне и заканчивающаяся предоставлением некоторого продукта или сервиса актору – пользователю системы. При моделировании информационной системы вариант использования соответствует отдельному сервису, предоставляемому ИС пользователю. При моделировании бизнеса варианты использования ставятся в соответствие бизнес-процессам.

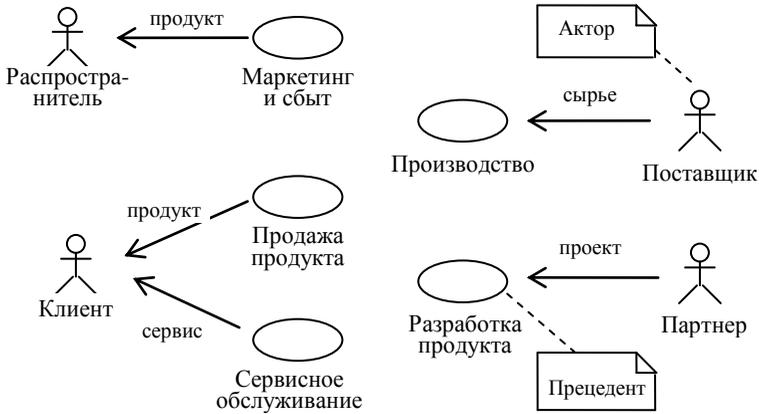


Рис. 1.2. Диаграмма вариантов использования

Актеры или действующие лица моделируют окружение системы. Это любые сущности, взаимодействующие с системой. Для информационных систем актерами являются пользователи, для бизнес-систем – клиенты, заказчики, поставщики, партнеры и т.д. Актеры взаимодействуют с системой через варианты использования.

Взаимосвязи между вариантами использования и актерами называются отношениями ассоциации. К ним, в частности относятся отношения коммуникации. При моделировании бизнеса они отражают материальные и информационные потоки, заключающиеся в передаче вещества (сырья, инструментов, продукции и т.д.) и информации.

Между вариантами использования отношения ассоциации недопустимы, т.к. каждый из них описывает самостоятельный законченный процесс. Однако между ними могут быть отношения обобщения, означающие, что один из процессов является частным случаем другого, более общего процесса. Отношения обобщения могут быть установлены и между актерами.

Кроме того, между вариантами использования могут устанавливаться отношения зависимости, означающие, что один из них некоторым образом зависит от другого. Разновидностями отношений зависимости являются отношение включения (include), означающее, что один процесс является частью другого, и отношение расширения (extend), означающее, что один процесс расширяет (дополняет) другой.

На диаграмму могут быть помещены также примечания, содержащие поясняющий текст для некоторого элемента диаграммы.

Таким образом, диаграмма вариантов использования может содержать следующие виды элементов:

- актер (Actor), который графически отображается в виде фигуры «человечка» под которой записывается имя актора;
- вариант использования (Use Case), который графически отображается в виде эллипса, под которым записывается имя.
- примечание (Note), которое может быть привязано к любому объекту диаграммы. Графически оно отображается в виде прямоугольника с «загнутым» правым уголком, внутри которого записывается текст примечания;
- отношение ассоциации (Association), которое графически отображается в виде сплошной линии со стрелкой на конце;
- отношение наследования или обобщения (Generalization), которое графически отображается в виде сплошной линии со стрелкой в виде незакрашенного треугольника;
- отношение зависимости или реализации (Dependency or instantiates), которое графически отображается в виде пунктирной линии со стрелкой на конце;
- якорь для замечания (Note Anchor), который позволяет соединить элемент Note с любым элементом на диаграмме. Графически отображается в виде пунктирной линии без стрелки.

Элементы на диаграмме могут располагаться на любом месте.

5. Построение Use case diagram

В окне браузера щелкните мышью на квадратике с '+' рядом с пакетом Use case view, чтобы раскрыть пакет представления вариантов использования, и дважды щелкните на пиктограмме Main (главная).

При этом на рабочем столе появится окно диаграммы вариантов использования. Специальная панель инструментов примет вид, соответствующий данному виду диаграмм. Назначение инструментов панели поясняется в таблице 1.1.

Прежде всего Вам необходимо поместить на диаграмму вариант использования (прецедент), соответствующий моделируемому бизнес-процессу, выданному в качестве индивидуального задания. Для этого необходимо выполнить следующие действия:

- на панели инструментов выберите инструмент Use Case;
- щелкните мышью в окне диаграммы на любом свободном месте.

Таблица 1.1

Инструмент	Наименование	Назначение
	Selection Tool	предоставляет возможность выделять объект
	Text Box	добавляет текст к диаграмме
	Note	добавляет к диаграмме примечание
	Anchor Note to Item	связывает примечание с объектом на диаграмме
	Package	помещает на диаграмму новый пакет
	Use case	помещает на диаграмму новый прецедент
	Actor	помещает на диаграмму нового актора
	Unidirectional Association	рисует направленную ассоциацию между актором и прецедентом
	Dependency or Instantiates	рисует отношение зависимости между элементами диаграммы
	Generalization	рисует отношение обобщения

На диаграмме появится изображение прецедента с маркерами изменения его геометрических размеров и предложенным программой именем по умолчанию. Вы можете поменять имя: щелкните клавишей мыши на выделенном элементе и введите новое имя (например, «Продажа») в поле под элементом. Переименовать можно и другим способом: щелкните на выделенном элементе двойным щелчком и в открывшемся диалоговом окне можно ввести новое имя в поле Name.

После того, как Вы создадите элемент, его можно перемещать, удалять, изменять размеры. В любой момент Вы можете активизировать любой из созданных элементов диаграммы (для этого на панели инструментов должен быть активен инструмент выбора Selection Tool) и производить с ним дальнейшие действия: перемещать с помощью мыши; «растягивать», потянув за маркер; удалять с помощью клавиши Delete или через всплывающее меню Edit/Delete.

Все создаваемые Вами элементы будут отображаться также в окне браузера (будут помещены в дерево диаграммы Use case View/Main).

Если моделируемый бизнес-процесс существует в нескольких версиях, то каждой из них нужно сопоставить отдельный прецедент. Например, прецедент «Продажа» имеет две версии: «Продажа готового продукта» и «Продажа заказного продукта». В этом случае на диаграмму нужно поместить соответствующие прецеденты и установить отношения обобщения между прецедентом «Продажа» и каждым из прецедентов-версий (рис. 1.3). Для этого:

- на панели инструментов выберите инструмент Generalization;
- не отпуская кнопки мыши, перемещайте указатель от одного элемента (прецедента-версии) к другому (обобщенному прецеденту).

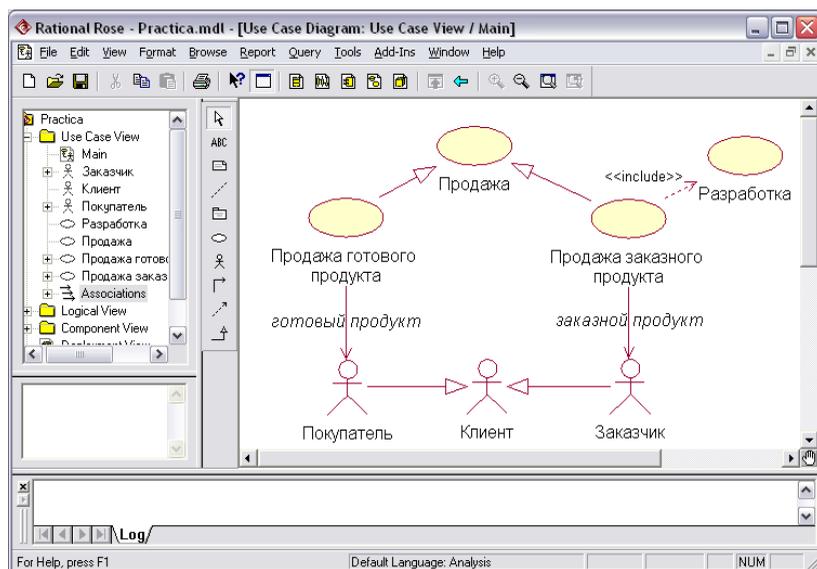


Рис. 1.3. Диаграмма вариантов использования в среде Rational Rose

Если некоторый бизнес-процесс содержит относительно самостоятельный фрагмент, то его можно выделить в самостоятельный прецедент, связанный с исходным прецедентом отношением включения. Например, из прецедента «Продажа заказного продукта» можно выделить фрагмент «Разработка» и создать соответствующий прецедент (см. рис. 1.3). Чтобы установить между прецедентами отношение включения нужно выполнить следующее:

- выберите инструмент Dependency or Instantiates (отношение зависимости),

- не отпуская кнопки мыши, перемещайте указатель от исходного прецедента к прецеденту-фрагменту;
- щелкните двойным щелчком на выделенном отношении зависимости;
- в появившемся диалоговом окне в поле Stereotype впишите или выберите из выпадающего списка стереотип include и закройте окно по кнопке ОК.

Обязательными элементами диаграммы Use case являются также акторы. Чтобы создать актора выберите инструмент Actor и щелкните мышью в том месте диаграммы, куда Вы хотите его поместить. Имя актора задается так же, как имя варианта использования.

Чтобы установить отношение ассоциации между актором и прецедентом выберите инструмент Unidirectional Association и, не отпуская кнопки мыши, перемещайте указатель от одного элемента к другому. Щелкните двойным щелчком на выделенной ассоциации и в появившемся диалоговом окне введите имя ассоциации в поле Name.

На диаграмме может быть размещено несколько акторов. Например, с прецедентом «Продажа готового продукта» взаимодействует актор «Покупатель», а с прецедентом «Продажа заказного продукта» – актор «Заказчик» (см. рис. 1.3). Можно добавить обобщенный актор (в данном случае «Клиент») и установить отношения обобщения между акторами.

6. Документирование

Последним этапом создания диаграммы является документирование объектов диаграммы. Документация (поясняющий текст) вносится в окне Documentation, находящемся под окном браузера. Выделите некоторый объект диаграммы (вариант использования, актор, отношение) и в окне Documentation введите поясняющий текст. Каждый раз, когда Вы будете выделять этот объект, в окне документации будет отображаться введенный Вами текст. Документацию для любого объекта можно также внести в окне спецификации, вызываемом двойным щелчком мыши на выделенном объекте.

Вы можете также разместить на диаграмме примечания с помощью инструмента Note и привязать его к некоторому элементу с помощью инструмента Anchor Note to Item. Кроме того, Вы можете создать произвольную надпись на диаграмме, не привязанную ни к какому элементу. Это можно сделать с помощью инструмента Text Box.

Сохраните проект с помощью меню File→Save.

1.2. Самостоятельная работа

1. Выберите бизнес-процесс, для которого будет разрабатываться модель на языке UML из списка, приведенного в приложении или предложите свой собственный вариант.

2. Соберите информацию о выбранном бизнес-процессе, в том числе:

- о входах и выходах процесса, т.е. о необходимых для выполнения процесса материалах, сырье, информации и о результатах выполнения процесса – продукции, услугах, документах;

- об окружении процесса, т.е. о субъектах, взаимодействующих с процессом – получающих его выходы или передающих что-либо на его вход.

3. Определите, каким образом можно структурировать процесс. Например, можно ли выделить различные версии бизнес-процесса, выполняемые в различных условиях; можно ли выделить крупные фрагменты процесса, выполняемые лишь при определенных условиях; можно ли выделить подпроцессы, расширяющие процесс. Можете также выделить обобщенный бизнес-процесс и обобщенные субъекты окружения.

4. На основе собранной информации составьте текстовое описание процесса (различных версий процесса, крупных фрагментов), его окружения, входов и выходов. Составьте рукописную Use case - диаграмму.

1.3. Выполнение индивидуального задания

1. Создайте новый проект, задав имя файла, в котором будет храниться новая модель, формируемая по индивидуальному заданию.

2. Создайте диаграмму Use case. На диаграмме должны быть: прецеденты, соответствующие выбранному процессу, его подпроцессам, его версиям; акторы, взаимодействующие с процессом; связи (отношения коммуникации) акторов с прецедентами; структурирующие отношения (обобщения, включения, расширения) между прецедентами или между акторами.

3. Документируйте основные объекты созданной диаграммы.

ЛАБОРАТОРНАЯ РАБОТА №2 СОЗДАНИЕ ДИАГРАММЫ ДЕЯТЕЛЬНОСТИ

2.1. Выполнение работы на примере «Продажа продукта»

Цель работы: Создать диаграмму деятельности (Activity diagram) для бизнес-процесса «Продажа продукта» с помощью инструментального средства Rational Rose.

Порядок выполнения работы

1. Основные понятия диаграммы деятельности (Activity diagram)

Данный тип диаграмм может использоваться для отражения последовательности действий (элементарных операций) во время выполнения некоторого бизнес-процесса, представленного на диаграмме вариантов использования. Традиционно для этой цели использовались блок-схемы или структурные схемы алгоритмов.

Пример диаграммы деятельности показан на рис. 2.1.

Графически диаграмма деятельности представляется в форме графа, вершинами которого являются действия (операции, шаги процесса), а дугами – переходы от одного действия к другому. На диаграмме можно также отразить ветвление, т.е. возможность перехода к различным действиям в зависимости от некоторых условий.

Начальное состояние (Start State), соответствующее началу процесса, обозначается в виде закрашенного кружка; конечное состояние (End State), соответствующее завершению процесса – в виде закрашенного кружка, помещенного в окружность.

Действие (activity) изображается фигурой, напоминающей прямоугольник с закругленными сторонами, внутри которого записывается выражение действия.

Переход (Transition) изображается сплошной линией со стрелкой.

Если после выполнения некоторого действия процесс должен разделиться на альтернативные ветви в зависимости от некоторого условия, то ставится знак ветвления (Desizion) в виде ромба, внутри которого нет никакого текста. В него может входить только одна стрелка. Выходящих стрелок может быть две или более. Для каждой из них указывается соответствующее условие, при котором выполняется данный переход.



Рис. 2.1. Диаграмма деятельности прецедента «Продажа»

2. Построение Activity diagram

Так как диаграмма последовательности раскрывает последовательность действий (событий) при выполнении некоторого варианта использования, то будем создавать ее как поддиаграмму (Sub Diagram) варианта использования.

Откройте проект, созданный во время выполнения практической работы №1, через меню File→Open. В окне браузера раскройте пакет представления вариантов использования (Use case view) и активизируйте диаграмму Main.

В окне диаграммы выберите вариант использования, для которого будете создавать диаграмму деятельности. Установите курсор на этот вариант и активизируйте всплывающее меню щелчком правой кнопки мыши. Выберите в меню Sub Diagrams → New Activity Diagram.

На рабочем столе появится пустое окно диаграммы деятельности. Специальная панель инструментов примет вид, соответствующий данному виду диаграмм. Назначение инструментов панели поясняется в таблице 2.1.

Таблица 2.1

Инструмент	Наименование	Назначение
	Selection Tool	предоставляет возможность выделять объект
	Text Box	добавляет текст к диаграмме
	Note	добавляет к диаграмме примечание
	Anchor Note to Item	связывает примечание с объектом на диаграмме
	State	помещает на диаграмму новое состояние
	Activity	помещает на диаграмму новое действие
	Start State	помещает на диаграмму новое начальное состояние
	End State	помещает на диаграмму новое конечное состояние
	Tranzition	рисует переход
	Tranzition to Self	рисует рефлексивный переход
	Horizontal Synchronization	помещает на диаграмму новый символ горизонтальной синхронизации
	Vertical Synchronization	помещает на диаграмму новый символ вертикальной синхронизации
	Desizion	помещает на диаграмму новое ветвление
	SwimLane	помещает на диаграмму новую дорожку

Для того, чтобы поместить на диаграмму некоторый элемент (начальное или конечное состояние, действие, ветвление), необходимо выполнить следующие действия:

- на панели инструментов выберите нужный инструмент (Start State, End State, Activity, Desizion);

- щелкните мышью в окне диаграммы в том месте, куда Вы хотите поместить элемент.

На диаграмме появится изображение элемента с маркерами изменения его геометрических размеров. Вы можете перемещать элемент, менять его размеры, удалить.

Внутри элемента Activity нужно ввести описание действия.

Для того, чтобы установить переход между элементами диаграммы, на панели инструментов выберите инструмент State Transition, затем, не отпуская кнопки мыши, перемещайте указатель от одного элемента к другому.

Стрелки перехода, выходящие из ветвления (Desizion), должны сопровождаться текстом, поясняющим условия, при которых выполняется тот или иной переход. Чтобы поместить текст возле стрелки (условие перехода), щелкните на выделенной стрелке двойным щелчком и в открывшемся диалоговом окне в поле Event введите текст.

Можно поместить на диаграмму, так называемые дорожки (SwimLanes). Имеется в виду визуальная аналогия с плавательными дорожками в бассейне. Поле диаграммы разделено на «дорожки», соответствующие различным исполнителям (или подразделениям компании). При этом все действия, выполняемые определенным исполнителем (подразделением), помещаются на соответствующую дорожку (см. рис. 2.2).

Чтобы поместить на диаграмму дорожку выберите на панели инструментов инструмент SwimLane и щелкните мышью в окне диаграммы. Затем введите наименование дорожки в ее верхней части. Переместите на дорожку те действия, которые выполняются соответствующим исполнителем. Добавьте столько дорожек, сколько необходимо.

Еще одна полезная возможность – синхронизация действий, т.е. возможность показать параллельные потоки действий. Для этого используется специальный символ в виде отрезка горизонтальной или вертикальной линии, который означает либо разделение на параллельные потоки (в этом случае в него входит один переход, а выходит – несколько), либо слияние параллельных потоков (в этом случае в него входят несколько переходов, а выходит – один).

На диаграмме, представленной на рис. 2.2, после выполнения действия «Принять заказ» происходит разделение на два параллельных потока, содержащие соответственно действия «Принять оплату» и «Изготовить продукт». А потом эти потоки сливаются.

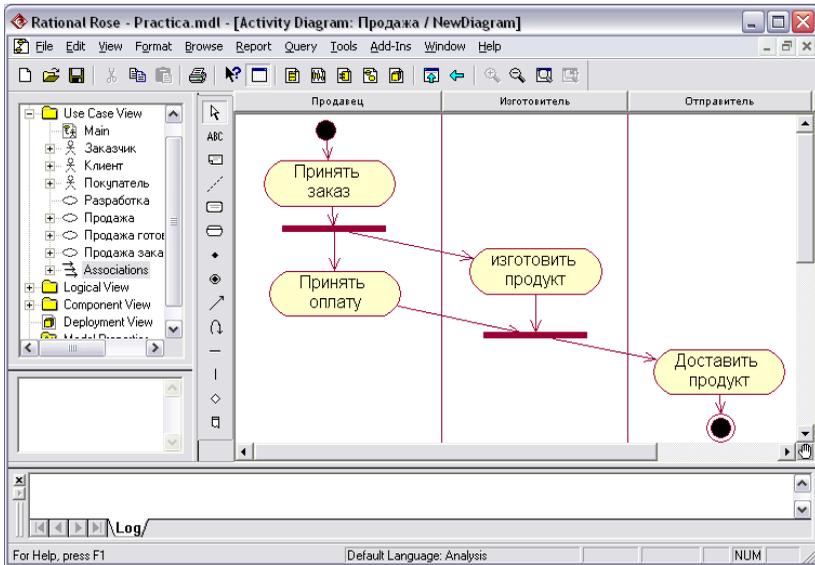


Рис. 2.2. Диаграмма деятельности в среде Rational Rose

Чтобы поместить на диаграмму символ синхронизации выберите инструмент Horizontal Synchronization или Vertical Synchronization и щелкните мышью в окне диаграммы.

В заключение можете документировать те действия, которые нуждаются в подробном объяснении.

Сохраните проект.

2.2. Самостоятельная работа

1. Соберите информацию о ходе выполнения выбранного бизнес-процесса, его отдельных версий и фрагментов.
2. Составьте рукописные диаграммы деятельности.

2.3. Выполнение индивидуального задания

1. Откройте файл, в котором хранится модель, созданная на предыдущей лабораторной работе по индивидуальному заданию.
2. Создайте диаграммы деятельности для каждого из прецедентов, представленных на диаграмме Use case.

ЛАБОРАТОРНАЯ РАБОТА №3 СОЗДАНИЕ ДИАГРАММ ВЗАИМОДЕЙСТВИЯ ОБЪЕКТОВ

3.1. Выполнение работы на примере «Продажа продукта»

Цель работы: Создать диаграммы последовательности взаимодействия объектов (Sequence diagram) и диаграммы кооперации (Collaboration diagram) для бизнес-процесса «Продажа продукта» с помощью инструментального средства Rational Rose.

Порядок выполнения работы

1. Основные понятия диаграммы последовательности (Sequence diagram)

Диаграмма вариантов использования показывает функции системы, диаграммы деятельности показывают в деталях ход выполнения процессов. Однако для полного понимания системы такого описания недостаточно. Необходима модель, показывающая как, за счет чего реализуются процессы, т.е. какие объекты участвуют в выполнении процесса и как они взаимодействуют между собой и с окружением системы. Диаграммы взаимодействия, к которым относятся диаграмма последовательности и диаграмма кооперации, как раз и отражают взаимодействие объектов. Диаграмма последовательности позволяет отразить динамику взаимодействий между участниками процесса, а диаграмма кооперации отражает статический взгляд на взаимодействие, абстрагируясь от последовательности действий.

Взаимодействие объектов в UML рассматривается, прежде всего, как коммуникация, т.е. передача сообщений (информации). Однако взаимодействие можно рассматривать и как передачу вещества (продукции, деталей, материалов), понимая под сообщением не только информацию, но и передачу некоторого объекта-сущности.

Передача сообщения осуществляется от объекта-инициатора к другому объекту. При этом сообщение инициирует выполнение определенных действий тем объектом, которому это сообщение отправлено. Поэтому объект, получающий сообщение, называют иногда исполнителем. При этом в разных ситуациях одни и те же объекты могут выступать и в качестве инициаторов, и в качестве исполнителей. Кроме того, объект может отправлять сообщение самому себе, т.е. он одновременно является и отправителем, и получателем сообщения. Объектом, участвующим во взаимодействии может быть и актер.

Пример диаграммы последовательности приведен на рис. 3.1.

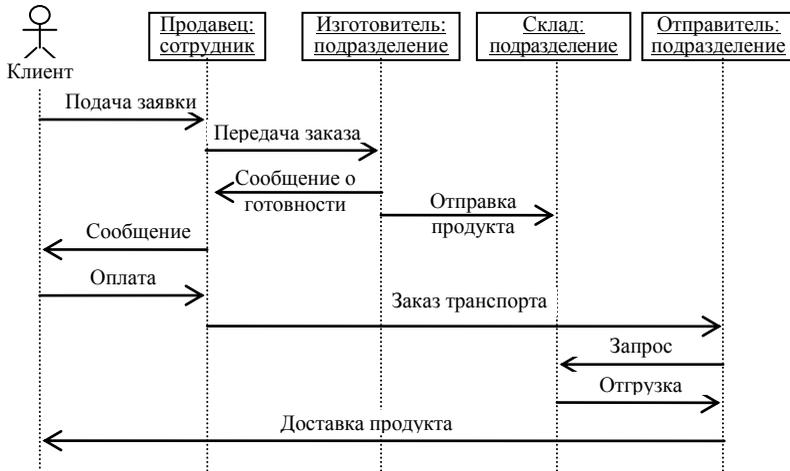


Рис. 3.1. Диаграмма последовательности прецедента «Продажа заказного продукта»

Графически каждый объект (Object), передающий или получающий сообщение, располагается в верхней части диаграммы и изображается в виде прямоугольника, от которого вниз идет пунктирная линия, называемая линией жизни объекта (см. рис. 3.1). Внутри прямоугольника записывается имя объекта и через двоеточие имя класса, к которому принадлежит объект (имя подчеркивается, что является признаком объекта).

Передача сообщения (Object Message) изображается в виде горизонтальной стрелки с именем сообщения от линии жизни объекта-инициатора к линии жизни объекта-получателя. Последовательность взаимодействий отражается через последовательное их расположение на диаграмме сверху - вниз. Другими словами, сообщения, расположенные на диаграмме выше, инициируются раньше тех, которые расположены ниже. Объекты тоже следует располагать в порядке их активности: крайним слева изображается объект, который является инициатором первого сообщения, правее – получатель сообщения и т.д.

В процессе функционирования системы одни объекты могут находиться в активном состоянии, непосредственно выполняя определенные действия или в состоянии пассивного ожидания сообщений от других объектов.

Чтобы явно выделить активность объектов, в UML применяется понятие «фокус управления». Фокус управления изображается в форме вытянутого узкого прямоугольника вдоль линии жизни, верхняя сторона которого обозначает начало активности, а нижняя – окончание активности.

На диаграмме могут также располагаться комментарии или примечания (Notes), которые могут ассоциироваться с отдельными объектами или сообщениями.

2. Построение Sequence diagram

Так как диаграмма последовательности раскрывает последовательность взаимодействия объектов при выполнении одного из вариантов использования, то будем создавать ее в пакете представления вариантов. В окне браузера установите курсор на соответствующем прецеденте и откройте всплывающее меню щелчком правой кнопки мыши. Выберите пункт New → Sequence Diagram (см. рис. 3.2). Дайте новой диаграмме последовательности имя. Дважды щелкнув на диаграмме в браузере, откройте ее.

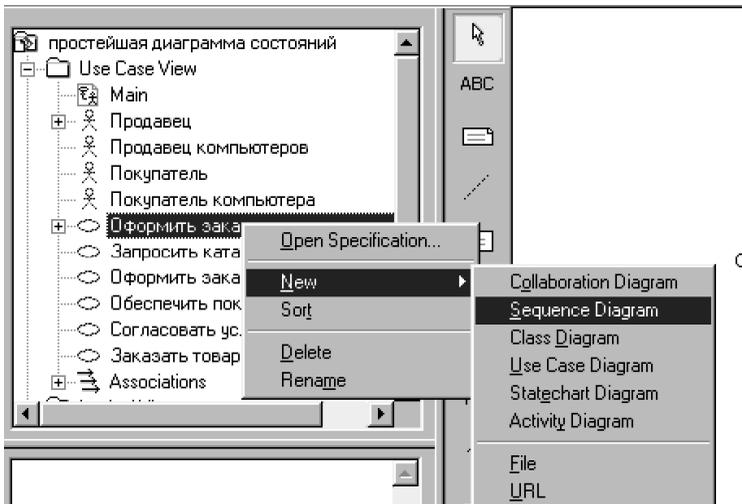


Рис. 3.2. Создание диаграммы последовательности

На рабочем столе появится окно диаграммы последовательности. Специальная панель инструментов примет вид, соответствующий данному виду диаграмм. Назначение инструментов панели поясняется в таблице 3.1.

Таблица 4.1.

Инструмент	Наименование	Назначение
	Selects or deselects an item	предоставляет возможность выделять объект
	Text Box	добавляет к диаграмме текст
	Note	добавляет к диаграмме примечание
	Anchor Note to Item	связывает примечание с элементом на диаграмме
	Object	помещает на диаграмму новый объект
	Object Message	рисует сообщение между двумя объектами
	Message to Self	Рисует рефлексивное сообщение (сообщение самому себе)

Пример диаграммы последовательности, построенной в среде Rational Rose, приведен на рис. 3.3.

Первым этапом создания диаграммы последовательности является выделение объектов-участников процесса.

Если инициатором сообщения является актер, его следует перенести с диаграммы вариантов использования, построенной в ходе выполнения практической работы №1. Для этого в окне браузера раскройте диаграмму Main, чтобы показать элементы, включенные в эту диаграмму. Выберите нужный элемент (актера) и отбуксируйте его (оставляя нажатой левую кнопку мыши) в окно диаграммы. Расположите его в верхней левой части диаграммы.

Чтобы создать объект (получателя или инициатора сообщений), на панели инструментов выберите инструмент Object и щелкните мышью в верхней части окна диаграммы правее уже помещенных объектов. Задать имя объекта можно двумя способами:

- щелкните на выделенном объекте и введите имя внутри прямоугольника, обозначающего объект;
- щелкните двойным щелчком на выделенном объекте и в появившемся диалоговом окне введите имя в поле Name.

Кроме того, в диалоговом окне Вы можете задать класс объекта. Для этого в поле Class в выпадающем списке выберите подходящее имя класса или выберите <New> и в появившемся диалоговом окне введите имя класса.

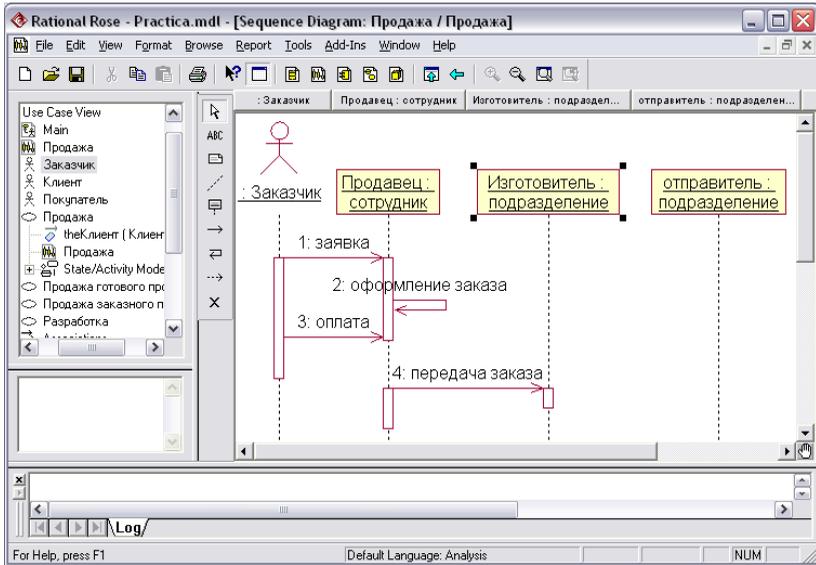


Рис. 3.3. Диаграмма последовательности в среде Rational Rose

В процессе работы с диаграммой Вам может понадобиться удалить некоторые объекты. При этом Rational Rose автоматически удалит также все сообщения, которые начинаются или заканчиваются на объекте, и переименует оставшиеся сообщения.

Для того, чтобы отобразить взаимодействие между объектами, необходимо выполнить следующие действия:

- на панели инструментов выберите инструмент Message (сообщение);
- установите курсор на линии жизни объекта – инициатора сообщения в нужном месте, соответствующем последовательности передачи сообщения;
- не отпуская кнопки мыши, перемещайте указатель к линии жизни объекта-получателя сообщений.

Чтобы задать имя сообщения, щелкните двойным щелчком на выделенной линии сообщения и в открывшемся диалоговом окне вве-

дите имя. В дальнейшем можно переименовать сообщение: щелкнуть на нем мышью и ввести имя в поле над линией сообщения.

Чтобы отобразить сообщение, посылаемое объектом самому себе, нужно выбрать на панели инструментов инструмент Message to Self и щелкнуть мышью на линии жизни объекта в нужном месте, соответствующем последовательности передачи сообщения.

При необходимости можно изменить порядок следования сообщений. Для этого достаточно перетащить сообщение на новое место. При изменении порядка следования сообщений они автоматически перенумеровываются.

В процессе работы с диаграммой последовательности может потребоваться удалить ранее нарисованные сообщения. При этом оставшиеся сообщения будут автоматически перенумерованы.

Дополнительно можно поместить на диаграмму примечания (комментарии) и связать их с сообщениями. Вы можете также разместить на диаграмме произвольный текст (с помощью инструмента Text Box).

Последним этапом создания диаграммы является документирование элементов диаграммы (объектов и сообщений). Документация (поясняющий текст) на активный элемент вносится в окне Documentation.

Создайте диаграмму последовательности действий в соответствии с индивидуальным заданием, выданным преподавателем.

Сохраните проект.

3. Построение Collaboration diagram

Диаграмма кооперации, как и диаграмма последовательности, предназначена для визуализации взаимодействия участников процесса. Если диаграмма последовательности служит для отображения временных аспектов взаимодействия, то диаграмма кооперации акцентирует внимание на структурных аспектах взаимодействия объектов. Это статическая модель процесса, т. е. является мгновенным снимком объектов системы в некотором состоянии.

Так как диаграммы последовательности (Sequence) и кооперации (Collaboration) являются разными взглядами на одни и те же процессы, Rational Rose позволяет создавать из Sequence- диаграммы диаграмму Collaboration и наоборот, а также производит автоматическую синхронизацию этих диаграмм.

Находясь в окне с диаграммой последовательности, нажмите клавишу <F5>. Диаграмма кооперации будет создана автоматически на

основе диаграммы последовательности. В дальнейшем с помощью клавиши <F5> Вы можете переключаться с одной диаграммы на другую. При этом изменения, вносимые в одну из диаграмм, будут автоматически вноситься и на другую диаграмму.

Пример диаграммы кооперации, построенной в среде Rational Rose, приведен на рис. 3.4.

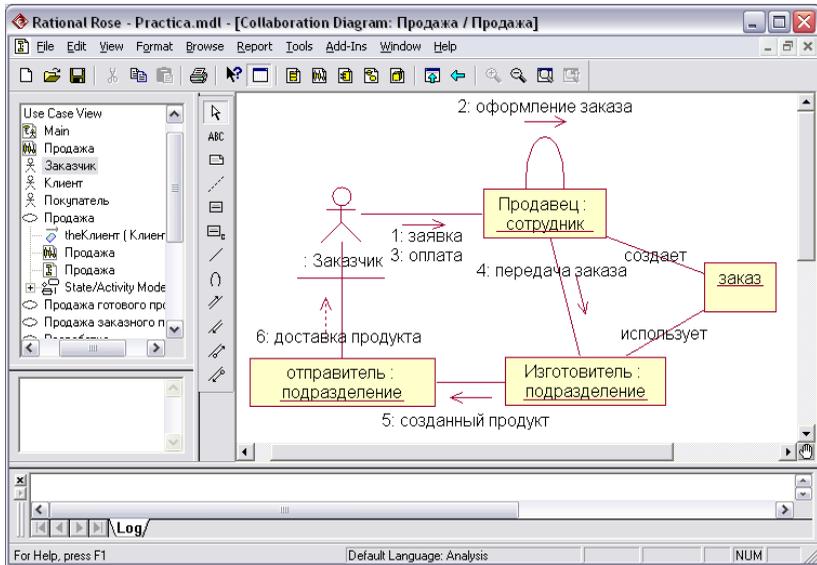


Рис. 3.4. Диаграмма кооперации в среде Rational Rose

В окне диаграммы будут отражены объекты и сообщения, перенесенные с диаграммы последовательности. Можете активизировать любой объект и передвинуть его, чтобы диаграмма приобрела более красивый вид.

Можно поместить на диаграмму новые объекты, в частности объекты-сущности. Для этого необходимо выполнить следующие действия:

- на панели инструментов выберите инструмент Object;
- щелкните мышью в окне диаграммы на любом свободном месте;
- введите имя объекта (класса).

В отличие от диаграммы последовательности на диаграмме кооперации не указывается время в виде отдельного измерения. Поэтому

последовательность взаимодействий может быть определена только с помощью порядковых номеров.

Кроме динамических связей (сообщений) на диаграмме кооперации могут быть показаны и разнообразные статические связи (Link), которые изображаются отрезком прямой линии, соединяющей два прямоугольника объектов (без стрелки). Рядом с линией может записываться имя соответствующей связи.

Для того, чтобы установить статическую связь между элементами диаграммы, необходимо выполнить следующие действия:

- на панели инструментов выберите инструмент Object Link,
- не отпуская кнопки мыши, перемещайте указатель от одного элемента к другому.

Чтобы задать имя связи, щелкните на выделенной связи двойным щелчком и в открывшемся диалоговом окне введите имя.

Вы можете также разместить на диаграмме примечания с помощью инструмента Note и привязать его к некоторому элементу с помощью инструмента Anchor Note to Item или поместить произвольный текст с помощью инструмента Text Box.

Сохраните проект.

3.2. Самостоятельная работа

1. Соберите информацию о том, какие объекты участвуют в выполнении выбранного вами бизнес-процесса, а также о том, каким именно образом они взаимодействуют в ходе выполнения процесса.

2. Создайте рукописные диаграммы последовательности и кооперации для бизнес-процесса (его отдельных версий и фрагментов).

3.3. Выполнение индивидуального задания

1. Откройте файл, в котором хранится модель, созданная на лабораторных работах №1 и №2 по индивидуальному заданию.

2. Создайте диаграммы последовательности для каждого из прецедентов, представленных на диаграмме Use case.

3. Преобразуйте диаграммы последовательности в диаграммы кооперации.

ЛАБОРАТОРНАЯ РАБОТА №4 СОЗДАНИЕ ДИАГРАММЫ КЛАССОВ

4.1. Выполнение работы на примере «Продажа продукта»

Цель работы: Создать диаграмму классов (Class diagram) для бизнес-процесса «Продажа продукта» с помощью инструментального средства Rational Rose.

Порядок выполнения работы

1. Основные понятия диаграммы классов (Class diagram)

Класс — это структура описания (набор характеристик) и шаблон поведения (набор операций) для некоторого множества реальных объектов, которые определяются на основе данного шаблона. Класс — это некоторая абстракция реального мира. Когда эта абстракция принимает конкретное воплощение, она называется объектом. Например, класс «Продавец» имеет набор характеристик (атрибутов), таких как «фамилия, имя, отчество», «стаж работы», «рабочее место» и т.д., а также набор операций, таких как «прием заявки клиента», «оформление заказа» и т.д. Описание конкретного продавца создается на базе данного класса и содержит конкретные значения атрибутов, характерные именно для этого продавца.

Класс изображается как прямоугольник, разделенный на три части (рис. 4.1, *а*): в верхней части записывается название класса, в середине — атрибуты, в нижней части — операции. Однако класс может иметь и другое изображение, связанное с категорией класса. В модели бизнеса используются две основных категории классов: business worker (исполнитель) и business entity (сущность). Обозначения этих категорий классов приведены на рис. 4.1, *б* и рис. 4.1, *в*.

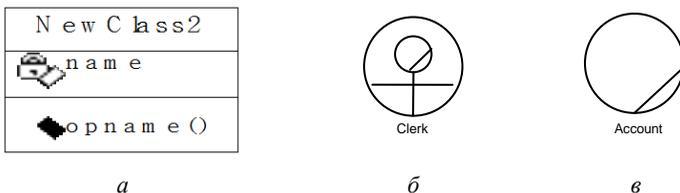


Рис.4.1 Обозначения классов

Диаграмма классов позволяет отразить не только классы объектов, участвующих в выполнении бизнес-процесса, но и отношения между классами.

На рис. 4.2 представлен пример диаграммы классов, построенной для прецедента «Продажа продукта».

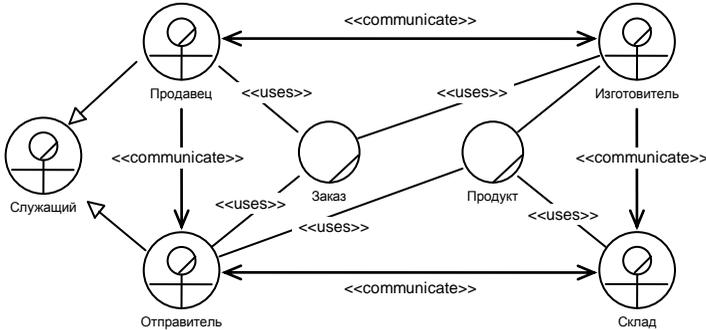


Рис. 4.2. Диаграмма классов для прецедента «Продажа продукта»

На диаграмме отражены классы исполнителей, выполняющих прецедент (Продавец, Исполнитель, Склад, Отправитель), а также классы объектов-сущностей, используемых в ходе выполнения прецедента (Заказ, Продукт). Между классами исполнителей установлены отношения *коммуникации* (ассоциации со стереотипом *communicate*), отражающие их взаимодействие. Между классами объектов-сущностей, как правило, отношения коммуникации не устанавливаются. Класс сущности может быть связан с классом исполнителя отношением *использования* (ассоциации со стереотипом *uses*) в случае, если исполнитель некоторым образом использует сущность. Например, Продавец создает Заказ, Изготовитель использует Заказ для получения описания продукта, Отправитель использует Заказ для получения информации о том, куда доставлять продукт. Соответствующие отношения использования представлены на рис. 4.2.

На диаграмме классов могут быть отражены также отношения структурирования – обобщения и включения. Так, на рис. 4.2 показаны отношения обобщения между абстрактным классом Служащий и более конкретными классами Продавец, Отправитель.

2. Построение Class diagram

Создайте новую диаграмму классов для прецедента «Продажа заказного продукта» аналогично тому, как вы создавали диаграмму последовательности (см. п.2 практической работы №3, рис. 3.2), только во всплывающем меню выберите пункт New → Class Diagram. Дважды щелкнув на диаграмме в браузере, откройте ее.

На рабочем столе появится окно диаграммы классов. Специальная панель инструментов примет вид, соответствующий данному виду диаграмм. Назначение некоторых инструментов панели поясняется в таблице 4.1.

Таблица 4.1.

Пиктограмма	Наименование	Назначение
	Class (Класс)	Добавляет на диаграмму новый класс
	Interface (Интерфейс)	Добавляет на диаграмму новый интерфейсный класс
	Association (Ассоциация)	Добавляет ненаправленную ассоциацию
	Aggregation (Агрегация)	Добавляет отношение агрегации
	Link Attribute (Атрибут отношения)	Связывает класс с отношением ассоциации
	Package (Пакет)	Добавляет на диаграмму новый пакет
	Dependency or instantiates (Зависимость или наполнение)	Добавляет отношение зависимости
	Generalization (Обобщение)	Добавляет отношение обобщения
	Realize (Реализация)	Добавляет отношение реализации
	Unidirectional Association (Однонаправленная ассоциация)	Добавляет однонаправленную ассоциацию

Чтобы создать класс, на панели инструментов выберите инструмент Class и щелкните мышью в любом месте окна диаграммы. Будет создан новый класс с именем NewClass. Замените это имя на новое, например, «Продавец».

Вы можете переименовать класс, задать его стереотип, добавить атрибуты и операции в окне спецификации класса. Для вызова окна щелкните двойным щелчком на выделенном классе. Пример окна спецификации класса приведен на рис. 4.3.

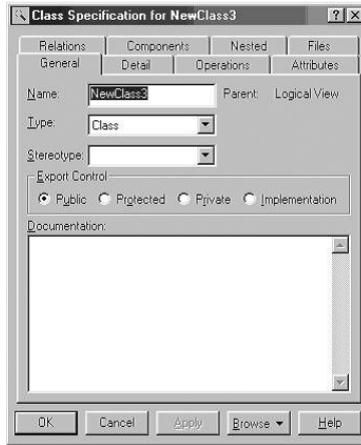


Рис. 4.3. Окно спецификации класса

Задайте стереотип класса, выбрав его из выпадающего списка в поле *Stereotype*. Например, для класса «Продавец» подойдет стереотип *business worker*. Щелкните кнопку **OK**. Вы увидите, что изображение класса поменялось.

Для добавления атрибута к классу щелкните правой кнопкой мыши на классе диаграммы. В открывшемся меню выберите пункт **New Attribute (Новый Атрибут)**. Введите имя атрибута в формате **Имя : Тип данных = Начальное значение**. Начальное значение необязательно.

Можно ввести атрибуты в окне спецификации. Откройте окно спецификации класса. Перейдите на вкладку **Attributes (Атрибуты)**. Если у класса уже имеются атрибуты, они будут перечислены на этой вкладке. Щелкните правой кнопкой мыши где-нибудь внутри области атрибутов. В открывшемся меню выберите пункт **Insert (Вставить)**. Введите имя нового атрибута, тип данных и значение по умолчанию в соответствующих колонках.

Операции класса вводятся аналогично. Перейдите на вкладку **Operations (Операции)** окна спецификации. Щелкните правой кнопкой мыши где-нибудь внутри области операций. В открывшемся меню выберите пункт **Insert (Вставить)**. Введите имя новой операции в колонке **Operation**.

Идентифицируя операции, изучите все сообщения на диаграммах последовательности и кооперации. Анализ этих диаграмм помогает выявить все операции активного объекта. Так, из диаграммы, пред-

ставленной на рис. 3.1, можно определить, что к обязательствам объекта Продавец относятся: Прием заявки, Оформление заказа, Передача заказа Изготовителю, Прием сообщения о готовности продукта, Сообщение клиенту о готовности продукта, Прием оплаты, Заказ транспорта. Данные операции должны быть внесены в спецификацию соответствующего класса Продавец.

Следующим этапом построения диаграммы классов является установление отношений между классами. Для того, чтобы показать, что объекты одного класса взаимодействуют с объектами другого класса, устанавливаются отношения ассоциации между соответствующими классами. Выберите инструмент Association (Ассоциация) на панели инструментов. Проведите мышью линию ассоциации от одного класса к другому.

Для задания имени отношения выделите его. Введите имя. Или откройте окно спецификации отношения. Перейдите на вкладку General (Общие). Введите имя отношения в поле имени. На этой же вкладке можете ввести стереотип в поле Stereotype. Если отношение связывает классы со стереотипом business worker, то оно, как правило, имеет стереотип «communicate». Если же ассоциация связывает класс business worker с классом business entity, то для нее может быть задан стереотип «uses».

При выборе имени отношения помните, что как правило это глагол или глагольная фраза, описывающая, зачем нужно отношение. Пример имени ассоциации между Продавцом и Заказом – «Формирует», между Изготовителем и Заказом – «Просматривает», между Изготовителем и Продуктом – «Создает».

На диаграмме классов могут быть отражены также отношения структурирования – обобщения и включения. С помощью **обобщений (generalization)** показывают отношения наследования между двумя классами. Введите класс Служащий, являющийся предком для классов Продавец и Отправитель. Добавьте в него атрибуты, общие для классов-потомков, например, «фамилия, имя, отчество», «стаж работы». Соответствующие атрибуты у классов-потомков можно удалить. Выберите кнопку Generalization на панели инструментов. Проведите линию обобщения от подкласса (потомка) к суперклассу (предку).

В заключение можете документировать основные элементы диаграммы. Для добавления к классу текстового описания, выделите класс в браузере или в окне диаграммы. Введите текст в окно документации. Или откройте окно спецификации класса и введите информацию в области Documentation.

4.2. Самостоятельная работа

1. Просмотрите диаграммы последовательности и кооперации, построенные для выбранного вами бизнес-процесса во время выполнения предыдущей лабораторной работы. Выделите классы для активных объектов, представленных на этих диаграммах. Добавьте классы для объектов-сущностей, обрабатываемых или создаваемых в ходе выполнения процесса. Введите обобщенные классы.

2. Для каждого класса определите перечень атрибутов и перечень операций. Операции классов для активных объектов должны быть соотнесены с соответствующими сообщениями, представленными на диаграмме последовательности или кооперации. У классов объектов-сущностей операции вообще могут отсутствовать.

3. Определите отношения, связывающие классы. Для каждого отношения задайте имя, стереотип.

3. Создайте рукописные диаграммы классов для выбранного бизнес-процесса.

4.3. Выполнение индивидуального задания

1. Откройте файл, в котором хранится модель, созданная на лабораторных работах №1-3 по индивидуальному заданию.

2. Создайте диаграммы классов для основных прецедентов, представленных на диаграмме Use case. На диаграммах обязательно должны быть представлены как классы активных объектов (категории business worker), так и классы объектов-сущностей (категории business entity). Обязательно должны присутствовать обобщенные классы (суперклассы). Должны быть показаны отношения коммуникации, использования, обобщения или включения.

Варианты индивидуального задания

1. Ремонт квартир
2. Ремонт автомобилей
3. Проведение праздничных мероприятий (свадеб, юбилеев и т.д.)
4. Организация турпоездов
5. Пошив и ремонт верхней одежды
6. Проведение рекламных компаний
7. Оказание услуг по операциям с недвижимостью
8. Гостиничное обслуживание
9. Организация выставок и ярмарок
10. Издание печатной продукции
11. Продажа и ремонт компьютеров
12. Производство и продажа мебели на заказ
13. Трудоустройство
14. Организация обучения и консультирования
15. Оказание жилищно-коммунальных услуг
16. Оказание услуг по автоперевозкам (пассажирским и/или грузовым)
17. Организация спортивных мероприятий (турниров, игр и т.д.)
18. Изготовление кондитерских изделий (тортов, пирожных)
19. Выпуск газеты
20. Оказание медицинских услуг
21. Оказание маркетинговых услуг
22. Организация выборных компаний
23. Оказание услуг брачного агентства
24. Производство, продажа и сопровождение программной продукции
25. Строительство гаражей, садовых домиков и т.д.