

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Бадриев Нарик Тимурович

Должность: Ректор

Дата подписания: 07.07.2025 17:04:03

Уникальный программный ключ:

5cf0d6f89e80f49a334f6a4ba58e91f3326b9926

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
ФГБОУ ВО
«ДАРЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»

Кафедра ИТиПИвЭ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к выполнению курсового проекта
по дисциплине:
«Проектирование информационных систем»

для студентов направления подготовки бакалавров
09.03.03 – «Прикладная информатика»

Махачкала-2019

УДК [004.415.2:33](075.8)

Методические указания к выполнению курсового проекта по дисциплине «Проектирование информационных систем» для студентов направления подготовки бакалавров 09.03.03 – «Прикладная информатика». Махачкала, ДГТУ, 2019, 30 с.

Методические указания предназначены для закрепления теоретических и практических навыков студентов по дисциплине «Проектирование информационных систем» при проектировании информационных систем. В методических указаниях приведены цели и задачи курсового проекта, краткое описание средств проектирования, индивидуальные задания для студентов, структура пояснительной записки и основные требования по ее оформлению.

Указания предназначены для студентов дневной и заочной форм обучения для студентов направления подготовки бакалавров 09.03.03 – «Прикладная информатика», изучающих дисциплину «Проектирование информационных систем». Данная дисциплина требует предварительного изучения курсов: «Информатика и программирование», «Теория экономических информационных систем», «Базы данных», «Информационные технологии в экономике».

Составители: доцент кафедры ИТиПИВЭ, к.э.н. Мурадов М.М.
доцент кафедры ИТиПИВЭ, к.э.н. Адеева М.Г.

Рецензент: Мирземагомедова М.М., к.т.н., доцент кафедры ИТиПИВЭ,
Джабраилов Х.С., к.э.н., директор ООО ИВЦ Сигма.

Печатается по решению Ученого совета Дагестанского государственного
технического университета от _____ 2019г.

Введение

Целью методических указаний является привитие навыков использования современных методов и средств проектирования информационных систем, основанных на применении CASE-технологий.

Создание современных информационных систем представляет собой сложнейшую задачу, решение которой требует применения специальных методик и инструментов. Неудивительно, что в последнее время среди системных аналитиков и разработчиков значительно вырос интерес к CASE - технологиям и инструментальным CASE - средствам, позволяющим максимально систематизировать и автоматизировать все этапы разработки программного обеспечения

В ходе выполнения курсового проекта изучается процесс создания информационной системы и в результате разрабатывается: функциональная модель организации в нотациях IDEF0, IDEF3, DFD, информационная модель в нотации IDEF1X, диаграммы FEO, диаграммы дерева узлов, функционально-стоимостной анализ статей затрат по работам организации. Кроме того создается клиентское приложение по работе с БД в одной из СУБД-ориентированной среде, например таких как: Borland C++ Builder, Visual FoxPro, MS Access и др..

1. Цель и задачи курсового проектирования

Целью курсового проектирования является изучение процесса создания информационной системы для моделирования и автоматизации текущих бизнес-процессов, освоение технологии проектных работ, выбор и обоснование проектных решений, развитие навыков самостоятельной работы.

Целью данного курсового проектирования является закрепление и расширение знаний, полученных на лекциях, лабораторных и практических занятиях по принципам построения информационной системы с использованием современных ИТ.

Курсовой проект выполняется с использованием инструментов визуального моделирования бизнес-процессов BPwin 7.1 и баз данных ERwin 7.1 автоматизированным способом и СУБД- ориентированной среды Visual FoxPro 9.0.

Курсовой проект посвящен системному этапу проектирования информационной системы и включает в себя задачи:

- ⇒ разработку концепции информационной системы;
- ⇒ анализ функций и построение дерева функций;
- ⇒ покрытие дерева функций функциональными модулями и построение структуры информационной системы в BPwin 7.1;
- ⇒ описание модели функционирования;
- ⇒ создание логической и физической моделей данных в ERwin 7.1;
- ⇒ создание приложения в СУБД -ориентированной среде Visual FoxPro 9.0;
- ⇒ описание функционирования информационной системы.

2. Выбор и закрепление темы курсового проекта

Область экономической/юридической деятельности выбирается из перечня примерной тематики курсовых проектов, приведенной в главе 5.

Темы курсовых проектов выбираются студентами по желанию, на основе имеющегося опыта работы, теоретической и практической подготовки. Студент может предложить свою тему работы, согласовав предварительно ее название и содержание с преподавателем (руководителем).

Выбор темы студентом производится в течение первых двух недель с момента начала занятий в том семестре, в котором предусмотрено выполнение курсового проекта. Факт выбора тем студентами академической группы и их закрепление оформляются в виде табл. 1, которая составляется в двух экземплярах: один хранится на кафедре, другой должен находиться у руководителя курсовых проектов.

В процессе выполнения курсового проекта руководитель оказывает методическую помощь студенту в определении перечня вопросов, подлежащих освещению в работе. Подборе литературы и справочных материалов, проводит консультации, осуществляет контроль за ходом выполнения курсового проекта, в целом и на стадии ее отдельных этапов.

Таблица 1

СПИСОК

студентов _____ группы факультета информационных систем ДГТУ, получивших темы курсового проектирования по дисциплине «Проектирование информационных систем»

№№ п/п	Название темы	Фамилия и инициалы студента	Дата получения темы	Подпись студента	Примечание
1					
...					
N					

Руководитель курсовых проектов _____
(подпись и ФИО преподавателя)

3. Содержание и порядок оформления пояснительной записки

Пояснительная записка по курсовому проектированию должна быть выполнена в машинописном виде с использованием ПК. Вместе с отпечатанной пояснительной запиской на кафедру представляется также ее электронная копия на дискете (doc - файл и папка с самим проектом).

Для оформления текстовой части курсового проекта необходимо создать макеты следующих документов при помощи инструмента моделирования бизнес-процессов BPwin 7.1 и моделирования БД ERwin 7.1.

Внимание! Обязательно соблюдать структуру и порядок пояснительной записки, а также правила ее оформления.

Структура пояснительной записки

Титульный лист

Содержание

Постановка задачи

Введение

Глава 1. Экономическая сущность комплекса задач

1.1. Исследование предметной области ИС.

1.2. Описание модели функционирования ИС.

1.2.1. Анализ возможностей методологии и инструментальных средств проектирования заданной ИС

1.2.2. Контекстная диаграмма

1.2.3. Диаграммы декомпозиции в методологии IDEF0

1.2.4. Диаграммы декомпозиции в методологии DFD

1.2.5. Диаграммы декомпозиции в методологии IDEF3¹

1.2.6. Функционально-стоимостной анализ

1.2.7. Диаграмма FEO² и диаграмма дерева узлов

1.2.8. Организационная диаграмма

Глава 2. Информационная система и ее описание

2.1. Информационная модель в нотации IDEF1.X

2.1.1. Логическая модель (модель «сущность-связь»; модель, основанная на ключах; полная атрибутивная модель)

2.1.2. Физическая модель (трансформационная модель, генерация модели СУБД)

2.2. Создание приложения в СУБД ориентированной среде

2.2.1. Описание входной информации

2.2.2. Описание результатной информации

2.2.3. Макеты отображения результатов

2.2.4. Блок-схема основных модулей

Заключение

Список использованной литературы

Приложение

¹ Необходимо создание сценария по какой-либо диаграмме методологии IDEF3.

² Необходимо создание по какой-либо диаграмме FEO-диаграммы (где именно определяет преподаватель на завершающем этапе проектирования).

1. Титульный лист.

Титульный лист является началом пояснительной записки и выполняется согласно СТП УГТУ-УПИ 1–96 (приложение №1).

2. Содержание.

Содержание помещается в начале пояснительной записки, за титульным листом, наглядно характеризует последовательность разделов и их элементов в пояснительной записке с указанием номеров страниц (ГОСТ 2.105–95).

3. Постановка задачи.

Постановка задачи составляется по установленной форме (приложение №2) и помещается за содержанием.

4. Введение.

Введение к пояснительной записке содержит общие сведения о курсовом проекте, цель и задачи курсового проектирования. Необходимо также перечислить вопросы, которые предполагается рассмотреть в проекте и вопросы, которые предполагается решить практически. Во введении также дается краткая характеристика содержания работы. Объем введения 1–3 страницы.

5. Глава 1. «Экономическая сущность комплекса задач».

Эта глава посвящена сущности и содержанию рассматриваемого в проекте комплекса задач. В этой главе необходимо провести анализ возможностей методологий и инструментальных средств проектирования при создании ИС, а также представить описание всех созданных диаграмм. Объем данной главы 15–20 страниц.

6. Глава 2 «Информационная система и ее описание».

В этой главе необходимо разработать информационную модель в нотации IDEF1.X. Создать логическую модель функционирования и на ее основе создать физическую модель данных с последующим экспортом ее в выбранную заранее СУБД-ориентированную среду (Borland C++ Builder, Visual FoxPro и др.). Затем на основании перенесенных данных разработать приложение. Привести описание входной/результатной информации, макеты отображения результатов, компоненты и отчеты выходных документов, блок-схемы основных модулей.

8. Заключение.

В выводах должны быть описаны основные результаты выполненной работы и рекомендации по их практическому использованию. В отличие от основной части проекта заключение и введение не нумеруются.

9. Список использованной литературы.

В нем указывается вся литература, которая была использована в процессе работы над проектом и на которую необходимо сделать ссылки в тексте пояснительной записки. Источники следует расположить по алфавиту.

10. Приложение.

Приложение оформляется как продолжение пояснительной записки на последующих ее листах. В приложении помещаются справочные материалы.

Например, сгенерированные отчеты по проекту в среде BPwin, ERwin; листинг программ и др.

Разработанная информационная система должна содержать не менее 2-х уровней детализации, не включая нулевой уровень, т.е. A1, A1.1, A1.1.1 и т.д. При проектировании информационной системы необходимо использовать все объекты и методологии проектирования ИС, используемые пакетами BPwin 7.1 и ERwin 7.1.

При построении иерархии контекстной диаграммы, диаграмм декомпозиции, а также диаграмм дерева узлов и ФЕО переходить к детализации процессов следует только после определения содержания всех работ, информационных потоков и взаимосвязей между ними.

После построения законченной модели системы необходимо, используя отчет о согласованности с выбранной методологией проверить на ошибки. В полной модели все ее объекты (работы, процессы, хранилища и т.д.) должны быть подробно описаны. В согласованной модели для всех потоков информации и хранилищ должно выполняться правило сохранения информации: все поступающие куда-либо данные должны быть считаны, а все считываемые данные должны быть записаны (таблицы баз данных).

4. Правила оформления пояснительной записки

К форме курсового проекта предъявляются определенные требования:

1. расстояние от левого края страницы до границы текста составляет не менее 30 мм;
2. расстояние от правого края страницы до границы текста составляет не менее 15 мм;
3. расстояние от верхней или нижней строки текста до верхней или нижней границы страницы должны быть не менее 15 мм;
4. абзац должен начинаться на расстоянии 25 мм от левого края страницы.

Не разрешается размещать заголовки и подзаголовки в нижней части страницы, если на ней не помещается более 4-5 строк последующего текста.

При оформлении отдельных глав курсового проекта следует помнить, что каждая глава должна начинаться с новой страницы.

Названия глав, параграфов должны соответствовать оглавлению (содержанию).

Все страницы работы должны быть пронумерованы последовательно арабскими цифрами. Нумерация страниц должна быть сквозной от титульного листа до последнего листа текста, включая иллюстрацию, таблицы, графики, диаграммы и т.д., расположенные внутри текста или после него, а также приложения. На титульном листе, который является первой страницей, номер страницы не ставится, но учитывается при общей нумерации.

Все иллюстрации - не таблицы именуется рисунками, которым присваивается последовательная нумерация, либо сквозь весь текст, либо в пределах главы.

Все рисунки должны иметь полные наименования. Номер и наименование рисунка записывается в строчку под его изображением, например: «Рис.3.

Контекстная диаграмма». Ссылки на рисунки даются по типу: «(См. рис.2)». Рисунки следует размещать сразу после ссылки на них в тексте. Рисунки размещаются таким образом, чтобы их можно было рассматривать без поворота листов.

Если текст курсового проекта иллюстрируется табличным материалом, то размещение его аналогично размещению рисунков. Таблицы также последовательно нумеруются арабскими цифрами в пределах всего материала и в пределах главы. Над правым верхним углом таблицы помещают надпись: «Таблица» с указанием ее порядкового номера, например «Таблица №2». При наличии тематического заголовка после слова «Таблица» пишут ее наименование в строчку над таблицей.

Приложения оформляются как продолжение курсового проекта на последующих его страницах, но в основной листаж не включается.

При включении в проект более одного приложения впереди всех приложений на отдельном листе пишется прописными буквами «Приложение». Затем следуют отдельные приложения. Каждое приложение должно начинаться с новой страницы с указанием в верхнем правом углу первого листа слова «Приложение», например, «Приложение 1» и иметь тематический заголовок (прил.1, 2).

5. Примерная тематика курсовых проектов

1. Система материально-технического снабжения производственного предприятия.
2. Система финансового планирования и управления предприятием.
3. Система финансового планирования и управления торговой фирмы.
4. Система календарно-сетевого планирования и управления производственного предприятия.
5. Система функционально-стоимостного анализа производственного предприятия.
6. Система учета затрат на производство и калькулирование себестоимости продукции производственного предприятия.
7. Автоматизированная система управления межбанковскими электронными переводами денежных средств в торговых организациях.
8. Система учета основных средств и нематериальных активов предприятия.
9. Система учета основных и оборотных производственных фондов на предприятии.
10. Автоматизированная система анализа управления оборотным капиталом.
11. Система осуществления безналичных расчетов платежными поручениями.
12. Система функционирования бухгалтерии на предприятии.
13. Система анализа полезного использования сырья и материалов.
14. Автоматизированная система учета и анализа производственных запасов на предприятии
15. Система финансового анализа итогов деятельности предприятия.
16. Система финансового анализа итогов деятельности торговой фирмы.
17. Система финансового анализа итогов деятельности коммерческого банка.
18. Система организации на предприятии финансового учета.

19. Система управления менеджментом и маркетингом коммерческого банка.
20. Система учета и управления кадрами предприятия.
21. Система учета и управления кадрами государственного учреждения.
22. Система анализа движения активов, собственного и заемного капитала предприятия.
23. Автоматизированная система анализа и прогнозирования бизнес-плана на торговой фирме.
24. Система управления документооборотом производственного предприятия.
25. Система управления документооборотом торговой фирмы.
26. Система управления документооборотом коммерческого банка.
27. Система управления закупками, ведения договоров, обработки заказов торговой фирмы.
28. Система управления продажами через торговый зал с использованием интеллектуальных кассовых аппаратов и кредитных карточек.
29. Система управления кассовыми операциями торговой фирмы/предприятия.
30. Система управления кассовыми операциями в коммерческом банке.

6. Пример разработки проекта ИС

В данном примере рассматривается проект на тему: «Автоматизированная система «Деканат факультета».

Проект реализован с использованием CASE – средств BPwin 7.1 и ERwin 7.1.

CASE – средство BPwin поддерживает три методологии проектирования: IDEF0, DFD, IDEF3.

ПОСТРОЕНИЕ МОДЕЛИ В BPWIN

Проект состоит из диаграмм следующих типов:

- Контекстная диаграмма;
- Диаграмма декомпозиции в методологии IDEF0;
- Диаграмма декомпозиции в методологии DFD;
- Диаграмма декомпозиции в методологии IDEF3;
- Диаграмма сценария;
- Диаграмма для экспозиции (FEO);
- Диаграмма дерева узлов;
- Организационная диаграмма.

Контекстная диаграмма

Контекстная диаграмма является вершиной древовидной структуры диаграмм и представляет собой самое общее описание системы и ее взаимодействия с внешней средой.

Контекстная диаграмма состоит из одной работы, которая называется «Документооборот деканата факультета». Взаимодействие работы с внешним

миром описывается в виде стрелок, которые представляют собой некую информацию и именуются существительными. В данной работе описаны стрелки:

- типа **вход** (Input): стрелка «БД студентов» представляет собой входную информацию.

- типа **выход** (Output): «Личная карточка студентов», «Экзаменационный лист (направление)», «Журнал учета выдачи экзаменационных и зачетных ведомостей», «Журнал регистрации входящих и исходящих документов», «Журнал контроля организации проведения учебных занятий». Они содержат в себе выходную информацию.

- типа **механизм** (Mechanism): «ПО и ТО» - эта стрелка входит в нижнюю грань работы.

- типа **управление** (Control): «Указания из учебного отдела», «Приказы», «Положение по факультету». Они входят в верхнюю грань работы и показывают правила, планы, которыми руководствуется предприятие при автоматизации документов в деканате факультета.

Контекстная (корневая) диаграмма имеет номер А-0 (рис.1).

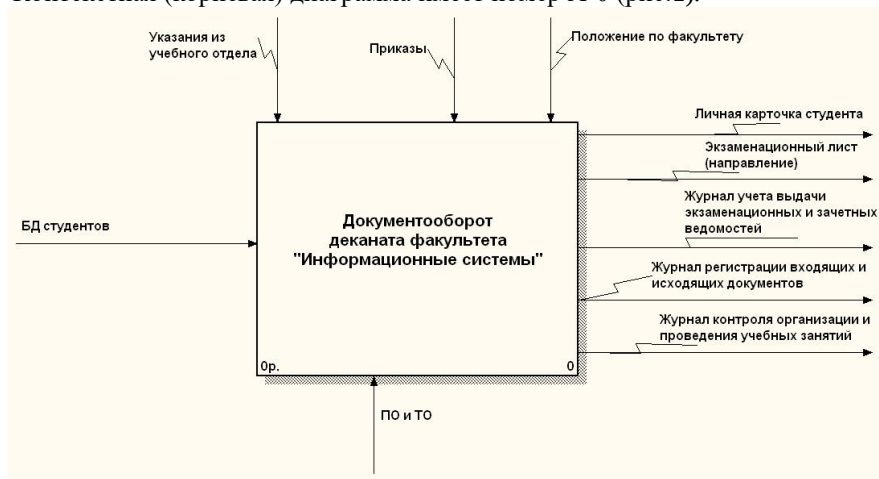


Рис. 1. Контекстная диаграмма

Диаграммы декомпозиции в методологии IDEF0

После описания системы в целом проводится разбиение ее на крупные фрагменты. Этот процесс называется функциональной декомпозицией, а диаграммы, которые описывают каждый фрагмент и взаимодействие фрагментов, называются диаграммами декомпозиции.

IDEFO является наиболее удобным языком моделирования бизнес-процессов, которая представляется как совокупность взаимодействующих работ или функций.

Такая чисто функциональная ориентация является принципиальной - функции системы анализируются независимо от объектов, которыми они оперируют. Это позволяет более четко смоделировать логику и взаимодействие процессов организации.

Контекстная диаграмма разбивается на четыре работы, которые связаны между собой внутренними явными, сливающимися и разветвляющимися стрелками (рис. 2). Также на диаграмме присутствуют граничные стрелки, которые переходят из диаграммы верхнего уровня и связываются с работами.

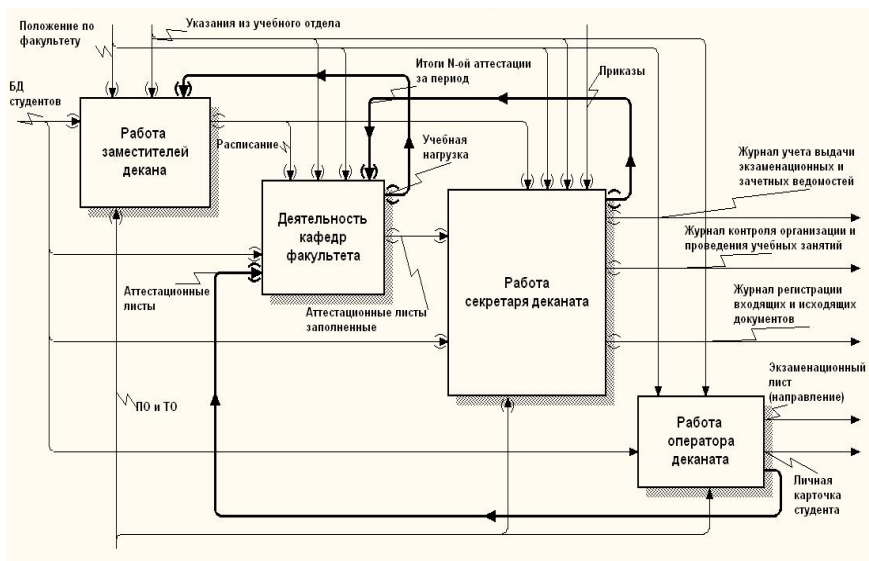


Рис. 2. Диаграмма декомпозиции в методологии IDEF0

Диаграммы декомпозиции в методологии IDEF3

Для описания логики взаимодействия информационных потоков более подходит IDEF3, называемая также методологией моделирования, использующая графическое описание информационных потоков, взаимоотношений между процессами обработки информации и объектов, являющихся частью этих процессов. С их помощью можно описывать сценарии действий сотрудников организации. IDEF3 - это метод, имеющий основной целью дать возможность аналитикам описать ситуацию, когда процессы выполняются в определенной последовательности, а также описать объекты, участвующие совместно в одном процессе.

Для отображения логики взаимодействия стрелок при слиянии и разветвлении или для отображения множества событий в методологии используются перекрестки, которые могут или должны быть завершены перед началом следующей работы.

Также в IDEF3 используются объекты ссылки, которые выражают некую идею, концепцию или данные, которые нельзя связать со стрелкой, перекрестком или работой.

Пример диаграммы декомпозиции в методологии IDEF3 представлен на рис. 3.

На диаграмме присутствуют 6 объектов ссылки: «Расписание», «Указания из учебного отдела», «Положения по факультету», «БД студентов», «Аттестационные листы», «Учебная нагрузка» и «Итоги N-ой аттестации за период». Перекресток типа



«Asynchronous AND» J2 показывает, что один предыдущий процесс должен быть завершен, а все последующие - запущены. Перекресток типа  «Asynchronous AND» J3 показывает, что все предыдущие процессы должны быть завершены, а один последующий - запущен.

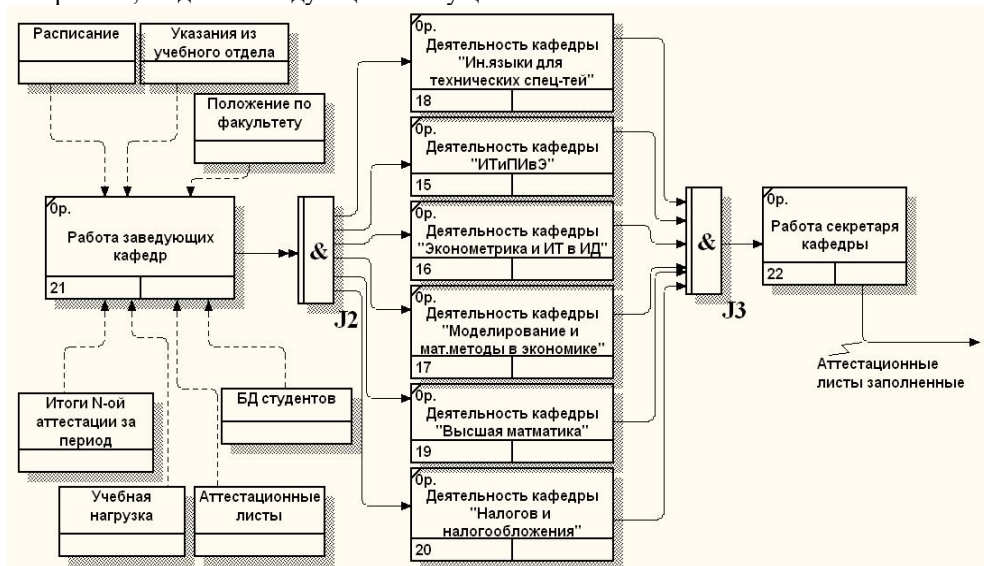


Рис. 3. Диаграмма декомпозиции в методологии IDEF3

Диаграмма сценария

Каждая работа IDEF3 описывает какой-либо сценарий бизнес-процесса и может являться составляющей другой работы. Поскольку сценарий описывает цель и рамки

модели, важно, чтобы работы именовались отглагольным существительным, обозначающим процесс действия, или фразой, содержащей такое существительное. Для создания сценария необходимо из диаграммы декомпозиции удалить работы, стрелки и перекрестки, не входящие в сценарий. На рис. 4 показана диаграмма сценария под номером, созданная на основе диаграммы IDEF3.

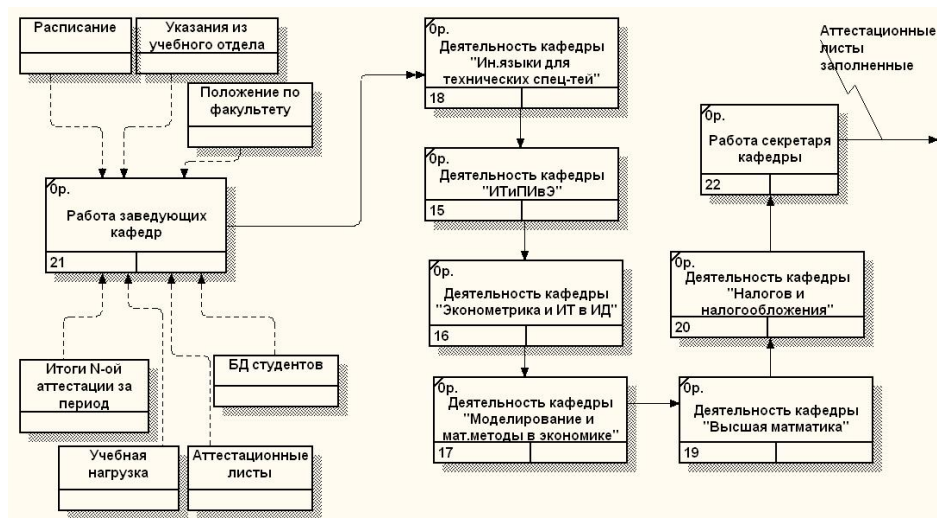


Рис. 4. Диаграмма сценария
Диаграммы потоков данных (Data Flow Diagramming)

Диаграммы потоков данных (Data flow diagramming, DFD) используются для описания документооборота и обработки информации. Подобно IDEF0, DFD представляет модельную систему как сеть связанных между собой работ. Их можно использовать как дополнение к модели IDEF0 для более наглядного отображения текущих операций документооборота в корпоративных системах обработки информации.

На рис. 5. представлена диаграмма потоков данных, которая содержит в себе два хранилища данных: «Студенты» и «Преподаватели», которые позволяют описать данные, необходимые для сохранения в памяти прежде, чем использовать в работах. Внешняя ссылка «Канцелярия», «Учебный отдел», «Кафедры факультета» и «Студенты» являются источниками или приемниками данных извне модели.

Стрелки DFD показывают, как объекты (включая данные) двигаются от одной работы к другой. Это представление потоков совместно с хранилищами данных и внешними сущностями делает модели DFD более похожими на физические характеристики системы - движение объектов (data flow), хранение объектов (data stores), поставка и распространение объектов (external entities).

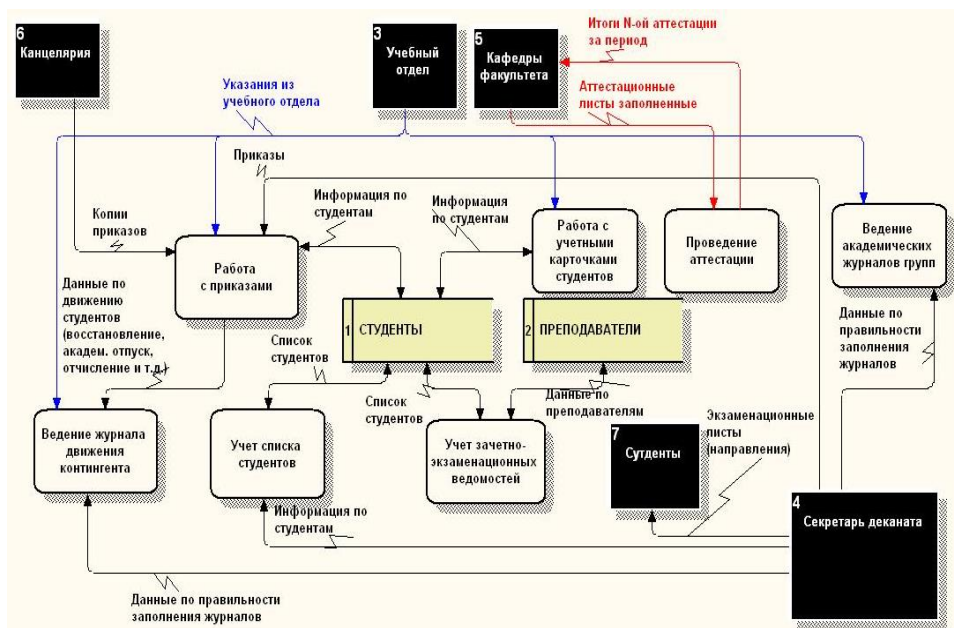


Рис. 5. Диаграмма потоков данных

FEО – диаграмма, диаграмма дерева узлов и организационная диаграмма

Диаграмма дерева узлов показывает иерархию работ в модели и позволяет рассмотреть всю модель целиком, но не показывает взаимосвязи между работами (стрелки). Процесс создания модели работ является итерационным, следовательно, работы могут менять свое расположение в дереве узлов многократно. Чтобы не запутаться и проверить способ декомпозиции, следует после каждого изменения создавать диаграмму дерева узлов.

Диаграммы «только для экспозиции» (FEO) часто используются в модели для иллюстрации других точек зрения, для отображения отдельных деталей, которые не поддерживаются явно синтаксисом IDEF0. Диаграммы FEO позволяют нарушить любое синтаксическое правило, поскольку по сути являются просто картинками – копиями стандартных диаграмм и не включаются в анализ синтаксиса.



Рис. 6. Диаграмма дерева узлов

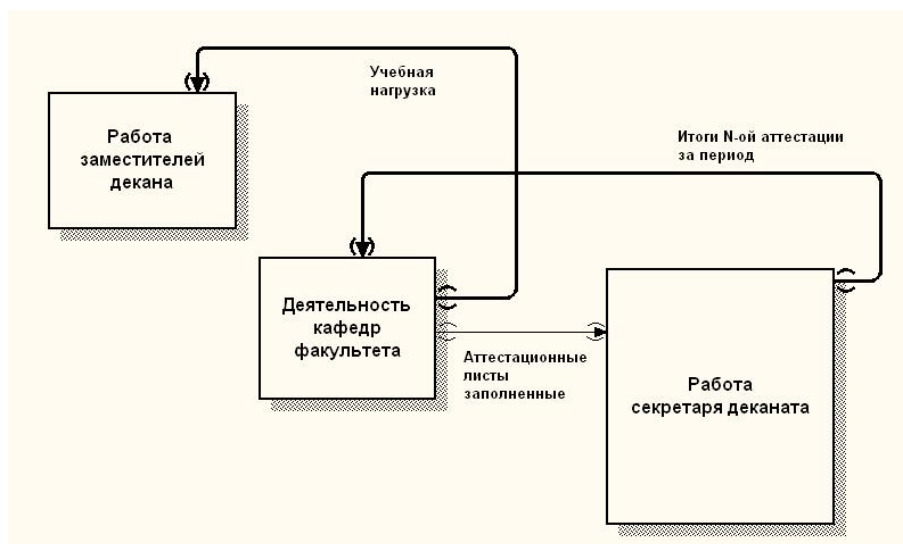


Рис. 7. Диаграмма FEO

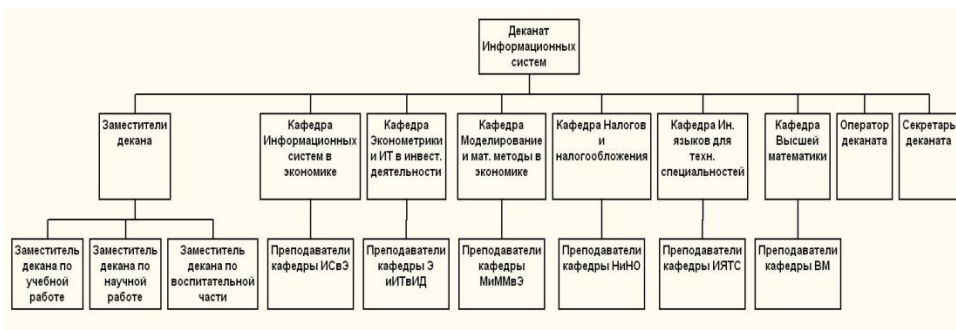


Рис. 8. Организационная диаграмма

ПОСТРОЕНИЕ МОДЕЛЕЙ В ERWIN

Возможны две точки зрения на информационную модель и, соответственно, два уровня модели. Первый – логический уровень (точка зрения пользователя) означает прямое отображение фактов из реальной жизни. Например, люди, столы, отделы, собаки и компьютеры являются реальными объектами. Они именуются на естественном языке, с любыми разделителями слов (пробелы, запятые и т.д.). На физическом уровне модели рассматривается использование конкретной СУБД, определяются типы данных (например, целое или вещественное число), индексы для таблиц.

ERwin предоставляет возможности создавать и управлять этими двумя различными уровнями представления одной диаграммы (модели), равно как и иметь много вариантов отображения на каждом уровне. Термин “логический уровень” в ERwin соответствует концептуальной модели.

Этапы построения информационной модели:

- определение сущностей;
- определение зависимостей между сущностями;
- задание первичных и альтернативных ключей;
- определение атрибутов сущностей;
- приведение модели к требуемому уровню нормальной формы;
- переход к физическому описанию модели: назначение соответствий имя сущности – имя таблицы, атрибут сущности – атрибут таблицы;
- задание триггеров, процедур и ограничений;
- генерация базы данных.

ERwin создает визуальное представление (модель данных) для решаемой задачи. Это представление может использоваться для детального анализа, уточнения и распространения документации, необходимой в цикле разработки. Однако ERwin

далеко не только инструмент для рисования. ERwin автоматически создает базу данных (таблицы, индексы, хранимые процедуры, триггеры для обеспечения ссылочной целостности и другие объекты, необходимые для управления данными).

Логическая модель

Три уровня модели, объединяющие в себя логические модели, состоят из: Entity Relationship Diagram (Диаграмма сущность-связь), the Key-Based (Модель данных, основанная на ключах) и Model the Fully Attributed model (Полная атрибутивная модель).

Верхний уровень состоит из диаграмм сущность - связь и модели данных, основанной на ключах.

Диаграмма сущность-связь является самым высоким уровнем в модели данных и определяет набор сущностей и атрибутов проектируемой системы. Целью этой диаграммы является формирование общего взгляда на систему для её дальнейшей детализации.

Модель данных, основанная на ключах описывает структуру данных системы, в которую включены все сущности и атрибуты, в том числе ключевые. Целью этой модели является детализация модели сущность-связь, после чего модель данных может начать реализовываться.

Нижний уровень состоит из Transformation Model (Трансформационная модель) и Fully Attributed (Полная атрибутивная модель). Трансформационная модель содержит всю информацию для реализации проекта, который может быть частью общей информационной системы и описывает предметную область. Трансформационная модель может проектировать и администрировать БД, представлять какие объекты БД хранятся в словаре данных, и проверить, насколько физическая модель данных удовлетворяет требованиям информационной системы. Фактически из трансформационной модели автоматически можно получить модель СУБД, которая является точным отображением системного каталога СУБД.

Полная атрибутивная модель включает в себя все сущности, атрибуты и является наиболее детальным представлением структуры данных. Полная атрибутивная представляет данные в третьей нормальной форме.

Первым шагом при создании логической модели БД является построение диаграммы ERD («сущность-связь»). ERD-диаграммы состоят из трех частей: сущностей, атрибутов и взаимосвязей. Сущностями являются существительные, атрибуты – прилагательными или модификаторами, взаимосвязи – глаголами.

Модель «сущность-связь» графически представляет структуру данных проектируемой ИС. Сущности отображаются при помощи прямоугольников, содержащих имя. Имена принято выражать существительными в единственном числе, взаимосвязи - при помощи линий, соединяющих отдельные сущности.

Взаимосвязь показывает, что данные одной сущности ссылаются или связаны с данными другой.

Сущность служит для представления набора реальных или абстрактных предметов (людей, мест, событий и т.п.), которые обладают общими атрибутами или характеристиками. Сущность «логический» объект, который в физической среде СУБД представлен таблицей. Сущность в ERwin обычно описывает три части информации: атрибуты, являющиеся первичными ключами, неключевые атрибуты и тип сущности.

Связями называются логические соединения или ассоциации между двумя сущностями.

Данные, относящиеся к связям, очень важны и часто являются критическими данными, которые мы используем в повседневном бизнесе. Например, важно знать о каком-то типе инструмента, но знание того, к кому относится конкретный инструмент (связь между человеком и инструментом) может иметь критическую важность. Связь - это соотношение либо между двумя сущностями, либо между сущностью и этой же сущностью. Связь - «логический» объект, представленный одним или несколькими атрибутами - внешними ключами. Связь в ERwin обычно содержит пять типов информации: тип связи, родительский конец связи, дочерний конец связи, ERwin toolbox содержит два типа сущностей: независимые и зависимые. Независимая СУЩНОСТЬ это сущность, экземпляры которой могут быть уникальным образом идентифицированы без определения ее связи с другой сущностью. Она представляется в ERwin в виде прямоугольника. Первичный ключ независимой сущности не включает в себя первичных ключей других сущностей. Зависимая СУЩНОСТЬ - это сущность, экземпляры которой не могут быть уникальным образом идентифицированы без определения ее связи с другой сущностью или сущностями. Она представляется на ER-диаграмме в виде прямоугольника с закругленными углами. Первичный ключ зависимой сущности включает первичные ключи одной или более родительских сущностей.

Связи в IDEFIX представляют собой ссылки; соединения и ассоциации между сущностями. Связи это глаголы, которые показывают, как соотносятся сущности между собой.

После экспорта в ERwin созданной нами модели в BPwin необходимо создать новый проект в ERwin. Кроме того определяем сущности и атрибуты. В данной модели сущностями являются: «Кафедра», «Студент», «Дисциплины» «Тип занятий», «Преподаватель» и «Группы».

Логическая модель комплекса задач, рассматриваемого в данном дипломном проекте, представлена на рис 9.

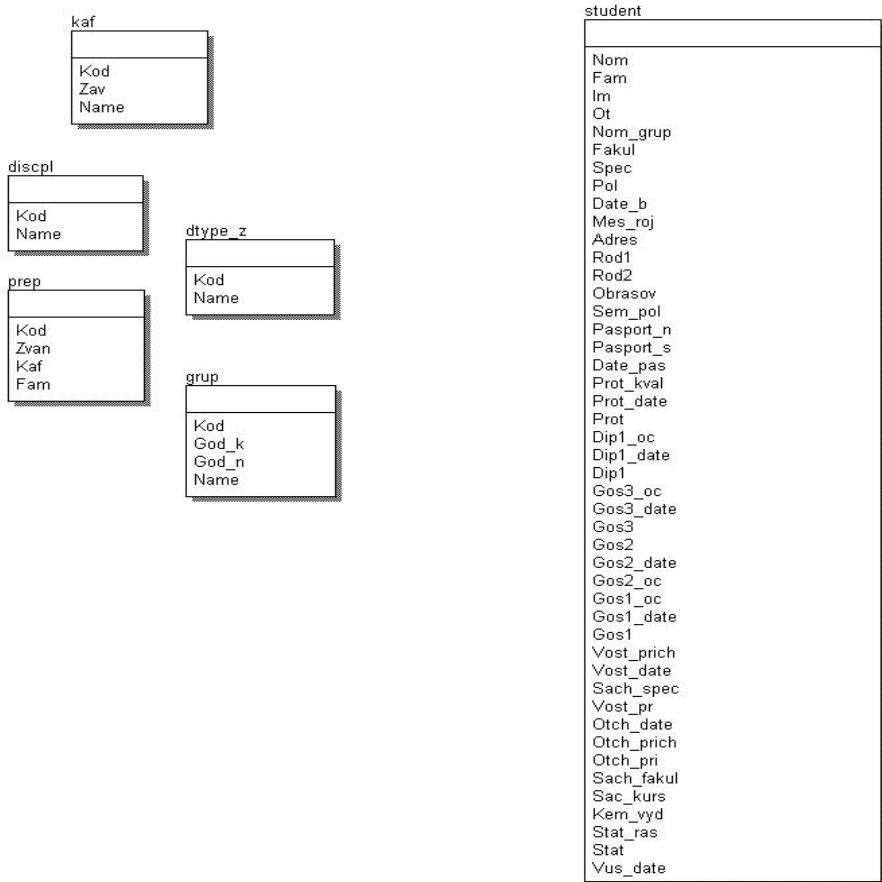


Рис. 9. Модель «сущность-связь»

Модель данных, основанная на ключах. Каждая сущность содержит горизонтальную линию, разделяющую атрибут на две группы.

Атрибуты, расположенные над линией, называются первичным ключом. Первичный ключ предназначен для уникальной идентификации экземпляра сущности.

При создании сущности необходимо выделить группу атрибутов, которые потенциально могут стать первичным ключом (потенциальные ключи), затем произвести отбор атрибутов для включения в состав первичного ключа, следуя следующим рекомендациям:

Первичный ключ должен быть подобран таким образом, чтобы по значениям атрибутов, в него включенных, можно было точно идентифицировать экземпляр сущности.

Никакой из атрибутов первичного ключа не должен иметь нулевое значение.

Значение атрибутов первичного ключа не должны меняться. Если значение изменилось, значит, это уже другой экземпляр сущности.

При выборе первичного ключа можно внести в сущность дополнительный атрибут и сделать его ключом. Так, для определения первичного ключа часто используют уникальные номера, которые могут автоматически генерироваться системой при добавлении экземпляра сущности в БД. Применение уникальных номеров облегчает процесс индексации и поиска в БД.

На рисунке 10 представлена модель, основанная на ключах, т.е. в каждой сущности мы выделили ключевые атрибуты, которые однозначно идентифицируют указанную сущность. Для сущности «Студент» ключевым атрибутом является «Номер зачетной книжки», для сущности «Преподаватель» - «Код удостоверения», для сущности «Дисциплины» - «Код дисциплины» и т.д.

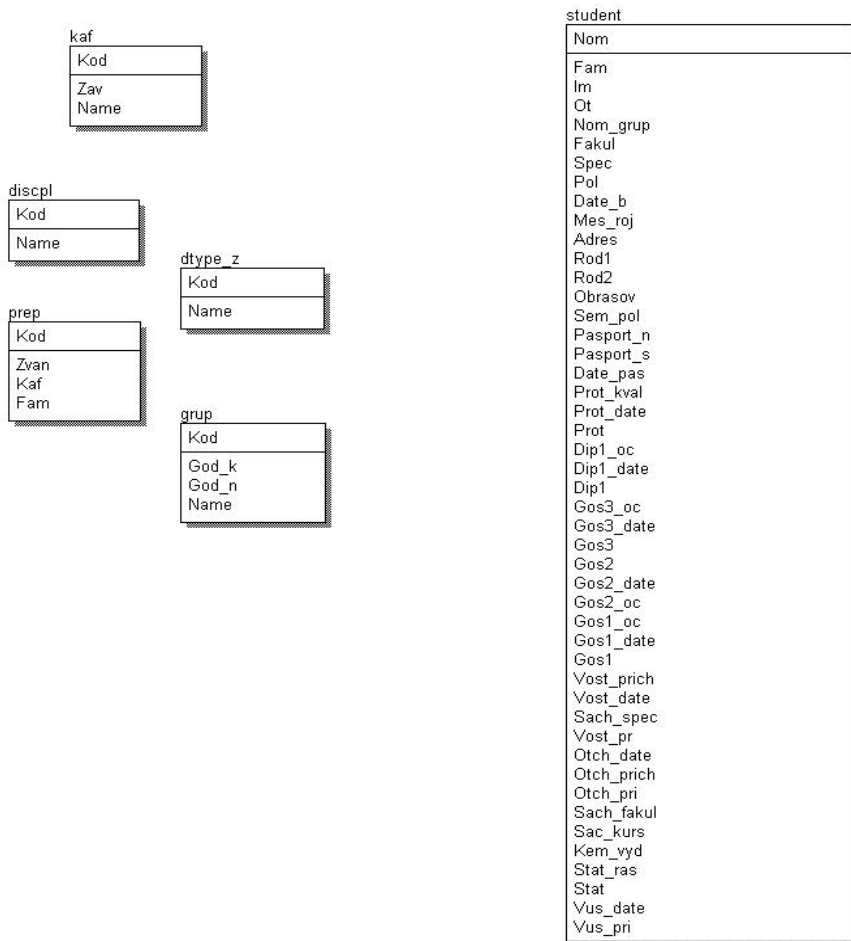


Рис.10. Модель, основанная на ключах

Для приведения таблиц БД к первой нормальной форме нужно выполнить условие, при которой все атрибуты содержат атомарные значения. Проверим соответствие БД второй нормальной форме. Все неключевые атрибуты должны зависеть от первичного ключа. Это условие выполняется для всех сущностей БД. Для приведения БД к третьей нормальной форме необходимо обеспечить отсутствие транзитивных зависимостей неключевых атрибутов, условие соблюдается.

Полученная полная атрибутивная модель в третьей нормальной форме изображена на рис. 11.

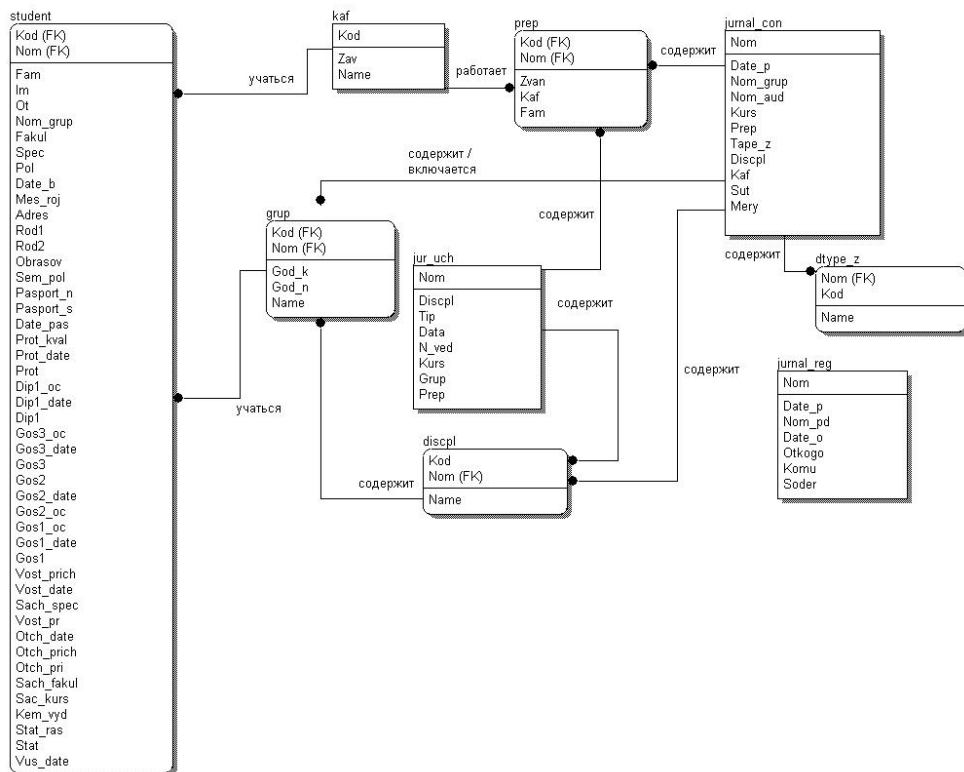


Рис.11 Полная атрибутивная модель

Физическая модель

Физическая модель содержит информацию, необходимую системным разработчикам для понимания механизма реализации логической модели и СУБД.

В ERwin также представлены два уровня физической модели: трансформационная модель и модель СУБД.

Целью трансформационной модели является предоставление информации администратору БД для создания эффективной структуры хранения, включающей в себя записи, формирующие БД. Трансформационная модель помогает разработчикам выбрать структуру хранения данных и реализовать систему доступа к ним.

Логическую модель на (рис.11) преобразуем в физическую модель. На рисунке 12 показана физическая модель, которую мы получили, путем подробной детализации имеющихся и внесением новых сущностей.

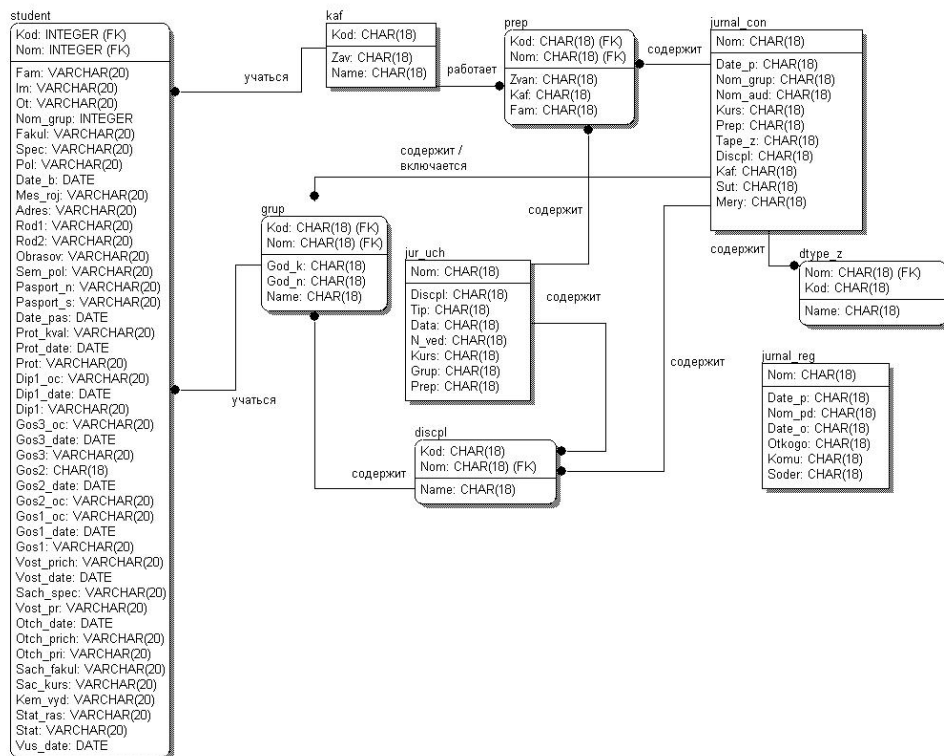


Рис.12. Трансформационная модель (логический уровень)

Перед началом проектирования БД необходимо убедиться в обеспечении следующих требований:

- физическая модель данных должна соответствовать требованиям, предъявляемым к проектируемой системе;
- выбор определённой физической модели должен быть аргументирован;
- должны быть определены возможность наращивания существующей структуры хранения, а также выявлены её ограничения.

Модель СУБД напрямую транслируется из трансформационной модели, являясь отображением системного каталога. ERwin напрямую поддерживает эту модель через

функцию генерации схемы БД. При составлении схемы БД в качестве индексов могут использоваться как ключевые атрибуты, так и остальные поля БД.

Для непосредственного создания физической модели данных необходимо в окне создания новой модели выбрать тип создаваемой модели Logical/Physical и тип базы данных, в которой необходимо создать таблицы (Рис.13). В данном курсовом проекте мы создадим физическую модель для FoxPro.

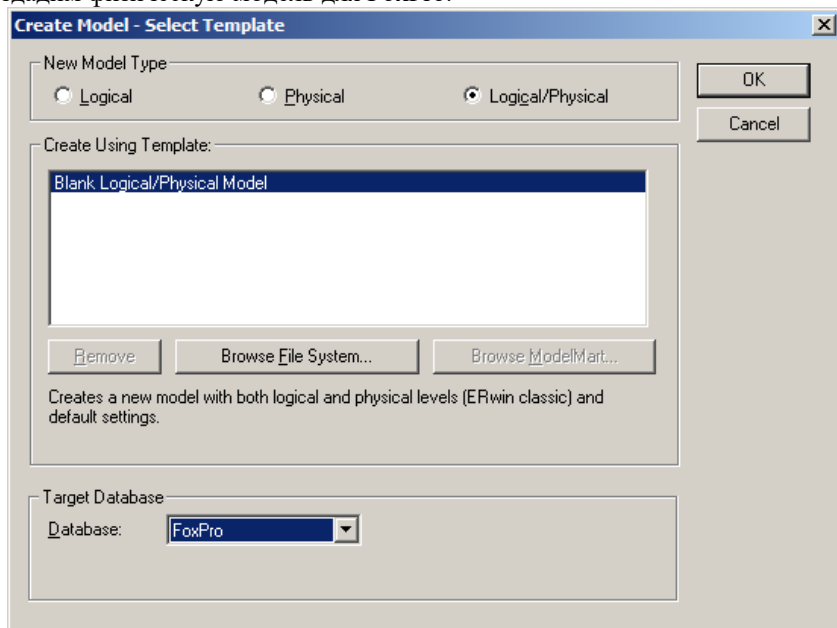


Рис.13 Выбор типа модели

Далее создается модель, в которую необходимо импортировать данные из BPwin. В случае если нам необходимо создать физическую модель из уже ранее существующей модели созданной в Erwin, то сначала надо переключиться на физический уровень данных (Рис.14).

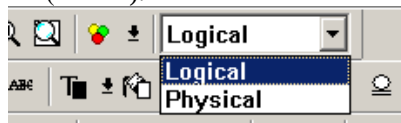


Рис. 14 Выбор уровня данных

Затем, в появившейся панели инструментов нажать кнопку  и в появившемся окне выбрать Model level compare и указать файл из которого необходимо импортировать данные (Рис. 15)

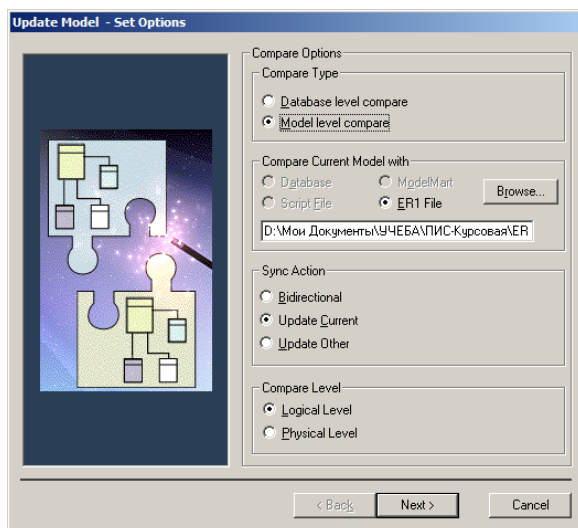


Рис. 15 Импорт данных из существующей модели

В последующих окнах мы должны выбрать данные, настройки которые необходимо импортировать. Импорт таблиц и их строк осуществляется путем нажатия на кнопку Import . Чтобы не импортировать данные, которые нам не нужны, надо выбрать их из списка и нажать на кнопку Ignore . Окно выбора данных представлено на рис.16.

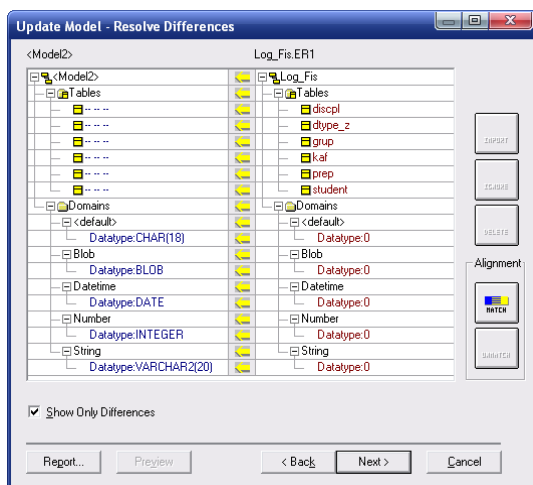


Рис 16. Окно выбора импортируемых данных

Если в процессе создания нам нужно поменять тип БД, то необходимо щелкнуть по кнопке смены сервера БД  и в появившемся окне поменять сервер БД (Рис. 17)

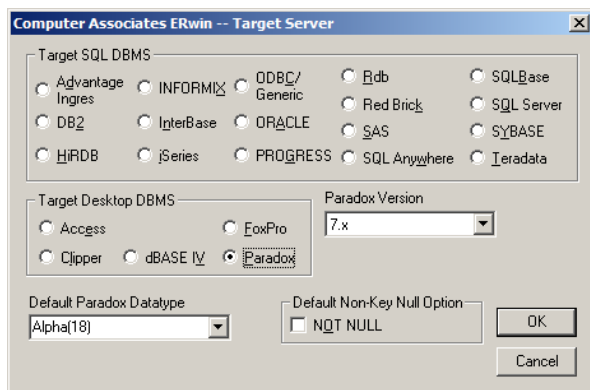


Рис 17. Окно смены сервера БД

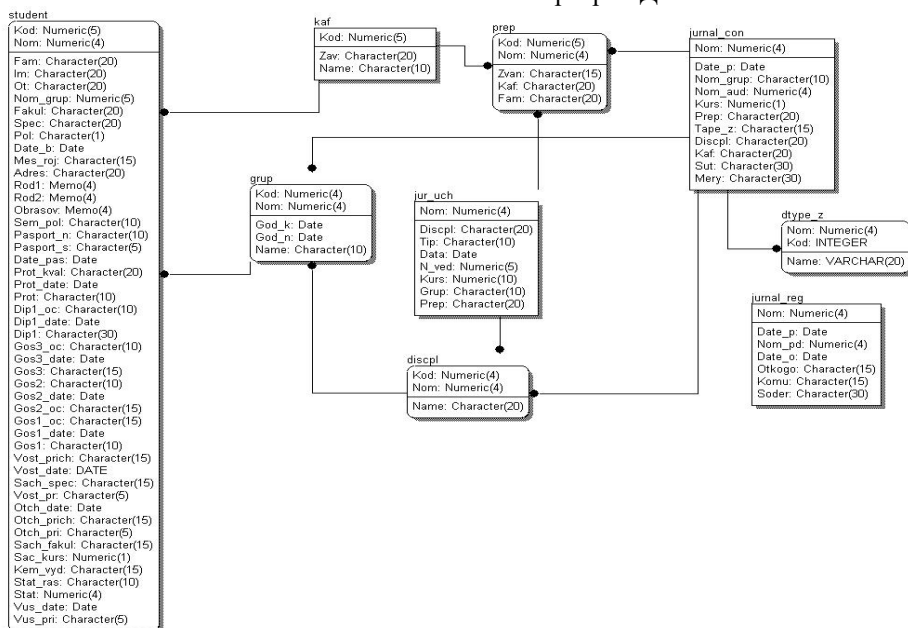


Рис 18. Трансформационная модель (физический уровень)

Генерация физической модели

После импорта данных, для экспорта физической модели данных, надо щелкнуть по кнопке генерации БД  и в появившемся окне настроить схему создания модели (Рис.19)

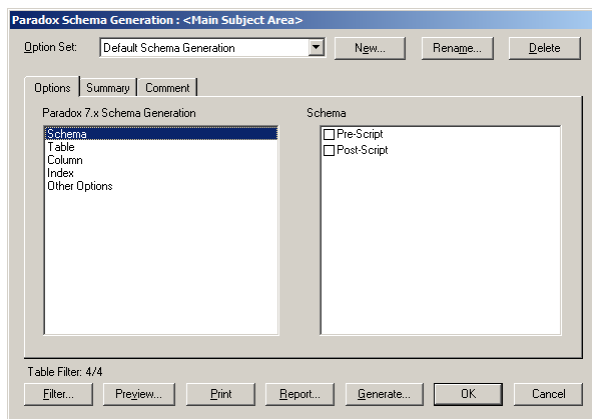


Рис. 19 Окно генерации БД и настройки схемы генерации

После настройки нужно нажать кнопку Generate и указать путь, куда будут созданы наши таблицы (Рис. 20)

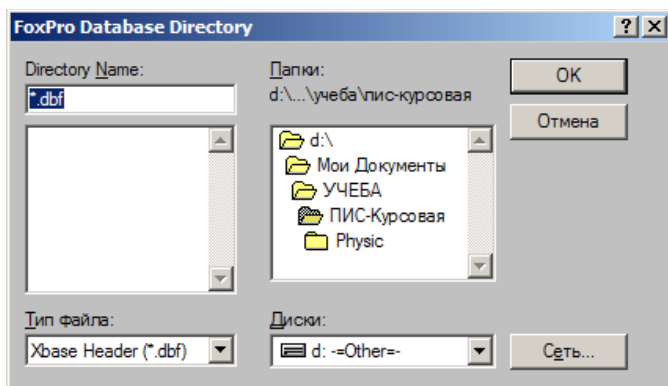


Рис. 20 Окно выбора пути сохранения таблиц

7. Литература:

1. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. – М.: Финансы и статистика, 2000.
2. Вендров А.М. . CASE-технологии. Современные методы и средства проектирования информационных систем. – М.: Финансы и статистика, 1998.
3. Грекул В.И., Денищенко Г.Н., Коровкина Н.Л. Проектирование информационных систем: учебное пособие / В.И. Грекул, Г.Н. Денищенко, Н.Л. Коровкина. – 2-е изд., испр. М.: Интернет-Университет Информационных Технологий; БИНОМ.Лаборатория знаний, 2008. – 300 с.
4. Дубейковский В.И. Практика функционального моделирования с AllFusion Process Modeler 4.1. Где? Зачем? Как? М.: ДИАЛОГ-МИФИ, 2004. – 464 с.
5. Смирнова Г.Н. и др. Проектирование экономических информационных систем: Учебник / Смирнова Г.Н., Сорокин А.А., Тельнов Ю.Ф. – М.: Финансы и статистика, 2001.
6. Базы данных: Учебник для высших учебных заведений/ Под ред. проф. А.Д.Хомоненко. – СПб.: КОРОНА принт, 2000. – 416 с.
7. Маклаков С.В. Моделирование бизнес процессов с AllFusion Process Modeler (BPWin 4.1). М.: ДИАЛОГ – МИФИ, 2004. – 240с.
8. Маклаков С.В. BPWin, ERWin. CASE-средства разработки информационных систем. – М. ДИАЛОГ-МИФИ, 1999.
9. Моделирование и анализ IDEF-технологии: практикум/ С.В.Черемных, И.О.Семенов, В.С.Ручкин. – М. Финансы и статистика, 2002. – 192 с.: ил.
10. Атре Ш. Структурный подход к организации баз данных. – М.: Финансы и статистика, 1983. – 320 с.
11. Бойко В.В., Савинков В.М. Проектирование баз данных информационных систем. – М.: Финансы и статистика, 1989. – 351 с.

8. Приложения

Приложение №1

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГБОУ ВО
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»**

**Факультет Информационных систем финансов и аудита
Кафедра ИТиПИВЭ**

КУРСОВОЙ ПРОЕКТ
по дисциплине:
«Проектирование информационных систем»

На тему: « _____ »
название темы

Выполнил (а): студент (ка) _____
номер группы, курс

факультет

ФИО студента

Руководитель проекта: _____
уч. степень, звание (должность)

ФИО преподавателя

Махачкала-20__ г.

Постановка задачи

Вариант № _____

Необходимо разработать курсовой проект на тему: « _____ »

- 1.** Произвести сбор необходимой литературы и анализ входной информации.
- 2.** Определить состав главных и вспомогательных работ, происходящих процессов, информационных потоков, хранилищ данных и т.д.
- 3.** Разработать эскизный проект будущей ИС.
- 4.** Определить взаимосвязь модулей и потоков информации.
- 5.** Реализовать эскизный проект с помощью инструментов моделирования BPwin 4.1 и ERwin 4.1.
- 6.** Проверить согласованность всех используемых объектов системы и соблюдение наследования потоков, наличие описания комментариев к каждому объекту.
- 7.** Создать отчет согласованности с выбранной методологией (для каждой нотации).
- 8.** Провести генерацию отчетов по каждому пакету моделирования (BPwin 4.1 и ERwin 4.1) в форматах HTML и RTF согласно требованиям к курсовому проектированию: вариант в формате RTF включается в приложение к отчету по курсовому проектированию; вариант в формате HTML сдается на электронном носителе руководителю проекта.
- 9.** Создать приложение в СУБД-ориентированной среде Visual FoxPro 9.0, Borland C++ Builder 6.0 и др.
- 10.** Оформить пояснительную записку (в печатном и электронном виде) по разработанному курсовому проекту и представить ее руководителю проекта.

Мурадов Марат Миязуллахович
Адеева Мариам Гасанагаевна

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к выполнению курсового проекта
по дисциплине:
«Проектирование информационных систем»**

для студентов направления подготовки бакалавров
09.03.03 – «Прикладная информатика»