

Департамент внутренней и кадровой политики
Белгородской области
Областное государственное автономное
профессиональное образовательное учреждение
«Валуйский колледж»

**Методические рекомендации по организации и выполнению
практических заданий**

МДК 02.02 Инструментальные средства разработки программного обеспечения

(МДК, УД)

Специальность

09.02.07 Информационные системы и программирование

Рассмотрено на заседании ПЦК предметно-цикловой комиссией математических дисциплин и информационных технологий, протокол
№ 1 от «31» августа 2020 г.

Руководитель: И. В. Крапивина

Методические рекомендации для проведения практических работ студентов по МДК 02.02
Инструментальные средства разработки программного обеспечения 09.02.07 Информационные системы и программирование

Составитель: Дураков С.Г., преподаватель

Пояснительная записка

К основным видам учебных занятий наряду с другими (урок, лекция, семинар, контрольная, лабораторная работа, консультация, практика, курсовая работа) относится практическое занятие, которое направлено на формирование учебных и профессиональных практических умений.

Состав и содержание практического занятия определяется его ведущей дидактической целью: формирование практических умений:

- профессиональных (выполнять определенные действия, операции, необходимые в последующем в профессиональной деятельности);
- учебных, необходимых в последующей учебной деятельности.

Состав и содержание практических занятий направлены на реализацию государственных требований к минимуму содержания и уровню подготовки студентов. Методические рекомендации предназначены для студентов второго курса многопрофильного отделения специальности 09.02.07 Информационные системы и программирование. Практические занятия в среднем профессиональном образовании являются специфическим педагогическим средством организации и управления деятельностью студентов в учебном процессе.

Практическая работа студентов по МДК 02.02 Инструментальные средства разработки программного обеспечения проводится с целью:

- систематизации и закрепления полученных теоретических знаний и практических умений студентов;
- углубления и расширения теоретических знаний;
- развития познавательных способностей и активности студентов: творческой инициативы, самостоятельности, ответственности и организованности;
- формирования самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- развития исследовательских умений.

В данных рекомендациях содержатся планы практических занятий с указанием целей, задач, самостоятельных работ, контрольных работ.

Практическая работа № 1

Создание модели данных с помощью ERwin 4.0

1. ЦЕЛЬ РАБОТЫ

Обрести навык работы с интерфейсом ERwin 4.0.

2. ПРИБОРЫ И ОБОРУДОВАНИЕ

ПЭВМ типа IBM PC/XT

Case-средство ERwin 4.0.

3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Изучите основные теоретические сведения (приложение А).

Запустите **ERwin** (рис.1.).

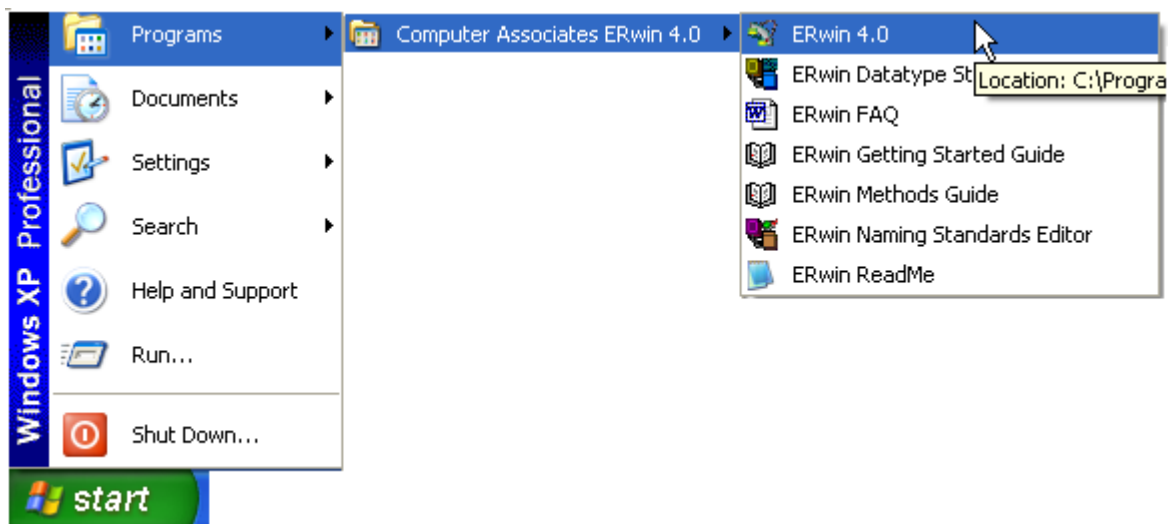


Рисунок 1. – Порядок запуска ERwin

Появляется диалоговое окно **Computer Associates ERwin** (рисунок 2). В этом окне выберите опцию **Create a new model** (Создание новой модели) и щелкните по кнопке **OK** (Создать).

Примечание

Обратите внимание на то, что диалоговое окно **Computer Associates ERwin** (рисунок 2) позволяет открыть уже существующую модель данных с помощью выбора опции **Open an existing file** (Открытие существующего файла). Этой опцией вы будете пользоваться в последующих упражнениях, когда у вас появится набор моделей. Список доступных для открытия моделей будет отображен в поле списка.

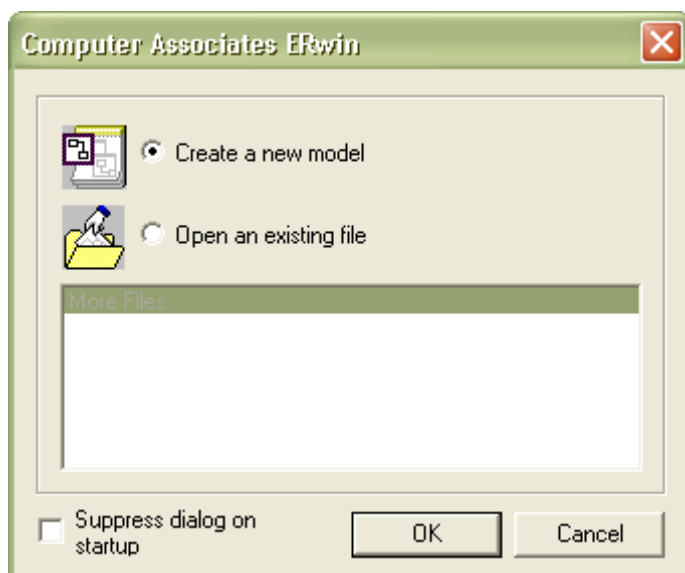


Рисунок 2 –Выбор типа модели

Откроется диалоговое окно **Create Model – Select Template** (Создание модели – Выбор шаблона) (рисунок 3).

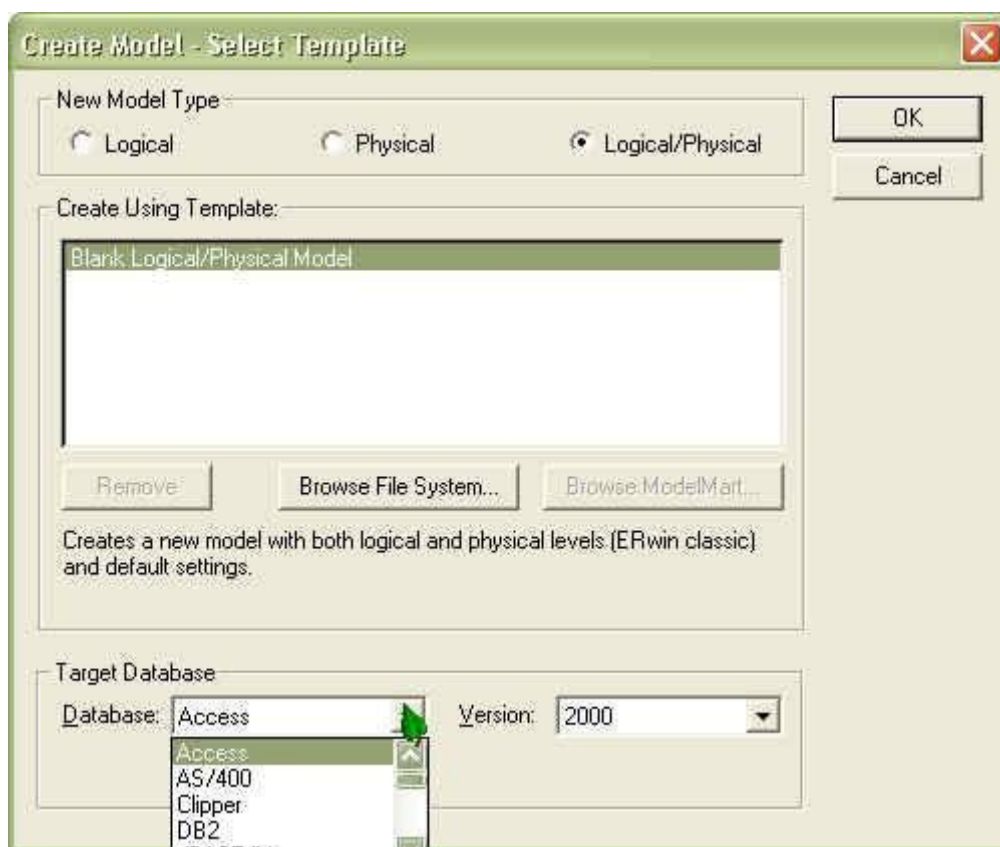


Рисунок 3 – Выбор шаблона модели

Во фрейме **New Model Type** установите опцию **Logical/Physical**, а во фрейме **Target Database** (Целевой сервер базы данных) установите в раскрывающемся окне списка систему управления базой данных (СУБД), например, **Access**, и ее версию, например, 2000.

Автоматически создается незаполненная модель данных **ERwin** (рисунок 4). Таким образом, вы получили доступ к интерфейсу среды **ERwin**. Знакомство с этим интерфейсом и составляет основное содержание упражнения 1.

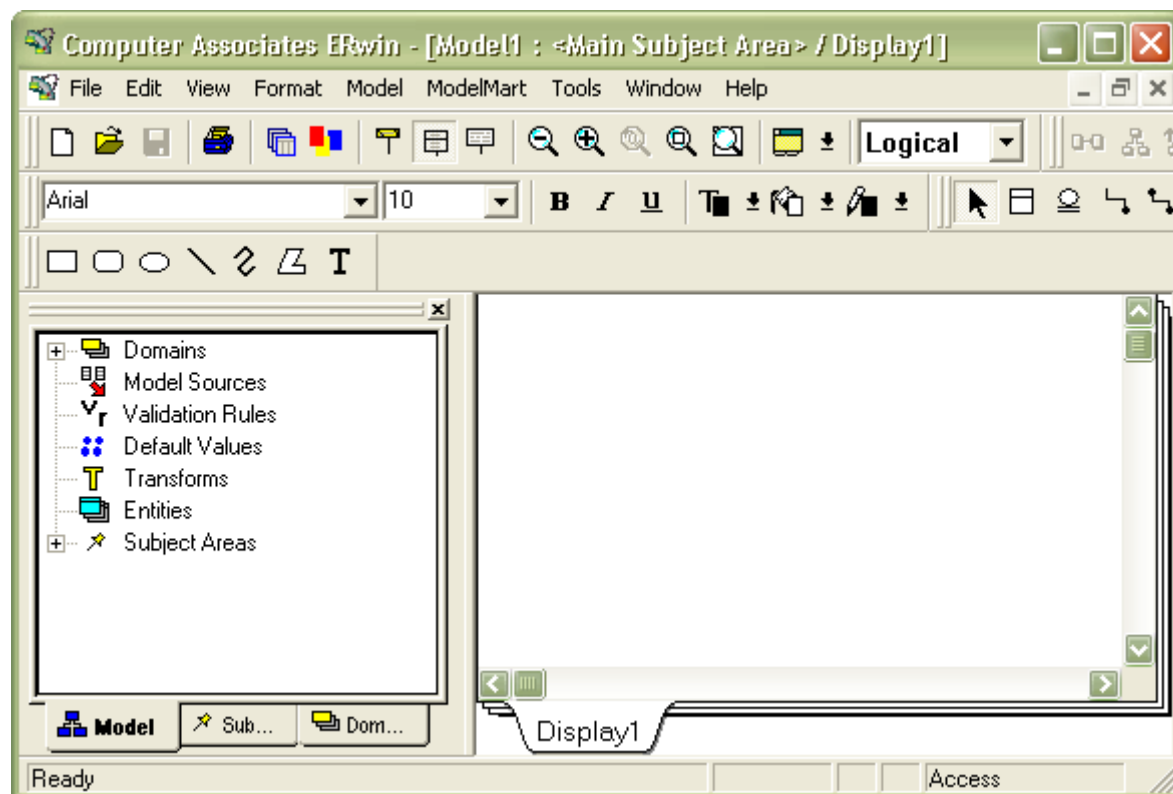


Рисунок 4 – незаполненная модель данных **ERwin**

Рабочее пространство ERwin включает «Окно диаграммы» (Diagram Window) и «Браузер (проводник) модели» (Model Explorer) (рисунок 5).

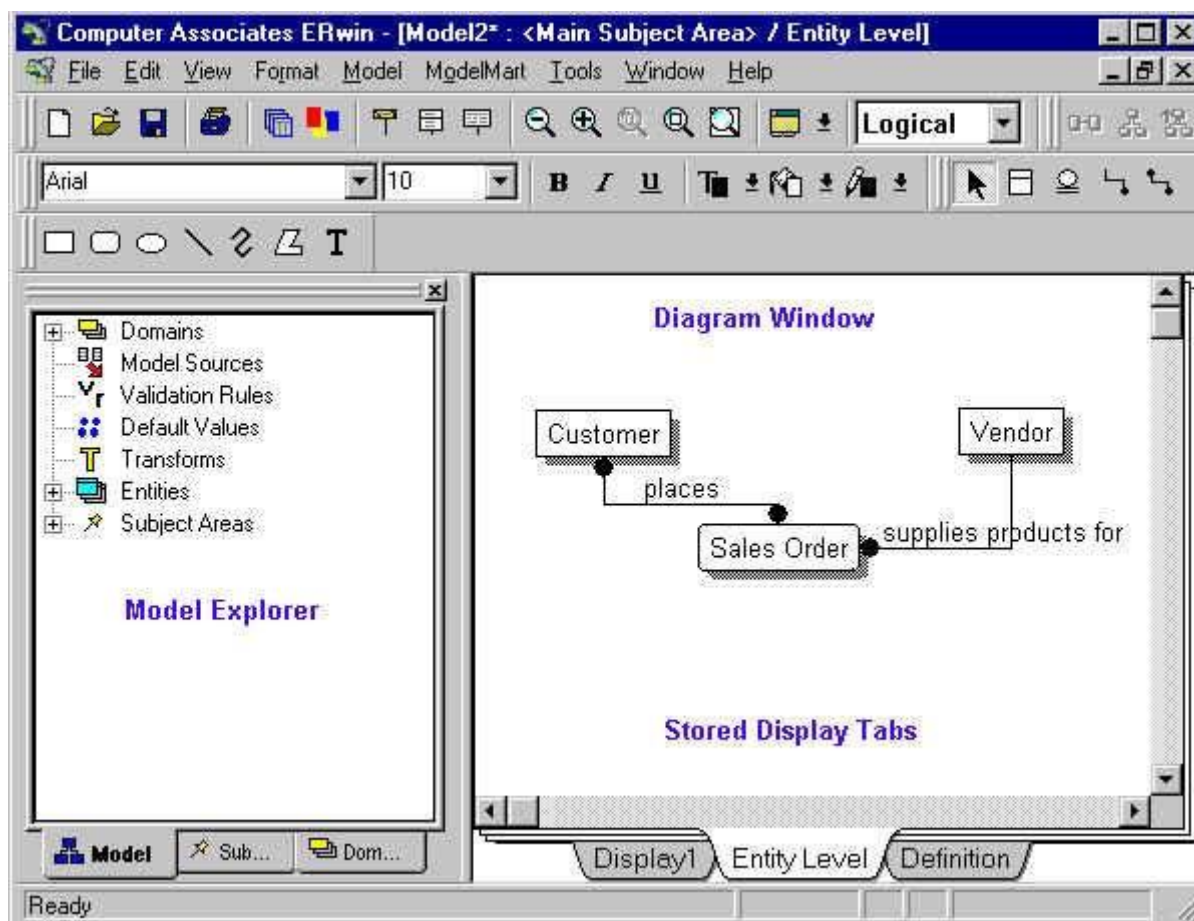


Рисунок 5 – Рабочее пространство ERwin

Главное меню представлено на рисунке 6.

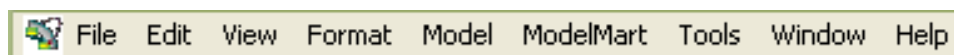


Рисунок 6 – Главное меню ERwin

Содержание пунктов главного меню раскрыто в таблицах 1-10 (приложение Б)

Содержание элементов панели инструментов раскрыто в таблицах 11-14 (приложение В)

В меню File выберите опцию Open. Появится диалоговое окно открытия модели (рисунок 7). Укажите путь и имя модели c:\Program Files\Computer Associates\Erwin 4.0\Samples\Standard\Emovies и нажмите OK.

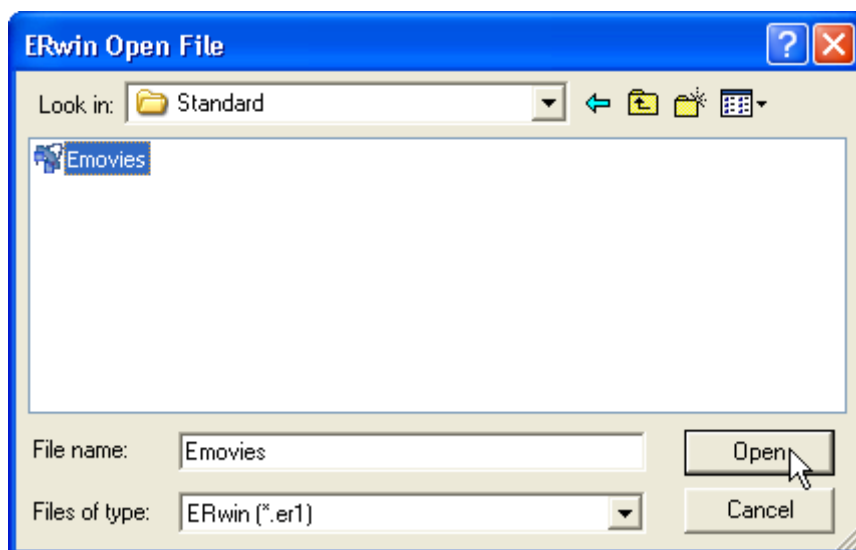


Рисунок 7 – Диалоговое окно для открытия файла примера

Повторно нажмите ОК (рисунок 8), когда появится диалоговое окно с сообщением о том, что этот файл имеет атрибут “read-only” (только для чтения).

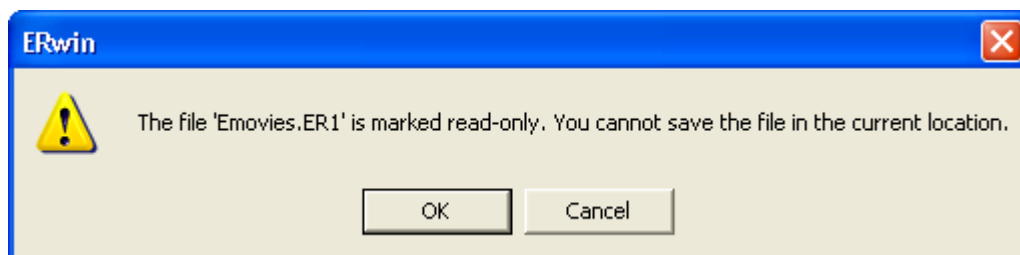


Рисунок 8 – Окно сообщения при открытии файла примера

В окне редактирования должна открыться модель Emovies.er1. Убедитесь в том, что переключатель типа модели на панели инструментов поставлен в положение Logical (см. таблицу 11 - приложение В). Сохраните модель под новым именем (рисунок 9).

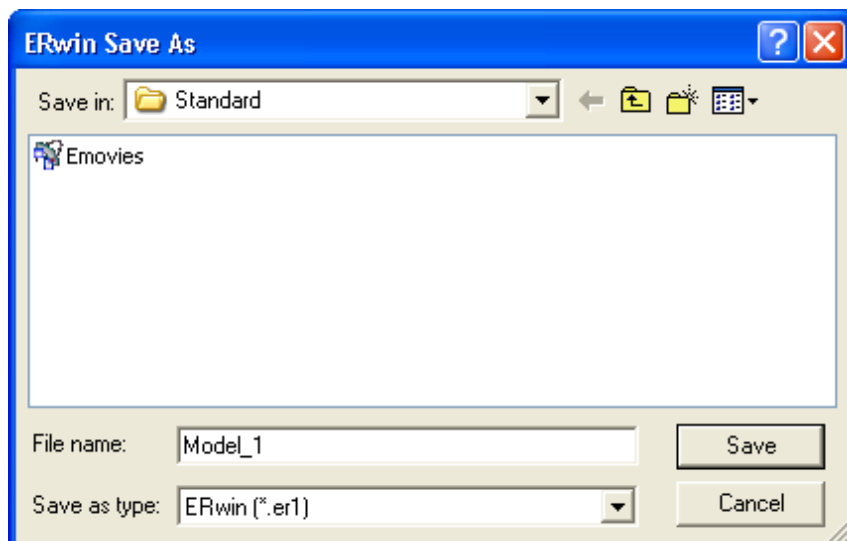


Рисунок 9 – Окно сохранения файла примера под новым именем Model_1

Выберите команды главного меню View → Toolbars. Обратите внимание на то, как реагирует на это действие инструментальная панель (должно появиться ниспадающее меню). Убедитесь в том, что некоторые опции в этом меню отмечены галочками (являются активными). Измените перечень активных опций в рассматриваемом меню, убрав отдельные галочки и посмотрите, как это повлияет на интерфейс среды **ERwin**.

Для того чтобы закрыть ниспадающее меню, кликните в любом месте на инструментальной панели. Потренируйтесь в этих действиях перед продолжением работы.

Потренируйтесь в переключении вида модели **Physical** и **Logical**.

Переключитесь на физическую модель (выберите опцию **Physical** списка в инструментальной панели **ERwin**).

Переключитесь на логическую модель (выберите опцию **Logical** списка в инструментальной панели **ERwin**).

Обратите внимание, как влияют такие переключения на вид модели в окне редактирования. Например, логические названия (имена) заменяются физическими названиями (именами) и браузер независимых атрибутов переключается на браузер независимых столбцов (если браузер невидим, нажмите Ctrl+B.)

Измените нотацию модели, перейдя от нотации **IDEF1X** к нотации **IE**.

Измените нотацию модели, перейдя от нотации **IE** к нотации **IDEF1X**.

Если вы выполнили шаги 1 – 5 и сделали все правильно, то должны получить диаграмму, совпадающую с контрольной диаграммой (рисунок 10).

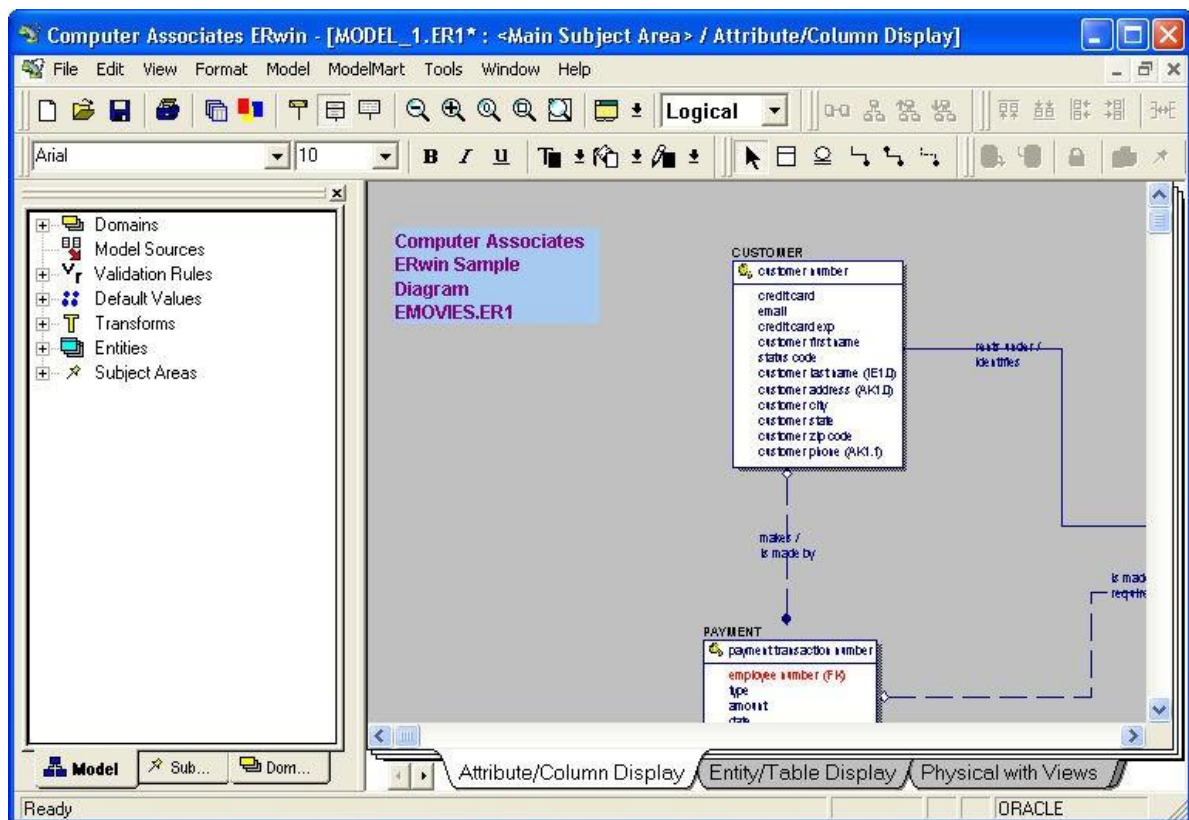


Рисунок 10 – Контрольная диаграмма

Закройте инструментальную среду создания модели процессов. Для этого, используя главное меню ERwin, последовательно выполните следующие команды: **File**→ **Exit**. Для закрытия ERwin без сохранения результатов моделирования воспользуйтесь опцией **Close without saving** и нажмите **OK**.

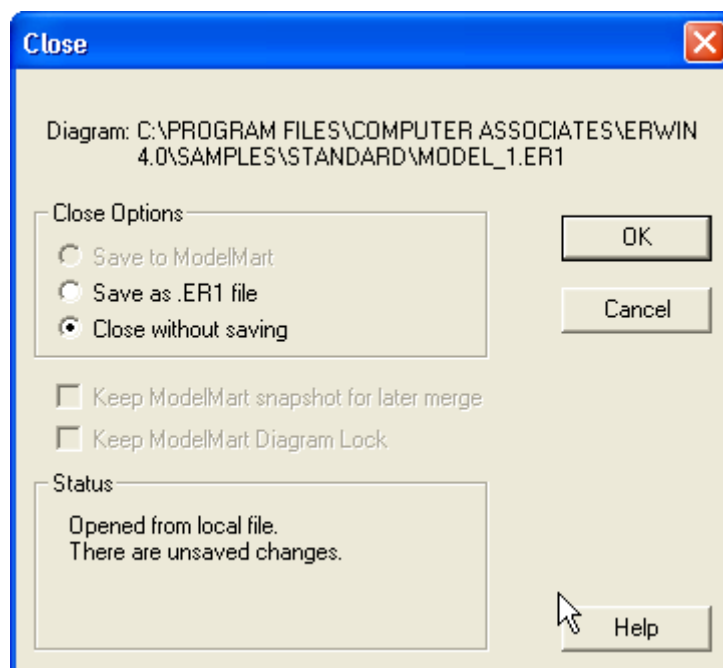


Рисунок 11 – Диалоговое окно Close

Сделайте вывод по проделанной работе.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

Что такое ERwin и зачем его используют?

Что такое физическая и логическая модель данных?

Как решаются в ERwin задачи документирования модели?

Как происходит управление окнами в ERwin?

Каков смысл термина масштабирование при работе с ERwin?

Каковы свойства и общая характеристика системы меню ERwin?

Дайте характеристику меню: File, Edit, View, Format, Model, Model Mart, Tools, Window, Help. Каков порядок настройки IDE PLATINUM ERwin?

Каково назначение кнопок стандартной панели инструментов?

Каковы функции палитры инструментов?

Какие связи можно создать с помощью кнопок палитры инструментов?

Каковы свойства справочной системы PLATINUM ERwin?

Каковы нотации представления ERwin модели?

ПРИЛОЖЕНИЕ А

Общие сведения о **ERwin**

ERwin - CASE-средство проектирования баз данных от фирмы Computer Associates. ERwin сочетает графический интерфейс Windows, инструменты для построения ER-диаграмм, редакторы для создания логического и физического описания модели данных и прозрачную поддержку ведущих реляционных СУБД. Для удобства изложения материала здесь и далее использована оригинальная терминология, принятая в ERwin.

ERwin не привязан к технологии какой-либо конкретной фирмы, поставляющей СУБД или средства разработки. Он поддерживает различные серверы баз данных и настольные СУБД, а также может обращаться к базе данных через интерфейс ODBC. Так, в текущей версии ERwin встроена поддержка 23 СУБД, среди которых: Oracle; Microsoft SQL Server и т.п. (см. таблицу 1.14). Заметим лишь, что речь идет только о реляционных СУБД.

ERwin можно использовать совместно с некоторыми популярными средствами разработки клиентских частей приложений: PowerBuilder, Visual Basic, Delphi. Кроме того, ERwin поддерживает работу в среде групповой разработки Model Mart, являющейся продуктом той же Computer Associates.

Процесс моделирования в ERwin базируется на методологии проектирования реляционных баз данных IDEF1X. Данная методология была разработана для ВВС США и теперь широко используется в правительственных учреждениях и частных компаниях, как в самих США, так и далеко за их пределами. Она определяет стандарты терминологии и графического изображения типовых элементов на ER-диаграммах. Заметим, что некоторые обозначения могут несколько расходиться с традиционными, принятыми в ER-модели, хотя в ERwin версии 4.0 существует возможность выбора традиционной нотации. (При изложении материала использована нотация IDEF1X). Кроме того, существует ряд отличий, связанных с тем, что данная методология ориентирована на разработку реляционных БД. Но это не вносит заметных коррективов в сам подход к разработке структуры БД, а жесткая стандартизация позволяет избежать такого недостатка ER-моделей, как возможность различной трактовки.

Для более детального изучения возможностей данного CASE-средства читатель может обратиться к книге [1].

Отображение модели данных в **ERwin**

Физическая и логическая модель данных

ERwin имеет два уровня представления модели - логический и физический. Логический уровень - это абстрактный взгляд на данные, на нем данные представляются так, как выглядят в реальном мире, и могут называться так, как они называются в реальном мире, например

«Постоянный клиент», «Отдел» или «Фамилия сотрудника». Объекты модели, представляемые на логическом уровне, называются сущностями и атрибутами (подробнее о сущностях и атрибутах будет рассказано ниже). Логическая модель данных может быть построена на основе другой логической модели, например на основе модели процессов (см. BPwin). Логическая модель данных является универсальной и никак не связана с конкретной реализацией СУБД.

Физическая модель данных, напротив, зависит от конкретной СУБД, фактически являясь отображением системного каталога. В физической модели содержится информация о всех объектах БД. Поскольку стандартов на объекты БД не существует (например, нет стандарта на типы данных), физическая модель зависит от конкретной реализации СУБД. Следовательно, одной и той же логической модели могут соответствовать несколько разных физических моделей. Если в логической модели не имеет значения, какой конкретно тип данных имеет атрибут, то в физической модели важно описать всю информацию о конкретных физических объектах - таблицах, колонках, индексах, процедурах и т. д. Разделение модели данных на логические и физические позволяет решить несколько важных задач.

Документирование модели. Многие СУБД имеют ограничение на именование объектов (например, ограничение на длину имени таблицы или запрет использования специальных символов - пробела и т. п.). Зачастую разработчики ИС имеют дело с нелокализованными версиями СУБД. Это означает, что объекты БД могут называться короткими словами, только латинскими символами и без использования специальных символов (т. е. нельзя назвать таблицу предложением - только одним словом). Кроме того, проектировщики БД нередко злоупотребляют "техническими" наименованиями, в результате таблица и колонки получают наименования типа ***RTD_324*** или ***CUST_A12*** и т. д. Полученную в результате структуру могут понять только специалисты (а чаще всего только авторы модели), ее невозможно обсуждать с экспертами предметной области. Разделение модели на логическую и физическую позволяет решить эту проблему. На физическом уровне объекты БД могут называться так, как того требуют ограничения СУБД. На логическом уровне можно этим объектам дать синонимы - имена более понятные неспециалистам, в том числе на кириллице и с использованием специальных символов. Например, таблице ***CUST_A12*** может соответствовать сущность *Постоянный клиент*. Такое соответствие позволяет лучше задокументировать модель и дает возможность обсуждать структуру данных с экспертами предметной области.

Масштабирование. Создание модели данных, как правило, начинается с создания логической модели. После описания логической модели, проектировщик может выбрать необходимую СУБД и ERwin автоматически создаст соответствующую физическую модель. На основе физической модели ERwin может сгенерировать системный каталог СУБД или

соответствующий SQL-скрипт. Этот процесс называется прямым проектированием (Forward Engineering). Тем самым достигается масштабируемость - создав одну логическую модель данных, можно сгенерировать физические модели под любую поддерживаемую ERwin СУБД. С другой стороны, ERwin способен по содержимому системного каталога или SQL-скрипту воссоздать физическую и логическую модель данных (Reverse Engineering). На основе полученной логической модели данных можно сгенерировать физическую модель для другой СУБД и затем сгенерировать ее системный каталог. Следовательно, ERwin позволяет решить задачу по переносу структуры данных с одного сервера на другой. Например, можно перенести структуру данных с Oracle на Informix (или наоборот) или перенести структуру dbf-файлов в реляционную СУБД, тем самым, облегчив решение по переходу от файл-серверной к клиент-серверной ИС. Заметим, однако, что формальный перенос структуры "плоских" таблиц на реляционную СУБД обычно неэффективен. Для того чтобы извлечь выгоды от перехода на клиент-серверную технологию, структуру данных следует модифицировать. Процессы прямого и обратного проектирования будут рассмотрены ниже.

Для переключения между логической и физической моделью данных служит список выбора в левой части панели инструментов Erwin (рисунок 1).

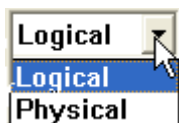


Рисунок 1 - Переключение между логической и физической моделью

При переключении, если физической модели еще не существует, она будет создана автоматически.

Для создания логических моделей данных в ERwin можно использовать две нотации: **IDEF1X** и **IE** (Information Engineering). Методология **IDEF1X** была разработана для армии США и широко используется в государственных учреждениях США, финансовых и промышленных корпорациях. Методология **IE**, разработанная Мартином, Финкельштейном и другими авторами, используется преимущественно в промышленности. Переключение между нотациями можно с помощью выбора команд Model → Model Properties. После того как откроется диалоговое окно Model Properties (рисунок 2) следует раскрыть вкладку Notation и переключить радиокнопку во фрейме Logical Notation в требуемое положение. В дальнейшем будем использовать нотацию IDEF1X, так как она получила большее признание.

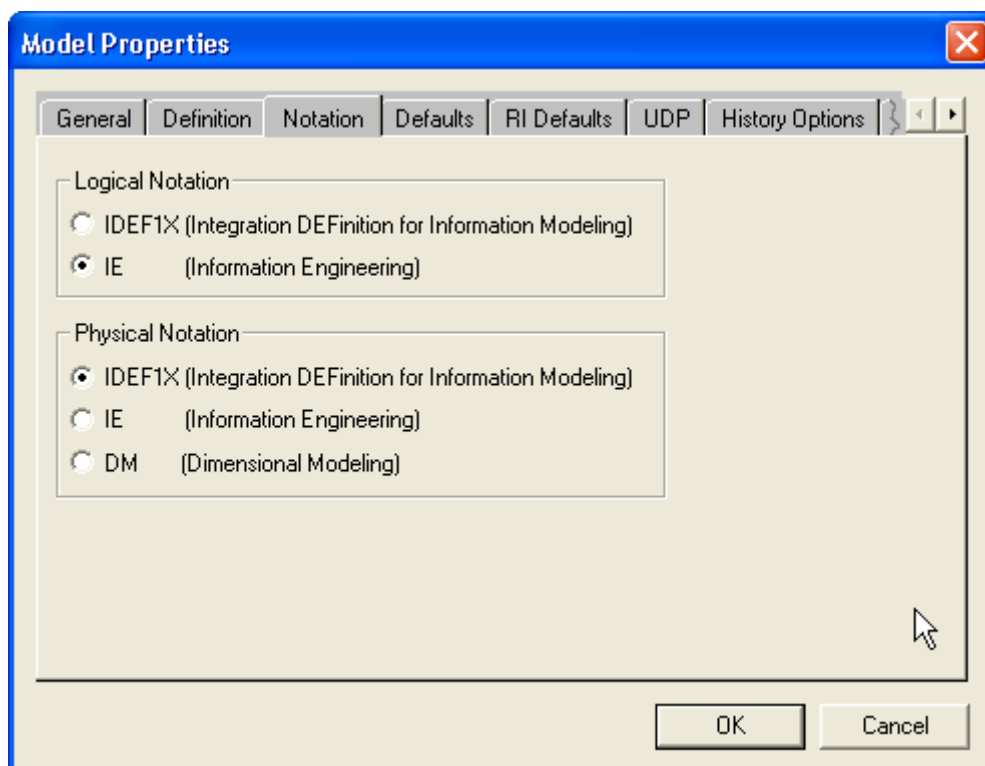
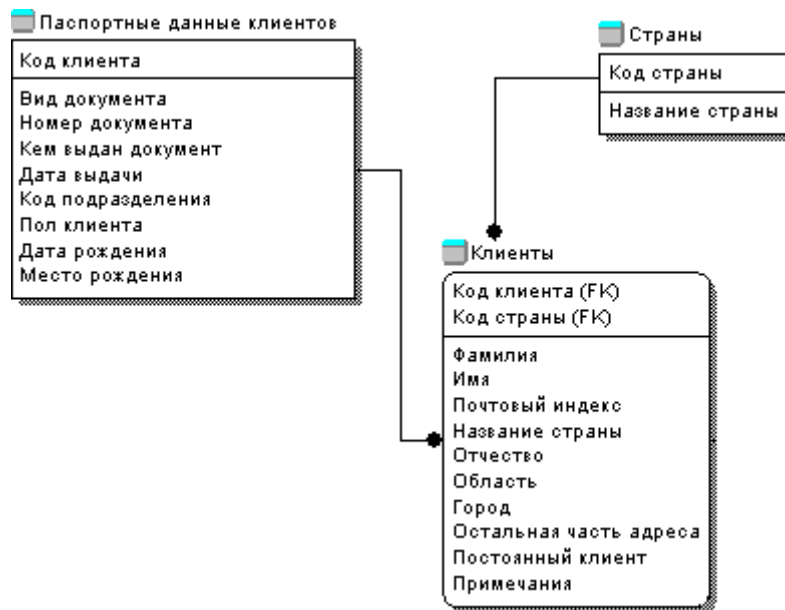


Рисунок 2 – Диалоговое окно Model Properties

На рисунке 3 представлен вид фрагмента логической модели данных для работы с заявками клиентов в двух нотациях: IDEF1X и IE.

Как видно из этого рисунка, внешне две указанные нотации отличаются только обозначениями соответствующих связей между сущностями модели. Здесь

рассматриваются три сущности: Паспортные данные клиентов, Клиенты и Страны. Между указанными сущностями Паспортные данные клиентов – Клиенты и Страны - Клиенты установлена идентифицирующая взаимосвязь.



а)



б)

Рисунок 3 – Две нотации IDEF1X и IE, используемые для построения
логической модели данных в ERwin

ПРИЛОЖЕНИЕ Б

Таблица 1 - Содержание пунктов главного меню

Название пункта меню	Назначение пункта меню
File	Открытие, закрытие, сохранение Модели
Edit	Редактирование модели
View	Вид модели
Format	Форматирование модели
Model	Свойства модели
ModelMart	Хранилище моделей
Tools	Инструментарий
Window	Окна
Help	Справка

Таблица 2 – Содержание меню **File**

Название подпункта меню	Назначение подпункта меню
New	Вызывает окно ERwin Template Selection , в котором вы можете выбрать шаблон, чтобы использовать как основание для создания новой модели.
Open	Показывает окно ERwin Open File , в котором вы можете выбрать существующую модель, чтобы открыть ее.
Close	Вызывает окно Close dialog , в котором вы получаете возможность сохранить или отказаться от сохранения изменений, внесенных в модель
Save	Вызывает окно Save As , в котором вы получаете возможность сохранить модель под старым или новым именем (команда доступна только после того как модель будет хотя бы раз сохранена)
Save As	Вызывает окно Save As , в котором вы получаете возможность сохранить модель под новым именем (команда доступна всегда)
Save As New Model	Вызывает окно Save As , в котором вы получаете возможность сохранить модель под новым идентификационным номером (команда доступна всегда)

Import	Импорт
BPwin	Импорт модели из BPwin в ERwin или экспорт модели ERwin в BPwin
Designer 2000	Импорт модели из Oracle Designer/2000 в ERwin или экспорт модели ERwin в хранилище Oracle Designer/2000
Export	Экспорт
BPwin	Экспорт модели ERwin в BPwin
Designer 2000	Экспорт модели ERwin в хранилище Oracle Designer/2000
Print	Вызывает окно Print dialog , в котором вы выбираете режим вывода на принтер активной модели ERwin
Print Setup	Вызывает окно Print Setup , в котором вы выбираете установки принтера
Recent File	Список из четырех последних моделей ERwin, доступных для открытия
Exit	Выход из ERwin

Таблица 3 – Содержание меню **Edit**

Название подпункта меню	Назначение подпункта меню
Cut	Вырезание выделенных объектов из модели
Copy	Копирование выделенных объектов модели
Paste	Вставка содержимого буфера обмена в модель
Select All	Выделение всех объектов активной (текущей) Модели
Go To	Переход к указанному объекту модели

Таблица 4 – Содержание меню **View**

Название подпункта меню	Назначение подпункта меню
Redraw Diagram	Перерисовать диаграмму модели
Toolbars	Отображение или скрытие панелей инструментов
Model Explorer	Отображение или скрытие проводника модели
Stored Display Tabs	Отображение или скрытие уровней вложенности Модели

Status Bar	Отображение или скрытие строки состояния Модели
Zoom	Управление масштабом отображения модели

Таблица 5 – Содержание меню **Format**

Название подпункта меню	Назначение подпункта меню
Display Level	Уровень отображения модели
Entity Display	Уровень отображения сущностей модели
Relationship Display	Уровень отображения отношений между сущностями модели
Stored Display Tabs	Вызов диалогового окна Stored Display для задания данных об авторе модели и установки параметров отображения модели
Preferences	Вызов диалогового окна Format Preferences для задания предпочтительных режимов отображения сущностей и других параметров модели
Default Fonts & Colors	Вызов диалогового окна Default Fonts & Colors для задания параметров шрифта и его цвета при отображении различных параметров модели
Align or Space Evenly	Выравнивание или перемещение выделенных объектов модели в пределах окна модели
Show Shadows	Отображение или скрытие теней для сущностей модели
Show Page Grid	Отображение или скрытие координатной сетки для упрощения размещения сущностей модели

Таблица 6 – Содержание меню **Model**

Название подпункта меню	Назначение подпункта меню
Subject Areas	Вызов диалогового окна Subject Areas (подмножество модели) для выбора тематически общих сущностей модели
Entities	Вызов диалогового окна Entities для задания параметров выделенной

	сущности модели
Attributes	Вызов диалогового окна Attributes для задания параметров атрибутов выделенной сущности модели
Relationships	Вызов диалогового окна Relationships для задания параметров отношения выделенной связи между сущностями модели
Key Groups	Вызов диалогового окна Key Groups для задания параметров ключевых атрибутов выделенной сущности модели
Domain Dictionary	Вызов диалогового окна Domain Dictionary для задания параметров доменов сущностей модели
Validation Rules	Вызов диалогового окна Validation Rules для задания параметров проверки корректности задания сущностей модели
Default Values	Вызов диалогового окна Default Values для задания параметров модели по умолчанию
UDP Dictionary	Вызов диалогового окна User Defined Property Dictionary для задания словаря параметров модели, определяемых пользователем
Model Sources	Вызов диалогового окна Model Sources для задания источников Модели
Model Properties	Вызов диалогового окна Model Properties для задания данных о Модели
Logical Mode	Отображение логической модели данных
Physical Model	Отображение физической модели данных

Таблица 7 – Содержание меню **ModelMart**

Название подпункта меню	Назначение подпункта меню
Open	Открыть хранилище моделей
Close	Закрыть хранилище моделей
Save	Сохранить модель в хранилище модели
Save As	Сохранить модель в хранилище модели под указанным Именем
Lock	Закрыть модель, сохраненную в хранилище моделей, для Редактирования
Merge	Разбить модель

Version	Версия модели
Change Control	Изменить параметры контроля состояния модели
Refresh	Перерисовать модель
Library	Вызов библиотеки моделей
Subject Areas	Работа с подмножеством модели
ModelMart Manager	Вызов менеджера хранилища моделей
Connection	Задание параметров соединения с хранилищем моделей
Session	Параметры сессии соединения с хранилищем моделей
Security	Задание параметров защиты модели
BP/ER Synchronizer	Задание параметров синхронизации моделей в средах BPwin и ERwin

Таблица 8 – Содержание меню **Tools**

Название подпункта меню	Назначение подпункта меню
Forward Engineer/Schema Generation	Вызов диалогового окна <Database> Server Schema Generation Report для выбора параметров генерации системного каталога для указанного целевого сервера СУБД и (или) сохранение сценария генерации системного каталога в формате сценария DDL или ASCII текстовом файле (только для физической модели)
Reverse Engineer	Вызов диалогового окна Reverse Engineer - Set Options для задания параметров генерации ERwin модели на основе имеющегося SQL DDL или DL сценария или системного каталога СУБД
Add Model Source	Вызов диалогового окна мастера Source Model, который позволяет задать сведения об источниках (удобно при большом количестве источников)
Sync with Model Source	Вызов диалогового окна мастера Sync with Model Source , который позволяет синхронизировать существующую модель ERwin с источниками модели при групповой работе над проектом
Derive New Model	Вызов диалогового окна мастера Derive New Model котрый позволяет создать новый вариант модели на основе же существующей ERwin модели
Split L/P Model	Разбиение логической и физической модели
Report	Вызов диалогового окна мастера генерации отчета о модели

Builder	
Data Browser	Вызов редактора отчетов о модели
Volumetrics	Вызов диалогового окна Volumetrics для определения количественных характеристик параметров модели (размер таблиц и прочее)
Names	Вызов диалогового окна Model Naming Options для редактирования параметров имен, используемых в модели (максимальное количество символов и прочее)
Datatypes	Вызов диалогового окна Model Datatype Options для использования стандартных типов файлов данных в модели
Add Ins	Вызов диалогового окна Add-In Manager, которое позволяет подключить к ERwin дополнительное программное обеспечение, например, ERwin Examiner и др.

Таблица 9 – Содержание меню **Window**

Название подпункта меню	Назначение подпункта меню
Cascade	Окна модели в рабочем пространстве ERwin ориентированы каскадом
Tile Horizontal	Окна модели в рабочем пространстве ERwin ориентированы с горизонтальной разбивкой
Tile Vertical	Окна модели в рабочем пространстве ERwin ориентированы с вертикальной разбивкой

Таблица 10 – Содержание меню **Help**

Название подпункта меню	Назначение подпункта меню
Help Topics	Вызов окна ERwin Online справочной системы
Tutorial	Вызов диалогового окна учебника по работе с ERwin
What's New	Вызов обзора новинок, включенных в текущую версию ERwin по сравнению с предыдущей версией
How to Use Help	Вызов диалогового окна мастера How to Use Help, в котором можно найти рекомендации по использованию справочной системы
About ERwin	Вызов диалогового окна справки о текущей версии ERwin

ПРИЛОЖЕНИЕ В

Таблица 11 – Стандартная панель инструментов (Standart)






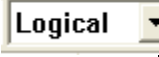
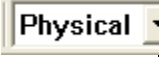
Кнопки	Назначение кнопок
	Создание, открытие, сохранение и печать модели
	Вызов диалога Report Browser для генерации отчета
	Изменение уровня просмотра модели: уровень сущностей, уровень атрибутов и уровень определений
	Изменение масштаба просмотра модели
	Создание и переключение между подмножествами модели – Subject Area
	Отображение логической модели данных
	Отображение физической модели данных

Таблица 12 – Инструментарий (палитра инструментов - Toolbox)







Кнопки	Назначение кнопок
	Кнопка указателя (режим мыши) – в этом режиме можно устанавливать фокус на каком-либо объекте модели. Всегда переходите в режим указателя мыши после редактирования объекта.
	Кнопка внесения сущности – для внесения сущности нужно щелкнуть левой кнопкой мыши по кнопке внесения сущностей и по свободному пространству на модели. Повторный щелчок приведет к внесению в модель новой сущности. Для редактирования сущностей или других объектов модели необходимо перейти в режим указателя
	Кнопка категории. Категория, или категориальная связь, - специальный тип связи между сущностями, которая будет рассмотрена ниже. Для установления категориальной связи нужно щелкнуть левой кнопкой мыши по кнопке категории, затем один раз щелкнуть по сущности - родовому предку, затем - по сущности - потомку
	Идентифицирующая связь между сущностями
	Связь между сущностями типа многие-ко-многим
	Неидентифицирующая связь между сущностями

Таблица 13 – Панель инструментов шрифт и цвет (Font and Color)

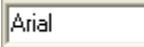
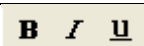





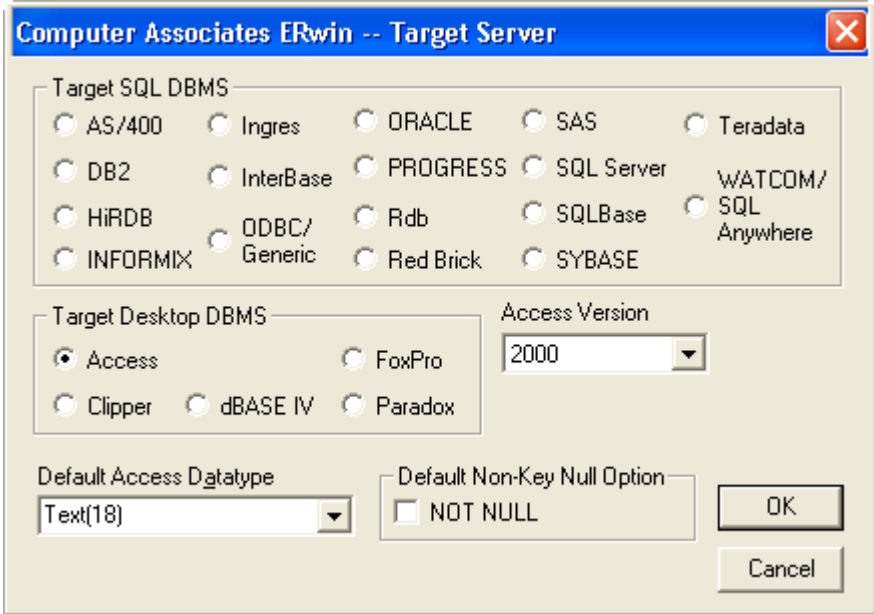
Кнопки	Назначение кнопок
	Выбор наименования шрифта
	Выбор размера шрифта
	Выбор стиля шрифта
	Выбор цвета символов
	Выбор цвета заливки
	Выбор цвета линий

Таблица 14 – Панель инструментов базы данных (Database)

Кнопки	Назначение кнопок
	Генерация схемы БД на основе логической модели данных (прямая задача)
	Реконструкция логической модели данных на основе схемы БД (обратная задача)
	Анализ адекватности результатов генерации схемы БД на основе логической модели данных
	<p>Выбор целевого сервера СУБД. Возможные варианты выбора показаны ниже.</p> 

Практическая работа № 2

Соответствие логической модели **ERwin** и модели процессов **BPwin**

1. ЦЕЛЬ РАБОТЫ

Изучить на практических примерах соответствие между логической моделью Erwin и моделью процессов Bpwin и генерацию отчетов в Bpwin.

2. ПРИБОРЫ И ОБОРУДОВАНИЕ

ПЭВМ типа IBM PC/XT

Case-средство ERwin 4.0.

3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Экспортируйте данные из ERwin в Bpwin и свяжите полученные данные с работами и стрелками. Для этого:

Из модели данных командой экспорта создаем файл экспорта данных из ERwin. В процессе экспорта данных из Erwin создается сообщение о результатах экспорта (Рис.1). В реальной задаче следует создавать сущности и атрибуты на русском языке.

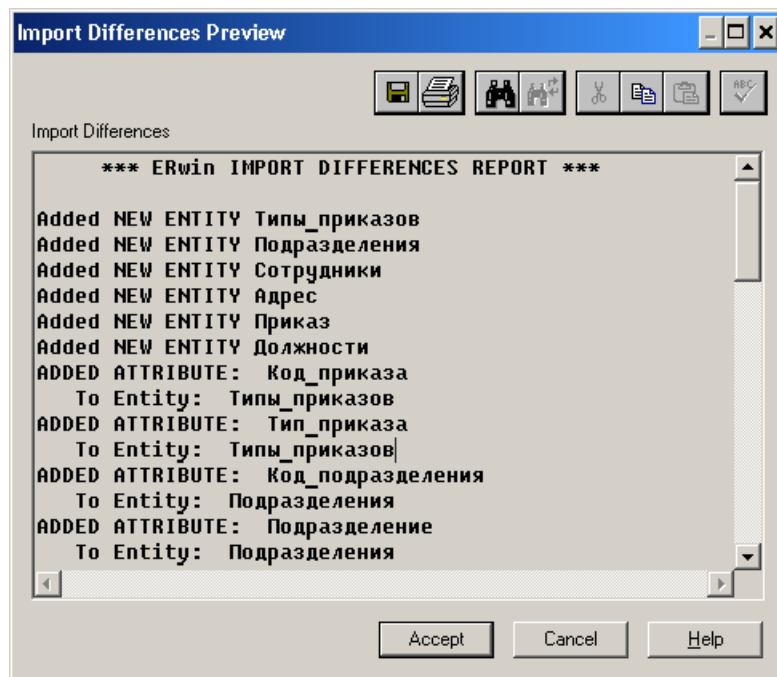


Рис.1. Экспортированные данные из ERwin в Bpwin

Далее импортируем данные в Bpwin и привязываем полученные данные к одной из работ и к связанным с ней стрелкам. Генерируем отчет DataUsage Report, содержащий данные о привязанных сущностях и атрибутах (Рис.2).

Report Format: Column		
Entity Name	Entity Definition	Attribute Name
Адрес		Город
		Дом
		Квартира
		Код_адреса
		Улица
Должности		Должность
		Код_должности
		Оклад
Подразделения		Код_подразделения
		Подразделение
Приказ		Дата
		Код_приказа
		Номер_приказа
		Табельный_номер
Сотрудники		Возраст
		Код_адреса

Рис.2. Отчет DataUsage Report

Этап дополнения словаря в Vрwin новой сущностью и атрибутами и экспорта данных из словаря сущностей из Vрwin в Erwin.

Новая импортированная в Erwin сущность не имеет первичного ключа и не связана с другими сущностями. Назначение атрибутов первичным ключом и связывание сущностей можно провести только средствами Erwin; другими словами, сущности и атрибуты, созданные в VРwin и затем импортированные в Erwin, можно рассматривать как заготовку для создания полноценной модели данных, а не как готовую модель.

Дополним словарь Vрwin новой сущностью и связанными с ней атрибутами. На рисунках 3 и 4 показан пример создания новых сущностей и атрибутов в словаре VРwin.

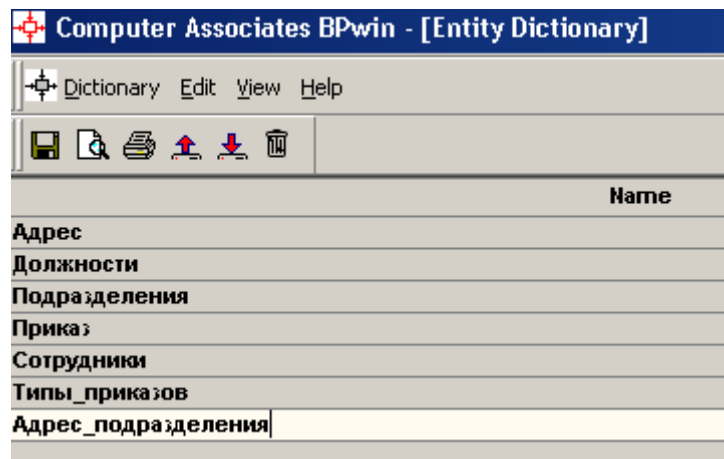


Рис.3. Добавление новой сущности

Computer Associates BPwin - [Attribute Dictionary]			
Dictionary Edit View Help			
Name Definition Exchange with ERwin Entity			
Возраст		<input checked="" type="checkbox"/> Exchange with ERwin	Сотрудники
Город		<input checked="" type="checkbox"/> Exchange with ERwin	Адрес
Дата		<input checked="" type="checkbox"/> Exchange with ERwin	Приказ
Должность		<input checked="" type="checkbox"/> Exchange with ERwin	Должности
Дом		<input checked="" type="checkbox"/> Exchange with ERwin	Адрес
Квартира		<input checked="" type="checkbox"/> Exchange with ERwin	Адрес
Код_адреса		<input checked="" type="checkbox"/> Exchange with ERwin	Сотрудники
Код_адреса		<input checked="" type="checkbox"/> Exchange with ERwin	Адрес
Код_должности		<input checked="" type="checkbox"/> Exchange with ERwin	Сотрудники
Код_должности		<input checked="" type="checkbox"/> Exchange with ERwin	Должности
Код_подразделения		<input checked="" type="checkbox"/> Exchange with ERwin	Подразделения
Код_подразделения		<input checked="" type="checkbox"/> Exchange with ERwin	Сотрудники
Код_приказа		<input checked="" type="checkbox"/> Exchange with ERwin	Типы_приказов
Код_приказа		<input checked="" type="checkbox"/> Exchange with ERwin	Приказ
Номер_приказа		<input checked="" type="checkbox"/> Exchange with ERwin	Приказ
Образование		<input checked="" type="checkbox"/> Exchange with ERwin	Сотрудники
Оклад		<input checked="" type="checkbox"/> Exchange with ERwin	Должности
Подразделение		<input checked="" type="checkbox"/> Exchange with ERwin	Подразделения
Табельный_номер		<input checked="" type="checkbox"/> Exchange with ERwin	Сотрудники
Табельный_номер		<input checked="" type="checkbox"/> Exchange with ERwin	Приказ
Телефон		<input checked="" type="checkbox"/> Exchange with ERwin	Сотрудники
Тип_приказа		<input checked="" type="checkbox"/> Exchange with ERwin	Типы_приказов
Улица		<input checked="" type="checkbox"/> Exchange with ERwin	Адрес
ФИО		<input checked="" type="checkbox"/> Exchange with ERwin	Сотрудники
Дом_1		<input checked="" type="checkbox"/> Exchange with ERwin	Адрес_подразделения
Улица_1		<input checked="" type="checkbox"/> Exchange with ERwin	Адрес_подразделения
Телефон_1		<input checked="" type="checkbox"/> Exchange with ERwin	Адрес_подразделения
		<input checked="" type="checkbox"/> Exchange with ERwin	Адрес_подразделения

Рис.4. Связывание атрибутов с новой сущностью

Командой экспорта экспортируем данные из словаря сущностей из Bpwin в файл экспорта. При импорте данных в Erwin создается сообщение (Рис.5).

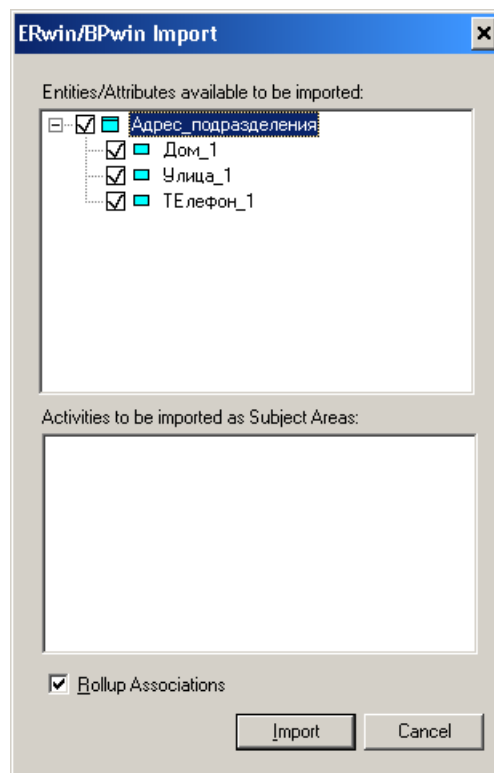


Рис.5. Импортирование данные из Bpwin в Erwin

Создайте отчеты.

После заполнения данных, характеризующих модель Bpwin в целом (автор, цель и т.д.) и одну из диаграмм (описание диаграммы), генерируем отчеты Model Report (Рис.6) и Diagram Report (Рис.7).

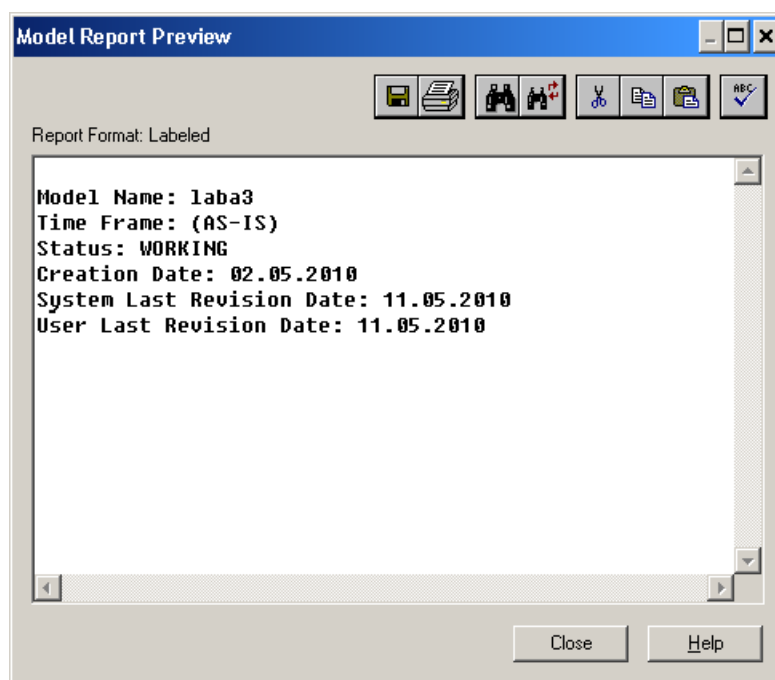


Рис.6. Отчет Model Report

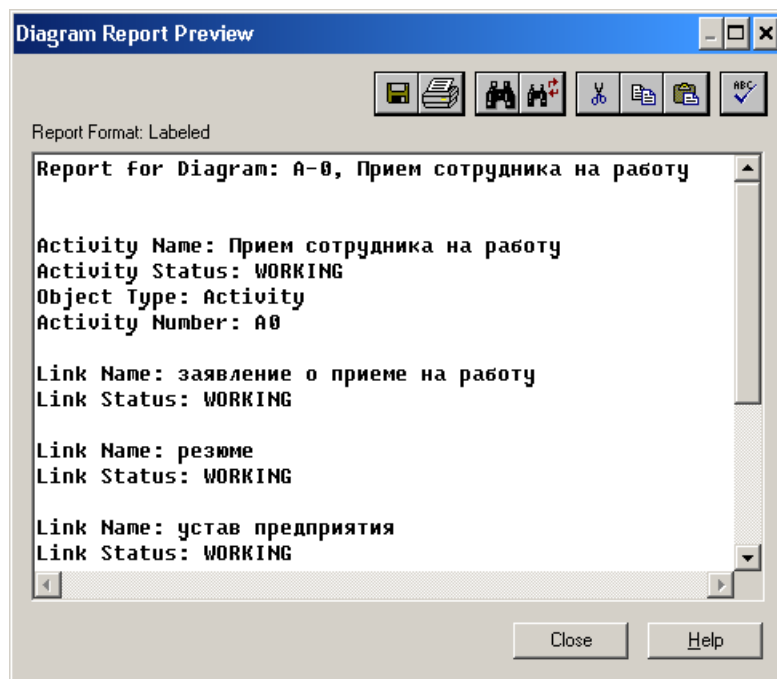


Рис.7. Отчет Diagram Report

Создаем в одной из работ Vrpwin пояснения к работе и относящимся к ней стрелкам и генерируем отчет Diagram Object Report (Рис.8).

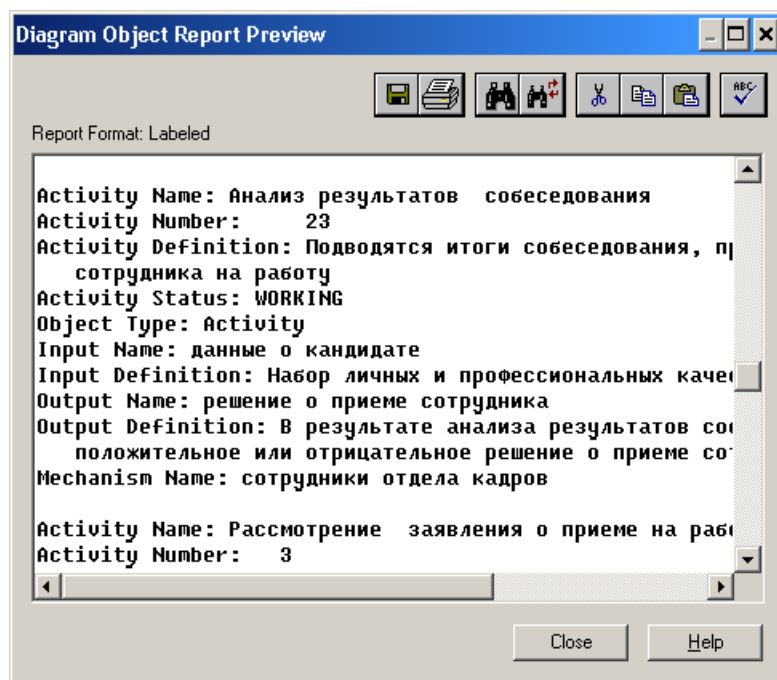


Рис.8. Отчет Diagram Object Report

Сделайте выводы по проделанной работе.

Практическая работа № 3
Введение в **Rational Rose**

1. ЦЕЛЬ РАБОТЫ

Изучить принципы работы с Case-пакетом Rational Rose 2003.

2. ПРИБОРЫ И ОБОРУДОВАНИЕ

ПЭВМ типа IBM PC/XT

Case-средство Rational Rose.

3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Изучите основные теоретические сведения (приложение А).

Запустите Rational Rose.

Выберите пункт **Tools > Options** и откройте вкладку **Toolbars**.

Чтобы сделать видимой или невидимой стандартную панель инструментов, пометьте (или снимите пометку) контрольный переключатель **show standard ToolBar** (или **show Diagram ToolBar**).

Увеличьте размер кнопок на панели инструментов:

Щелкните правой кнопкой мыши по требуемой панели.

Выберите во всплывающем меню пункт Use Large Buttons (Использовать большие кнопки), вернитесь к нормальному размеру кнопок.

Настройте панель инструментов:

Щелкните правой кнопкой мыши по панели диаграммы Main пакета Use Case View.

Выберите пункт Customize (настроить) и добавьте несколько кнопок.

Чтобы добавить или удалить кнопки, выберите соответствующую кнопку и затем щелкните мышью по кнопке **Add** (добавить) или **Remove** (удалить).

Сделайте выводы по проделанной работе.

ПРИЛОЖЕНИЕ А

Rational Rose – семейство объектно-ориентированных Case-средств, предназначенных для автоматизации процессов анализа и проектирования ПО, для генерации кодов на различных языках программирования и выпуска проектной документации.

Назначение элементов экрана интерфейса Rose:

Браузер (browser) - используется для быстрой навигации по модели. С помощью браузера можно добавлять к модели элементы, просматривать существующие элементы модели и связи между ними, перемещать и переименовывать элементы модели, добавлять элементы модели к диаграмме, группировать элементы в пакеты, связывать элемент с файлом или адресом Интернета, работать с детализированной спецификацией элемента, открывать диаграмму. Браузер поддерживает четыре представления (view): представление вариантов использования, компонентов, размещения и логическое представление.

Окно документации (documentation window) – применяется для работы с текстовым описанием элементов модели. С его помощью можно документировать элементы модели Rose. Например, можно сделать краткое описание каждого действующего лица. При документировании класса все, что будет написано в окне документации, появится затем как комментарий в сгенерированном коде. Документация будет выводиться также в отчетах, создаваемых в среде Rose.

Панели инструментов (toolbars) - применяются для быстрого доступа к наиболее распространенным командам. Панели инструментов Rose обеспечивают быстрый доступ к наиболее распространенным командам. В этой среде существуют два типа панелей инструментов: стандартная панель и панель диаграммы. Стандартная панель видна всегда, ее кнопки соответствуют командам, которые могут использоваться для работы с любой диаграммой. Панель диаграммы своя для каждого типа диаграмм UML.

Все панели инструментов могут быть изменены и настроены пользователем. Для этого используется пункт меню **Tools > Options**, затем вкладку **Toolbars**.

Окно диаграммы (diagram window) - используется для просмотра и редактирования одной или нескольких диаграмм UML. В нем показано, как выглядят диаграммы UML-модели. При внесении в элементы диаграммы изменений Rose автоматически обновит браузер. Аналогично при внесении изменений в элемент с помощью браузера Rose автоматически обновит соответствующие диаграммы. Это помогает поддерживать модель в непротиворечивом состоянии.

Журнал (log) - применяется для просмотра ошибок и отчетов о выполнении различных команд. По мере работы над моделью определенная информация будет направляться в окно

журнала. Например, туда помещаются сообщения об ошибках, возникающих при генерации кода. Не существует способа закрыть журнал совсем, но его окно может быть минимизировано.

На рис.1 показаны различные части интерфейса Rose.

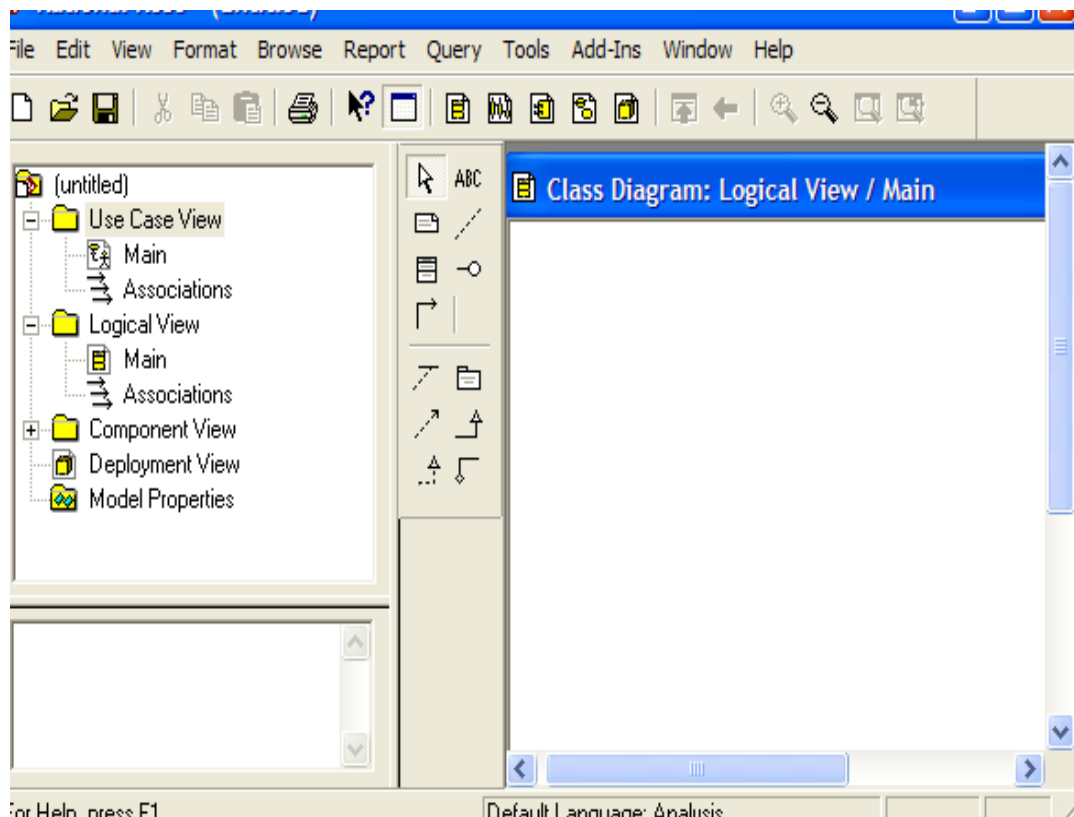


Рис.1. Интерфейс Rose

Четыре представления модели **Rose**

В модели Rose поддерживаются четыре представления - это представление вариантов использования, логическое представление, представление компонентов и представление размещения. Каждое из них предназначено для своих целей.

Представление вариантов использования содержит всех действующих лиц, все варианты использования и их диаграммы для конкретной системы. Оно может также содержать некоторые диаграммы последовательности и кооперативные диаграммы. На рис.2 изображено представление вариантов использования в браузере Rose.

Представление вариантов использования содержит:

1. Действующих лиц.
2. Варианты использования.
3. Документацию по вариантам использования, описывающую происходящие в них процессы (потoki событий), включая обработку ошибок. Эта пиктограмма соответствует внешнему файлу, прикрепленному к модели Rose.

4. Диаграммы вариантов использования. Обычно у системы бывает несколько таких диаграмм, каждая из которых показывает подмножество действующих лиц и/или вариантов использования.

5. Пакеты, являющиеся группами вариантов использования и/или действующих лиц.

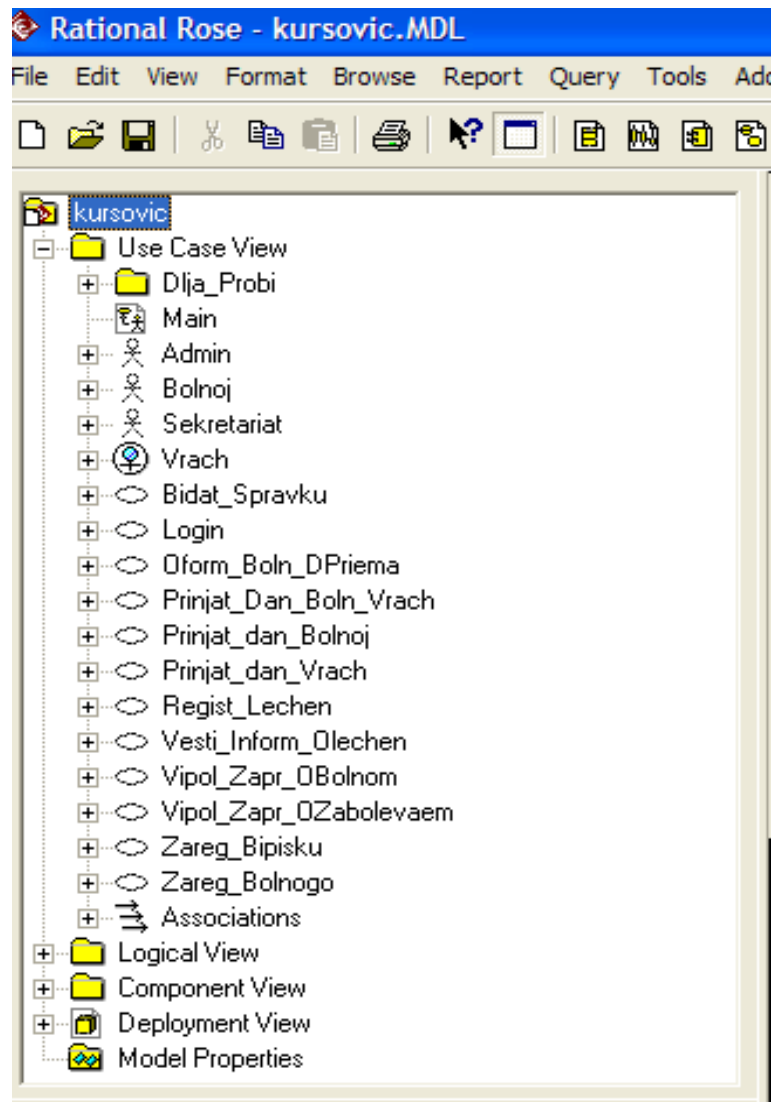


Рис.2. Представление вариантов использования

Логическое представление (рис. 3) показывает, как система будет реализовывать поведение, описанное в вариантах использования. Оно дает подробную картину составных частей системы и описывает взаимодействие этих частей. Логическое представление включает конкретные классы, диаграммы классов и диаграммы состояний. С их помощью конструируется детальный проект создаваемой системы.

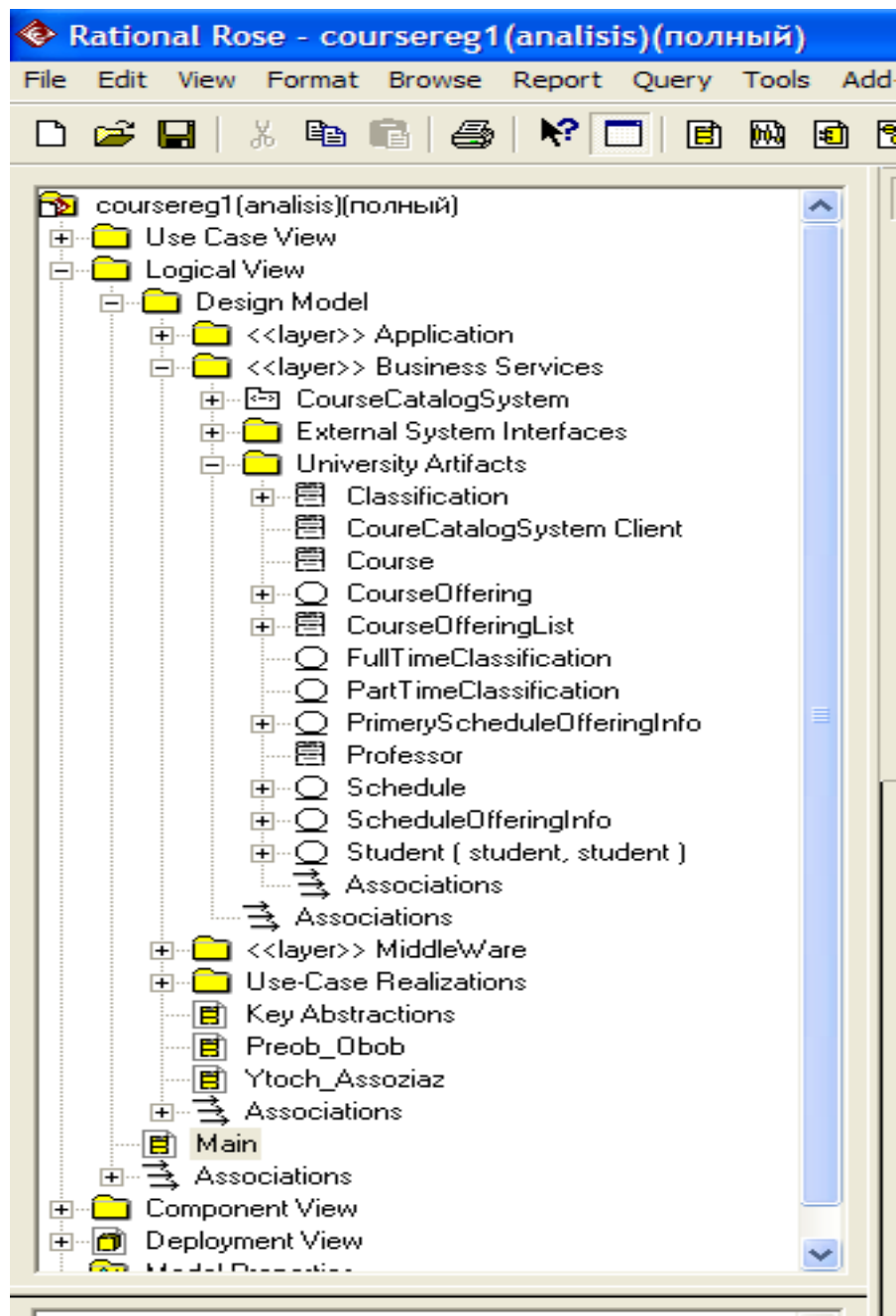


Рис. 3 Логическое представление системы

Логическое представление содержит:

1. Классы.
2. Диаграммы классов. Как правило, для описания системы используется несколько диаграмм классов, каждая из которых отображает некоторое подмножество всех классов системы.
3. Диаграммы взаимодействия, применяемые для отображения объектов, участвующих в одном потоке событий варианта использования.
4. Диаграммы состояний.
5. Пакеты, являющиеся группами взаимосвязанных классов.

Представление компонентов содержит:

1. Компоненты, являющиеся физическими модулями кода.

2. Диаграммы компонентов.
3. Пакеты, являющиеся группами связанных компонентов.

Представление размещения - это последнее представление Rose. Оно соответствует физическому размещению системы, которое может отличаться от ее логической архитектуры.

В представление размещения входят:

1. Процессы, являющиеся потоками (threads), исполняемыми в отведенной для них области памяти.
2. Процессоры, включающие любые компьютеры, способные обрабатывать данные. Любой процесс выполняется на одном или нескольких процессорах.
3. Устройства, т.е. любая аппаратура, не способная обрабатывать данные (например, терминалы ввода-вывода и принтеры).
4. Диаграмма размещения.

Параметры настройки отображения (изображение атрибутов и операций на диаграммах классов)

В Rose имеется возможность настроить диаграммы классов так, чтобы:

1. Показывать все атрибуты и операции.
2. Скрыть операции/ Скрыть атрибуты.
3. Показывать только некоторые атрибуты или операции.
4. Показывать операции вместе с их полными сигнатурами или только их имена.
5. Показывать или не показывать видимость атрибутов и операций.
6. Показывать или не показывать стереотипы атрибутов и операций.

Значения каждого параметра по умолчанию можно задать с помощью окна, открываемого при выборе пункта меню Tools > Options.

Существуют два способа изменения параметров представления атрибутов на диаграмме. Можно установить нужные значения у каждого класса индивидуально. Можно также изменить значения нужных параметров по умолчанию до начала создания диаграммы классов. Внесенные таким образом изменения повлияют только на вновь создаваемые диаграммы.

Для переключения между нотациями видимости Rose и UML:

1. В меню модели выберите пункт Tools > Options.
2. Перейдите на вкладку Notation.
3. Для переключения между нотациями воспользуйтесь переключателем Visibility as Icons.

Если этот переключатель помечен, будет использоваться нотация Rose, в противном случае - нотация UML. Изменение этого параметра повлияет только на новые диаграммы. Существующие диаграммы классов останутся прежними.

Практическая работа № 4

Создание модели вариантов использования

1. ЦЕЛЬ РАБОТЫ

Построить диаграмму вариантов использования для определения основных функций будущей системы.

2. ПРИБОРЫ И ОБОРУДОВАНИЕ

ПЭВМ типа IBM PC/XT

Case-средство Rational Rose.

3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Создайте следующие действующие лица в среде Rational Rose:

- Student (Студент) - записывается на курсы.
- Professor (Профессор) - выбирает курсы для преподавания.
- Registrar (Регистратор) - формирует учебный план и каталог курсов, ведет все данные о курсах, профессорах и студентах.
- Billing System (Расчетная система) - получает от данной системы информацию по оплате курсов.
- Course Catalog (Каталог курсов) - передает в систему информацию из каталога курсов, предлагаемых университетом.

Для того чтобы поместить действующее лицо в браузер:

1. Щелкните правой кнопкой мыши по пакету представления вариантов использования в браузере.
2. Выберите пункт New > Actor в открывшемся меню.
3. В браузере появится новое действующее лицо под названием NewClass. Слева от его имени вы увидите пиктограмму действующего лица UML.
4. Выделив новое действующее лицо, введите его имя.
5. После создания действующих лиц сохраните модель под именем courserereg (analysis) с помощью пункта меню File > Save.

Определите варианты использования.

Исходя из потребностей действующих лиц выделяются следующие варианты использования:

- Login (Войти в систему).
- Register for Courses (Зарегистрироваться на курсы).

- View Report Card (Просмотреть таблицу успеваемости).
- Select Courses to Teach (Выбрать курсы для преподавания).
- Submit Grades (Проставить оценки).
- Maintain Professor Information (Вести информацию о профессорах).
- Maintain Student Information (Вести информацию о студентах).
- Close Registration (Закрыть регистрацию).

Создайте варианты использования в среде Rational Rose.

Для того чтобы поместить вариант использования в браузер:

1. Щелкните правой кнопкой мыши по пакету представления вариантов использования в браузере.

2. Выберите в появившемся меню пункт New > Use Case.

3. Новый вариант использования под названием NewUseCase появится в браузере.

Слева от него будет видна пиктограмма варианта использования UML.

4. Выделив новый вариант использования, введите его название.

5. Результат выполнения задания показан на рис.1.

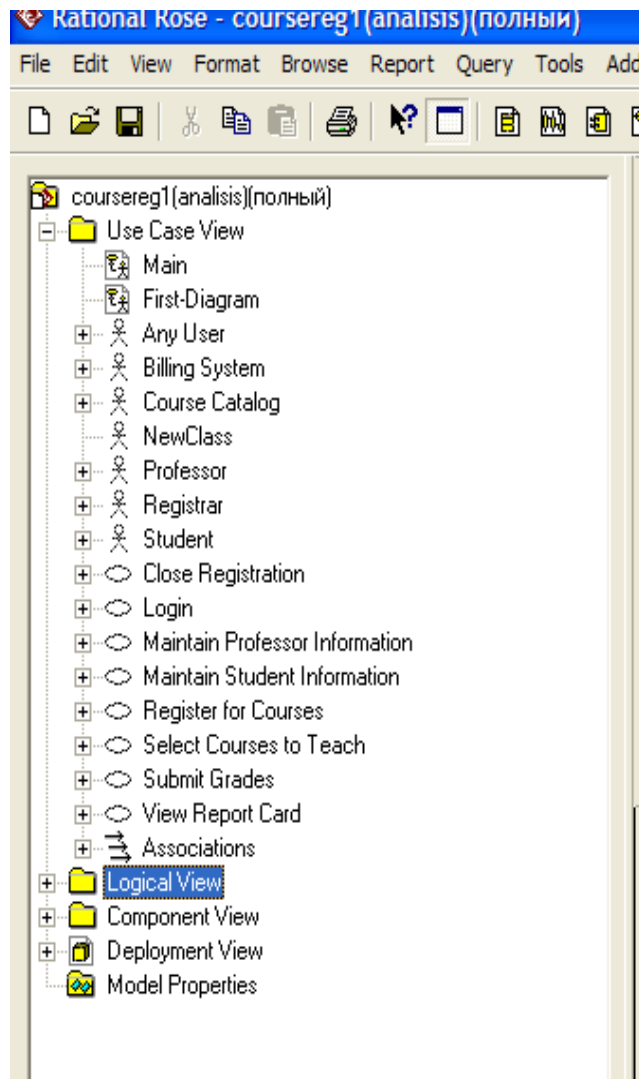


Рис.1. Представление вариантов использования в браузере

Создайте диаграмму вариантов использования для системы регистрации. Требуемые для этого действия подробно перечислены далее. Готовая диаграмма вариантов использования изображена на рис.2.

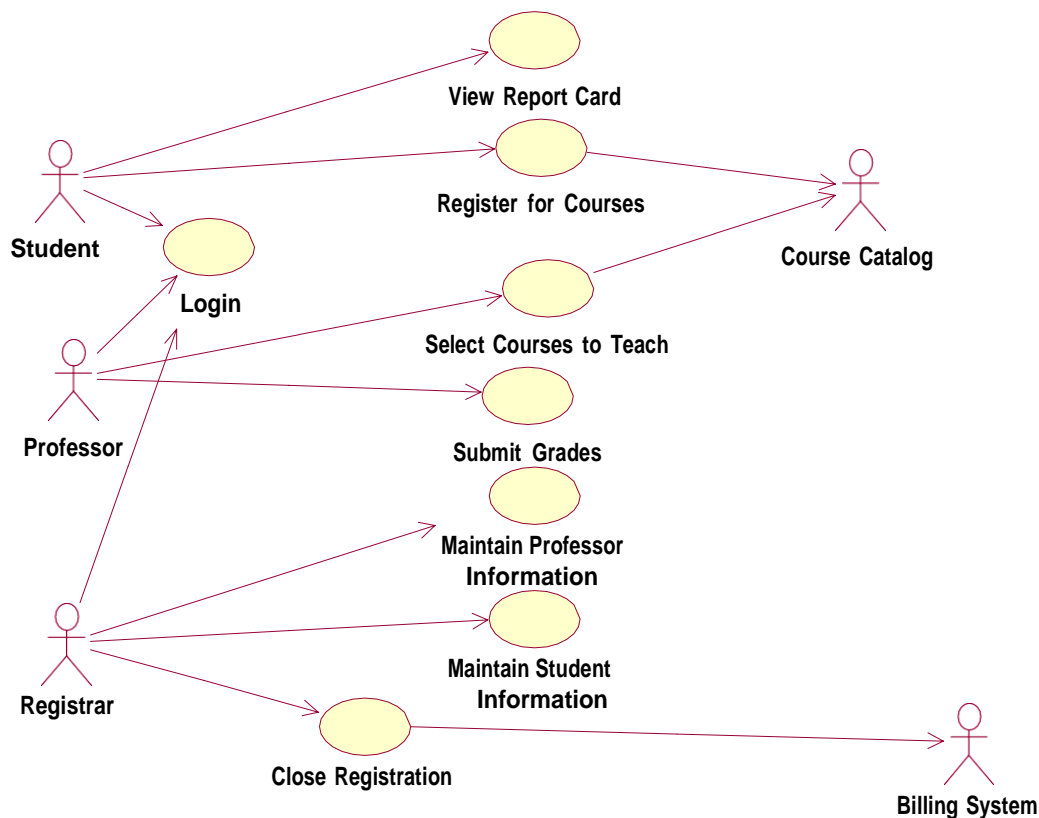


Рис.2 Диаграмма вариантов использования для системы регистрации

В среде Rose диаграммы вариантов использования создаются в представлении вариантов использования. Главная диаграмма (Main) предлагается по умолчанию. Для моделирования системы можно затем разработать необходимое количество дополнительных диаграмм.

Для того чтобы получить доступ к главной диаграмме вариантов использования:

1. Откройте данное представление, щелкнув по значку «+» рядом с представлением вариантов использования в браузере.
2. Откройте главную диаграмму, дважды щелкнув мышью. Строка заголовка изменится, включив фразу [Use Case Diagram: Use Case view / Main].

Для создания новой диаграммы вариантов использования:

1. Щелкните правой кнопкой мыши по пакету представления вариантов использования в браузере.

2. Выберите пункт New > Use Case Diagram из всплывающего меню.
3. Выделив новую диаграмму, введите ее имя.
4. Дважды щелкните по названию этой диаграммы в браузере, чтобы открыть ее.

Постройте диаграмму вариантов использования

Откройте диаграмму вариантов использования Main.

Перетащите действующее лицо или вариант использования мышью из браузера на диаграмму вариантов использования.

С помощью кнопки Unidirectional Association (Однонаправленная ассоциация) панели инструментов нарисуйте ассоциации между действующими лицами и вариантами использования.

Наличие общего варианта использования Login для трех действующих лиц позволяет обобщить их поведение и ввести новое действующее лицо Any User. Модифицированная диаграмма вариантов использования показана на рис.3.

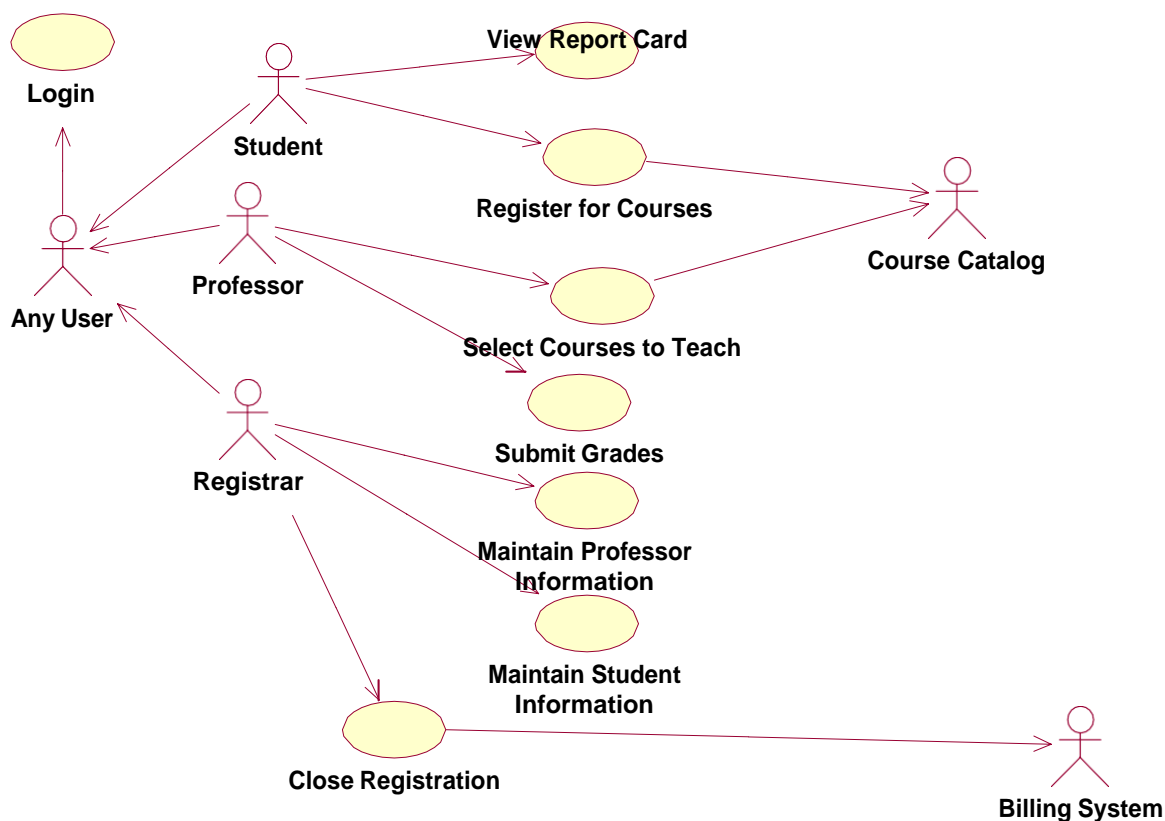


Рис.3. Модифицированная диаграмма вариантов использования

Добавьте описания к вариантам использования.

1. Выделите в браузере вариант использования Register for Courses.

2. В окне документации введите следующее описание к этому варианту использования: «This use case allows a student to register for courses in the current semester» («Этот вариант использования дает студенту возможность зарегистрироваться на курсы в текущем семестре»).

3. Создайте с помощью MS Word три текстовых файла с описаниями вариантов использования Login (Войти в систему), Register for Courses (Зарегистрироваться на курсы) и Close Registration (Закрыть регистрацию) (приложение А).

ПРИЛОЖЕНИЕ А

Вариант использования **Login**

Краткое описание. Данный вариант использования описывает вход пользователя в систему регистрации курсов.

Основной поток событий

Данный вариант использования начинает выполняться, когда пользователь хочет войти в систему регистрации курсов.

1. Система запрашивает имя пользователя и пароль.
2. Пользователь вводит имя и пароль.
3. Система проверяет имя и пароль, после чего открывается доступ в систему.

Альтернативные потоки

Неправильное имя/пароль. Если во время выполнения Основного потока обнаружится, что пользователь ввел неправильное имя и/или пароль, система выводит сообщение об ошибке. Пользователь может вернуться к началу Основного потока или отказаться от входа в систему, при этом выполнение варианта использования завершается.

Предусловия

Отсутствуют.

Постусловия

Если вариант использования выполнен успешно, пользователь входит в систему. В противном случае состояние системы не изменяется.

Вариант использования **Register for Courses**

Краткое описание. Данный вариант использования позволяет студенту зарегистрироваться на конкретные курсы в текущем семестре. Студент может изменить свой выбор (обновить или удалить курсы), если изменение выполняется в установленное время в начале семестра. Система каталога курсов предоставляет список всех конкретных курсов текущего семестра.

Основной поток событий

Данный вариант использования начинает выполняться, когда студент хочет зарегистрироваться на конкретные курсы или изменить свой график курсов.

1. Система запрашивает требуемое действие (создать, обновить, удалить график).
2. Когда студент указывает действие, выполняется один из подчиненных потоков (создать, обновить, удалить или принять график).

Создать график

1. Система выполняет поиск в каталоге курсов доступных конкретных курсов и выводит их список.

2. Студент выбирает из списка 4 основных и 2 альтернативных курса.
3. После выбора система создает график студента.
4. Выполняется подчиненный поток «Принять график».

Обновить график

1. Система выводит текущий график студента.
2. Система выполняет поиск в каталоге курсов доступных конкретных курсов и

выводит их список.

3. Студент может обновить свой выбор курсов, удаляя или добавляя конкретные курсы.
4. После выбора система обновляет график.
5. Выполняется подчиненный поток «Принять график».

Удалить график

1. Система выводит текущий график студента.
2. Система запрашивает у студента подтверждения удаления графика.
3. Студент подтверждает удаление.
4. Система удаляет график. Если график включает конкретные курсы, на которые

записался студент, он должен быть удален из списков этих курсов.

Принять график

Для каждого выбранного, но еще не «зафиксированного» конкретного курса в графике система проверяет выполнение студентом предварительных требований (прохождение определенных курсов), факт открытия конкретного курса и отсутствие конфликтов графика. Затем система вносит данные о студенте в список выбранного конкретного курса. Курс фиксируется в графике, и график сохраняется в системе.

Альтернативные потоки

Сохранить график

В любой момент студент может вместо принятия графика сохранить его. В этом случае шаг «Принять график» заменяется на следующий:

1. «Незафиксированные» конкретные курсы помечаются в графике как «выбранные».
2. График сохраняется в системе.

Не выполнены предварительные требования, курс заполнен или имеют место конфликты графика.

Если во время выполнения подчиненного потока «Принять график» система обнаружит, что студент не выполнил необходимые предварительные требования, или выбранный им конкретный курс заполнен, или имеют место конфликты графика, то выдается сообщение об ошибке. Студент

может либо выбрать другой конкретный курс и продолжить выполнение варианта использования, либо сохранить график, либо отменить операцию, после чего основной поток начнется с начала.

График не найден. Если во время выполнения подчиненных потоков «Обновить график» или «Удалить график» система не может найти график студента, то выдается сообщение об ошибке. После того как студент подтвердит это сообщение, основной поток начнется с начала.

Система каталога курсов недоступна. Если окажется, что невозможно установить связь с системой каталога курсов, то будет выдано сообщение об ошибке. После того как студент подтвердит это сообщение, вариант использования завершится.

Регистрация на курсы закончена. Если в самом начале выполнения варианта использования окажется, что регистрация на текущий семестр закончена, будет выдано сообщение, и вариант использования завершится.

Удаление отменено. Если во время выполнения подчиненного потока «Удалить график» студент решит не удалять его, удаление отменяется, и основной поток начнется сначала.

Предусловия

Перед началом выполнения данного варианта использования студент должен войти в систему.

Постусловия

Если вариант использования завершится успешно, график студента будет создан, обновлен или удален. В противном случае состояние системы не изменится.

Вариант использования **Close Registration**

Краткое описание. Данный вариант использования позволяет регистратору закрывать процесс регистрации. Конкретные курсы, на которые не записалось достаточного количества студентов, отменяются. В расчетную систему передается информация о каждом студенте по каждому конкретному курсу, чтобы студенты могли оплатить курсы.

Основной поток событий

Данный вариант использования начинает выполняться, когда регистратор запрашивает прекращение регистрации.

1. Система проверяет состояние процесса регистрации. Если регистрация еще выполняется, выдается сообщение, и вариант использования завершается.
2. Для каждого конкретного курса система проверяет, ведет ли его какой-либо профессор и записалось ли на него не менее трех студентов. Если эти условия выполняются, система фиксирует конкретный курс в каждом графике, который включает данный курс.
3. Для каждого студенческого графика проверяется наличие в нем максимального количества основных курсов; если их недостаточно, система пытается дополнить

альтернативными курсами из списка данного графика. Выбирается первый доступный альтернативный курс. Если таких курсов нет, то никакое дополнение не происходит.

4. Система закрывает все конкретные курсы. Если в каком-либо конкретном курсе оказывается менее трех студентов (с учетом добавлений, сделанных в п.3), система отменяет его и исключает из каждого содержащего его графика.

5. Система рассчитывает плату за обучение для каждого студента в текущем семестре и направляет информацию в расчетную систему. Расчетная система посылает студентам счета для оплаты с копией их окончательных графиков.

Альтернативные потоки

Конкретный курс никто не ведет. Если во время выполнения основного потока обнаруживается, что некоторый конкретный курс не ведется никаким профессором, то этот курс отменяется. Система исключает данный курс из каждого содержащего его графика.

Расчетная система недоступна. Если невозможно установить связь с расчетной системой, спустя некоторое установленное время система вновь попытается связаться с ней. Попытки будут повторяться до тех пор, пока связь не установится.

Предусловия

Перед началом выполнения данного варианта использования регистратор должен войти в систему.

Постусловия

Если вариант использования завершится успешно, регистрация закрывается. В противном случае состояние системы не изменится.

Упражнение 5. Прикрепление файла к варианту использования

Щелкните правой кнопкой мыши по варианту использования.

В открывшемся меню выберите пункт Open Specification.

Перейдите на вкладку файлов.

Щелкните правой кнопкой мыши по белому полю и из открывшегося меню выберите пункт Insert File.

Укажите созданный ранее файл и нажмите на кнопку Open, чтобы прикрепить файл к варианту использования.

Удаление вариантов использования и действующих лиц. Существуют два способа удалить элемент модели - из одной диаграммы или из всей модели. Для удаления элемента модели из диаграммы:

1. Выделите элемент на диаграмме.

2. Нажмите на клавишу Delete.

3. Обратите внимание, что хотя элемент и удален с диаграммы, он остался в браузере и на других диаграммах системы.

Для удаления элемента из модели:

1. Выделите элемент на диаграмме.

2. Выберите пункт меню Edit > Delete from Model или нажмите сочетание клавиш CTRL + D.

4 Перечень лабораторных работ

Номер практического занятия	Наименование темы лабораторной работы (практического занятия)
Лабораторная работа №1.	Создание модели данных с помощью Erwin 4.0
Лабораторная работа №2.	Создание уровней модели и сущностей, входящих в объектную область
Лабораторная работа №3.	Внесение атрибутов сущностей объектной области в схему и задание их на диаграмме. Генерация базы данных
Лабораторная работа №4.	Соответствие логической модели Erwin и модели процессов BPwin
Лабораторная работа №5.	Введение в Rational Rose
Лабораторная работа №6.	Создание модели вариантов использования. Анализ системы
Лабораторная работа №7.	Диаграммы взаимодействия
Лабораторная работа №8.	Построение диаграммы планов с операциями анализа. Проектирование системы
Лабораторная работа №9.	Проектирование классов
Лабораторная работа №10.	Проектирование базы данных

Перечень вопросов к Экзамену

1. Rational Rose – инструмент логического проектирования программ.
2. UML – средства описания проекта на логической стадии разработки.
3. Базы данных. Основные понятия.
4. Базы знаний.
5. Виды инструментальных средств.
6. Диаграмма взаимодействия ПО, как способ выражение сценария ПО.
7. Диаграмма классов: структура, состав, связи.
8. Диаграмма компонентов для объектно-ориентированной системы и web-системы.
9. Диаграмма коопераций: определение, идеология, структура, пример.
10. Диаграмма последовательностей: определение, структура, состав, пример.
11. Диаграмма развертывания и архитектура ПО: сходство и отличие.
12. Диаграмма развертывания: назначение, структура, пример.
13. Диаграмма состояний: определение, назначение, структура, пример.
14. Идеологический смысл технического задания.
15. Установка и установка программных систем – проблемы, пути решения, инструменты.
16. Инструментальные средства разработки программного обеспечения (ПО).
17. Информационный поиск. Модели поиска. Стратегии поиска.
18. История развития программного инструмента.
19. Качество ПО.
20. Классификация направлений программирования и их особенностей.
21. Классификация стандартов программирования.
22. Логическая форма графического описания взаимодействия активных объектов системы.
23. Методы разработки программы.
24. Моделирование программного обеспечения.
25. Модель данных "сущность–связь".
26. Модель: определение, классификация, пример.
27. Общие требования технического задания на разработку ПО.
28. Оптимизация программных продуктов – методы и инструменты.
29. Отличие идеологии разработки от цели разработки ПО.
30. Парадигмы связывания и видимости объектов – глобальные и локальные, статические и динамические, внутренние и внешние - методы и инструменты реализации.
31. Перспективы развития инструментальных средств.
32. Полнофункциональность и целостность ПО.
33. Понятие концептуальной, логической, физической структуры БД.
34. Понятие модели данных.
35. Последовательность действий при разработке программ.
36. Построение контекстной помощи – средства и методики.
37. Психологические особенности разработки ПО.
38. Разработка технического задания.
39. Реляционная модель данных.
40. Современное программирование – базовые понятия и инструменты.
41. Современные технологии разработки ПО.
42. Современные языки программирования ПО.
43. Специфические требования технического задания.

44. Сравнение возможностей пакетов программирования баз данных.
45. Сравнение возможностей систем управления базы данных.
46. Сравнение диаграммы деятельности и алгоритма работы программы.
47. Сравнение диаграммы классов и структуры базы данных.
48. Сравнение диаграммы объектов и диаграммы компонент.
49. Сравнения возможностей объектных языков программирования.
50. Тестирование и отладка ПО.
51. Техничко-экономическое обоснование ПО.
52. Технические требования к разработке ПО.
53. Требования, предъявляемые к разработке ПО.
54. Файл - менеджеры – программы управления файлами при разработке – возможности и их наращивание, разнообразие и характеристики использования.
55. Целостность и защита данных. Структуры БД.
56. Экономические требования разработки ПО.
57. Этап выработки требований к программе - методы и инструменты.
58. Перспективы инструментальных средств

Список литературы

1. Леффингуал, Дин, Ундри, Дон Принципы работы с требованиями к ПО. Унифицированный подход. М., 2002г.
2. Сэм Канер и др. Тестирования программного обеспечения. Киев, 2000 г.
3. А. Якобсон, Г. Буч, Дж. Рамбо Унифицированный процесс разработки ПО, М., 2000 г.
4. Крэг Ларман, Применение UML и шаблонов проектирования. М., Вильямс, 2001 г.
5. Д. Шниер, Толковый словарь компьютерных технологий. М.:, 2002 г.
6. Стив Тексейра и Ксавье Пачеко, Delphi 7., Руководство разработчика Т 1, 2 . Вильямс 2006г.
7. Дж. Шмюллер, Освой самостоятельно UML 2.0. М., 2006 г.
8. Орлов С.А., Технологии разработки программного обеспечения. Питер, 2002г.
9. Р. Денис Гиббс, Управление проектами с помощью IBM Rational Unified Process, М.: КУДИЦ-ПРЕСС, 2007 г.
10. Терри Кватрани, Джим Палистрант Визуальное моделирование с помощью IBM Rational Sostware Architect и UML. М.: КУДИЦ-ПРЕСС, 2007 г.
11. Луиза Тампе, Введение в тестирование программного обеспечения. М.: Вильямс, 2003 г.
12. Иан Соммервилл, Инженерия программного обеспечения. М.: Вильямс, 2002 г.