

**Министерство образования Украины**  
**Запорожский государственный университет**

Факультет математики и  
экономической  
кибернетики  
Кафедра ММ и ИТ

# **ДИПЛОМНАЯ РАБОТА**

**на тему: Разработка концептуальной модели клиентской  
части информационной интегрированной системы  
“Студент”**

Выполнил Шапар Владимир Владимирович

ст. группы 8222-1

/шифр/      /подпись и дата/      /ф., и., о./  
Руководитель доцент кафедры ММиИТ Ермолаев В. А.

/должность/      /подпись и дата/      /ф., и., о./  
Нормоконтролер Бакурова А. В.  
/подпись/      /ф., и., о./

**Запорожье**  
**1997**

## РЕФЕРАТ

Дипломная работа: 39 страниц, 3 главы, 6 рисунков, 1 таблицу, 11 терминов, 17 источников

Объектом исследования является методы реляционной алгебры для разработки концептуальных моделей.

Цель данной дипломной работы состоит в выборе оптимальной теоретической и методологической базы концептуального моделирования, разработки концептуальной модели данных клиентской части информационной интегрированной системы “Студент”, выбор системы управления базами данных наиболее подходящей для физической реализации информационной системы.

В результате дипломной работы была создана концептуальная модель клиентской части интегрированной информационной системы “Студент”, позволяющая построить на ее основе приложение обслуживающее работу ВУЗа. Данная концептуальная модель более полно описывает предметную область чем рассмотренные аналоги.

КОНЦЕПТУАЛЬНАЯ МОДЕЛЬ, ПРЕДМЕТНАЯ ОБЛАСТЬ, РЕЛЯЦИОННАЯ МОДЕЛЬ, СХЕМА ОТНОШЕНИЙ, КЛЮЧ, ФУНКЦИОНАЛЬНАЯ ЗАВИСИМОСТЬ, НОРМАЛЬНАЯ ФОРМА, НОРМАЛИЗАЦИЯ, КОРТЕЖ, ДОМЕН, АТТРИБУТ

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	6
1 ОБЗОР СОВРЕМЕННЫХ ПОДХОДОВ К СОЗДАНИЮ ИНФОРМАЦИОННЫХ МОДЕЛЕЙ.....	8
1.1 ОБЗОР И АНАЛИЗ АНАЛОГОВ РАЗРАБАТЫВАЕМОЙ МОДЕЛИ.....	11
1.2 ОПИСАНИЕ БАЗОВЫХ СРЕДСТВ ДЛЯ ИНФОРМАЦИОННОГО МОДЕЛИРОВАНИЯ.....	14
1.3 ОПРЕДЕЛЕНИЕ ОГРАНИЧЕНИЙ НА РАЗРАБАТЫВАЕМУЮ МОДЕЛЬ.....	17
1.4 ВЫВОДЫ.....	18
2 РАЗРАБОТКА И НОРМАЛИЗАЦИЯ КОНЦЕПТУАЛЬНОЙ МОДЕЛИ.....	19
2.1 РАЗРАБОТКА ПРОТОТИПА КОНЦЕПТУАЛЬНОЙ МОДЕЛИ.....	19
2.1.1 Анализ предметной области.....	20
2.1.2 Определение элементов прототипа.....	21
2.2 НОРМАЛИЗАЦИЯ МОДЕЛИ.....	23
2.2.1 Построение схемы отношения.....	24
2.2.2 Нормализация посредством синтеза.....	26
2.3 ВЫВОДЫ.....	31
3 ВЫБОР СУБД ДЛЯ ФИЗИЧЕСКОЙ РЕАЛИЗАЦИИ КОНЦЕПТУАЛЬНОЙ МОДЕЛИ.....	34
3.1 ОБЗОР И АНАЛИЗ СОВРЕМЕННЫХ СУБД.....	36
3.2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ К СУБД ДЛЯ ФИЗИЧЕСКОЙ РЕАЛИЗАЦИИ КОНЦЕПТУАЛЬНОЙ МОДЕЛИ.....	39
3.3 ВЫВОДЫ.....	<b>ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.</b>
ЗАКЛЮЧЕНИЕ.....	43
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ.....	44



## ВВЕДЕНИЕ

В последнее время в различных организациях широко внедряются компьютерные системы и различные автоматизированные системы управления, используемые для хранения, поиска и обработки различного рода информации. Разработка таких систем в настоящее время очень актуальны.

В нашем университете также уделяется большое внимание внедрению компьютерной обработки данных в различные сферы деятельности ВУЗа. Компьютер в ЗГУ используется для выполнения научных исследований на большинстве факультетов: Математическом, Физическом, Биологическом, Историческом, Юридическом, Филологическом, факультете здоровья и спорта и других. Компьютер используется для автоматизации работ бухгалтерии, учебного отдела, для выполнения издательских работ, для учета данных о поступающих в ВУЗ абитуриентах [1]. В частности уже несколько лет в университете работает система “Абитуриент” [2] разработанная в нашем ВУЗе.

Очевидно, что следующим этапом возникает необходимость создания информационной системы для автоматизации работы деканатов и отдела кадров, с возможностью полного учета и анализа различных показателей в учебном процессе, научной и другой деятельности студентов. Известно множество приложений автоматизирующие описанные выше области однако большинство из них представляют собой обособленные программы. В задачу данной дипломной работы входит создание модели не отдельных компонент а общую модель описывающую все в целом. Поскольку данная задача довольно сложна, будет рассмотрена часть модели содержащая информацию о студенте.

Достижение цели данной работы предполагает решение следующих задач:

- исследование современных подходов к реализации концептуальных моделей баз данных (провести обзор теоретических подходов, и выбрать наиболее подходящий для реализации)
- исследование аналогов разрабатываемой модели (проанализировать аналоги и выявить их плохие и хорошие стороны)
- разработка и нормализация концептуальной модели (проанализировать предметную область задачи, составить набор отношений и связи между ними, используя метод синтеза провести нормализацию отношений)
- исследование современных систем управления базами данных
- выбор оптимальной системы управления базами данных (СУБД) для реализации разработанной концептуальной модели (определить ряд требований к СУБД, и на их основе выбрать оптимальную СУБД для физической реализации модели)

На основе построенной концептуальной модели данных и используя выбранную СУБД может быть легко разработано программное обеспечение, позволяющее автоматизировать процесс ведения и учета личных дел студентов, работу деканатов и отдела кадров, генерации разнообразных отчетов по существующей информации о студентах.

Данная разработка может быть использована учебными заведениями различных уровней аккредитации.

## **1 ОБЗОР СОВРЕМЕННЫХ ПОДХОДОВ К СОЗДАНИЮ ИНФОРМАЦИОННЫХ МОДЕЛЕЙ**

Моделирование данных и реализация баз данных формируют существенную отрасль научных и промышленных разработок. Базы данных как источники и носители информации применяются в разнообразных областях, широко распространены и играют немаловажную роль в достижении прогресса во всех отраслях науки и техники, а также человеческой и социальной деятельности. Поэтому, исследования направленные на разработку формальной основы, методов и инструментальных средств для надежного проектирования и реализации устойчивых информационных систем очень актуальны.

В настоящее время в различных областях применения БД (банковское дело, автоматизация бухгалтерского учета, автоматизация офиса, робототехника) требуются средства моделирования сложных объектов. Следовательно, любые разрабатываемые или существующие модели (реляционные, функциональные) расширяются для лучшего соответствия этим требованиям.

Один из основополагающих подходов к созданию баз данных - реляционный был разработан в 70 гг. Самый общий термин - реляционный подход - обозначает определенную идеологию создания баз данных и баз знаний, идеологию которая была объявлена основной в японском проекте ЭВМ пятого поколения (1980 г.) [3]. Создатель реляционной модели данных Э.Ф. Кодд в 1981 г. получил за ее разработку [4] премию Тьюринга Американской ассоциации по вычислительной технике.

Одним из основных преимуществ реляционной модели является ее однородность. Все данные рассматриваются как хранимые в таблицах, в

которых каждая строка имеет один и тот же формат. Каждая строка в таблице представляет собой некий объект реального мира или отношение между объектами.

Большое количество моделей которые разрабатывались позднее базировались на расширении или уточнении классической реляционной модели.

A. Rosenthal и D. Reiner [5] представили проект Проектирование Баз Данных и Инструментальные средства проверки - система которая использует информацию содержащуюся внутри себя для предотвращения трансформации модели.

P. Fraternali, L. Tanca [6] добавляют некоторое дополнение к системам Базы Данных - Активный подход Баз Данных. Это основано на правилах ЕСА и некоторых расширениях ЕСА. Подход к системе Баз Данных определен согласно некоторому набору правил ЕСА. Другие авторы работают в области Исследований Наборов Активных Правил Базы Данных. Пример - Система Правил Starbust см. A. Aiken, J. M. Hellerstein, J. Widom [7].

Разнообразие подходов нацелено на обработку традиционных концептуальных методов моделирования и соответствующей теории. Рассмотрим следующие примеры.

Некоторые решение проблемы организации управления связями давались E. F. Codd в [3] и R. L. Haskin, R. A. Lorie в [8]. Добавление теоретических средств для концептуального моделирования и преодоления ограничений 1НФ (первой нормальной формы) либо анализа альтернативных представлений (ЧНФ (частичной), не -1НФ) обсуждены например M. A. Roth, H. F. Korth, A. Silberschatz в [9].

Реляционная Модель Данных может быть также расширена при введении дополнительных измерений. Многочисленные авторы



предлагают добавлять временное измерение к Реляционной Базе Данных расширяя соответственно реляционную алгебру. Один из хороших примеров дается J. Clifford, A. Croker, A. Tuzhilin [10].

На подход ERC + поддерживающий PRC BD3 следует обратить особое внимание. Модель ERC + описанная например [11] C. Parent, H. Rolin, K. Yetongnon, S. Spaccapietra - базируется на модели ER, P. P. Chen [12] предложил расширить эту модель для описания и управления сложными объектами с параметрами имеющими древовидную структуру с возможными неоднозначными либо необязательными ветвлениями. Это - обобщение самых лучших особенностей Сетевой и Иерархической модели в пределах одного усиленного реляционного подхода. Из другой области модель ERC + берет преимущество 3-х уровневой архитектуры проектирования модели данных - внешней, концептуальной и внутренней. Алгебра ERC + вводит операторы для соответствия представления пользователя об объектах и базовой концептуальной схемой.

Еще одно направление разработанное настоящее время - Вероятностные Модели Данных. Этот подход добавляет некоторые возможности обработки неполных и неопределенных данных. Вероятностный подход к Расширенной Реляционной Модели и Алгебры предложен например D. Deu и S. Sarkar [13].

В качестве другого подхода к разработке приложений работающих с базами данных можно привести метод активных словарей данных разработанный в нашем университете [15,16].

Данный метод основан на расширении основных принципов теории реляционных моделей данных вычисляемыми атрибутами [17].

## 1.1 Обзор и анализ аналогов разрабатываемой модели

В настоящее время во многих учебных заведениях стали использоваться различные автоматизированные системы управления. Однако известные нам реализации в данной области в большинстве случаев велись без создания четкой информационной модели и без глубокого анализа моделируемой предметной области.

В качестве одного из модельных примеров разрабатываемой концептуальной модели можно рассмотреть концептуальную модель “Университет” подробно описанную в [14].

Схематично предметная область данной модели изображена на Рисунке 1.1



Рисунок 1.1

Однако так как данная схема является модельной то в ней не рассматриваются некоторые аспекты реальной модели. Например не

учитывается индивидуальная деятельность студентов, оплата студентом обучения, не полностью учитывается успеваемость и др.

В качестве реальной модели можно указать концептуальная модель разработанную в Украинском транспортном университете [13]. Автоматизированная система управления данного ВУЗа состоит из следующих подсистем:

- "Абитуриент";
- "Деканат";
- "Заочный деканат";
- "Электронный каталог";
- "Электронный каталог диссертаций";
- "Статистическая отчетность профессорско-преподавательского состава";
- "Электронная почта".

Схематично данная концептуальная модель изображена на Рисунке 1.2.

Однако данная модель имеет некоторые недостатки. Данная система построена на основе файлового сервера, а системы с файловым сервером имеют недостатки, не позволяющие им обслуживать многопользовательские приложения. Модель с файловым сервером не обеспечивает необходимой многопользовательским приложениям согласованности данных (одновременного доступа к одному набору данных множества пользователей).

## АСУ ВУЗ. Концептуальная схема

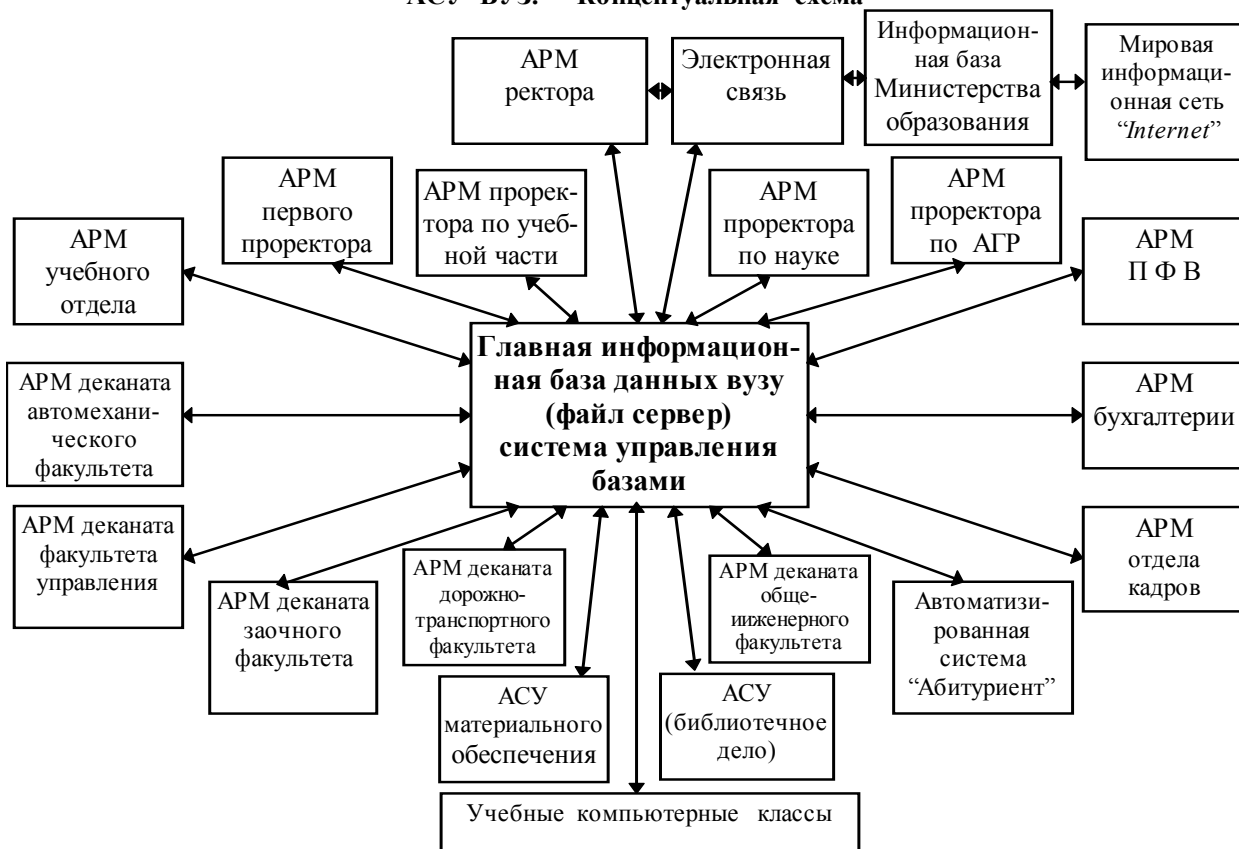


Рисунок 1.2

К тому же, если множество файлов запрашивают и передают по сети сразу много рабочих станций, то сеть быстро насыщается, и производительность системы ухудшается.

Другим недостатком данной модели является то, что, как видно из схемы, данная модель является фактически набором отдельных АСУ, и поэтому в данной модели будет присутствовать дублирование данных, а также возникать затруднения в синхронизации при обновлении данных.

1.2 Описание базовых средств для информационного моделирования.

Информационные модели, лежащие в основе известных разработок в области автоматизации деятельности ВУЗа в Украине, к сожалению, строились на базе эвристических подходов и не отличаются использованием известных формальных методов реализации таких моделей. Одной из задач данной работы является создание законченной формальной модели данных на базе теоретических принципов реляционного проектирования данных.

Для этого в качестве инструментального средства построения данной модели выбран ER подход к проектированию баз данных [1], так как теория реляционных баз данных является областью приложений математической логики и современной алгебры и опирается на точный математический формализм.

Приведем некоторые определения элементов базового теоретического аппарата используемых в качестве основы проектирования модели данных в дипломной работе.

### **Определение 1.**

Схемой отношений  $R$  называется конечное множество имен атрибутов  $\{A_1, A_2, \dots, A_n\}$ . Каждому имени атрибута  $A_i$  ставится в соответствие множество  $D_i$  называемое доменом атрибута  $A_i$ ,  $1 \leq i \leq n$ . Домен атрибута  $A_i$  будем обозначать  $dom(A_i)$ . Домен является произвольным непустым конечным или счетным множеством. Пусть  $D = D_1 \cup D_2 \cup \dots \cup D_n$ . Отношение  $r$  со схемой  $R$  - это конечное множество отображений  $\{t_1, t_2, \dots, t_p\}$  из  $R$  в  $D$ , причем каждое отображение  $t \in r$  должно удовлетворять следующему ограничению  $t(A_i) \in D_i$ ,  $1 \leq i \leq n$ . Эти отношения называют кортежами.

### Определение 2.

Ключ отношения  $r$  со схемой  $R$  является подмножеством  $K = \{B_1, B_2, \dots, B_m\} \subseteq R$  со следующими свойством: для любых двух кортежей  $t_1$  и  $t_2$  в  $r$  существует такое  $B \in K$  что  $t_1(B) \neq t_2(B)$ , другими словами не существует двух кортежей имеющих одно и то же значение на всех атрибутах из  $K$ . Это условие записывается так:  $t_1(K) \neq t_2(K)$ . Таким образом достаточно знать  $K$  значение кортежа чтобы однозначно идентифицировать кортеж. Более точно ключ отношения  $r(R)$  является подмножеством  $K \subseteq R$ , таким что для любых различных кортежей  $t_1$  и  $t_2$  из  $r$  выполняется  $t_1(K) \neq t_2(K)$ , и ни одно собственное подмножество  $K' \subset K$  не обладает этим свойством. Множество  $K$  является суперключом относительно  $r$ , если  $K$  содержит ключ отношения  $r$ .

### Определение 3.

Пусть  $r$  - отношение со схемой  $R$ ,  $X$  и  $Y$  - подмножества  $R$ . Отношение  $r$  удовлетворяет функциональной зависимости  $X \rightarrow Y$ , если  $\pi_Y(\sigma_{X=x}(r))$  имеет не более чем один кортеж для каждого  $X$  значения  $x$ . Где  $\sigma_{X=x}(r)$  обозначает операцию выбора (выбрать в  $r$  кортеж в котором значение  $X$  равно  $x$ ), а  $\pi_X(\sigma(r))$  - проекция  $r$  на  $X$ , то есть отношение  $r'(X)$ , полученное вычеркиванием соответствующих атрибутам в  $R - X$  и исключением из оставшихся столбцов повторяющихся строк.

### Определение 4.

Схема отношения  $R$  находится в первой нормальной форме (1НФ) если значения в  $\text{dom}(A)$  являются атомарными для каждого атрибута  $A$  в  $R$ , т.е. значение в домене не являются ни списком ни сложным значением. Преимущество 1НФ в том, что 1НФ позволяет выразить  $F$ -зависимости с той степенью детализации с какой нам требуется, что невозможно без 1НФ.

**Определение 5.**

Для данной схемы отношения  $R$ , подмножества  $X$  множества  $R$ , атрибута  $A$  в  $R$  и множества функциональных зависимостей  $F$  атрибут  $A$  называется транзитивно зависимым от  $X$  в  $R$ , если существует подмножество  $Y \subseteq R$ , такое что  $X \rightarrow Y, Y \rightarrow X, Y \rightarrow A$  относительно  $F$  и  $A \notin XY$

**Определение 6.**

Схема отношений  $R$  находится в 3НФ относительно множества функциональных зависимостей  $F$ , если она находится в 1НФ и ни один из первичных атрибутов в  $R$  не является транзитивно зависимым от ключа для  $R$ .

**Определение 7.**

Пусть  $G$  - множество CF - зависимостей над  $R$  и  $F$  - множество F - зависимостей над  $R$ .  $G$  эквивалентно  $F$ , если каждое отношение  $r(R)$ , удовлетворяющее  $G$ , удовлетворяет  $F$  и обратно.

**Определение 8.**

Множество CF - зависимостей  $F$  называется кольцевым, если не существует различных левых множеств  $X$  и  $Z$ , таких, что  $X \leftrightarrow Z$  в  $F$ .

**Определение 9.**

Пусть  $G$  - кольцевое множество. CF - зависимость  $(X_1, X_2, \dots, X_r) \rightarrow Y$  из  $G$  называется редуцированной если левые множества не содержат перемещаемых атрибутов, а правые - посторонний. Множество  $G$  редуцировано, если редуцирована каждая CF зависимость в  $G$ .

**Определение 10.**

Множество  $G$  минимально, если содержит столько же левых множеств, сколько любое другое эквивалентное множество.

### 1.3 Определение ограничений на разрабатываемую модель.

На основе рассмотренных аналогов и при помощи выбранной методологической базы можно вывести ряд требований накладываемых на разрабатываемую концептуальную модель:

1. Данная концептуальная модель должна содержать информацию о студентах взятую из отдела кадров, деканатов, учебного отдела.
2. Построить интегрированную модель для описанных выше подразделений вуза.
3. Исключить в модели семантическое дублирование данных.

Например информация о студенте (номер курса, факультет, специальность и др.) храниться отдельно в деканате и отделе кадров, необходимо построить модель которая исключала бы данное дублирование.

Для достижения описанных выше целей необходимо проанализировать предметную область задачи, построить схему отношений для данной предметной области, описать взаимосвязь между данными отношениями, используя метод синтеза произвести нормализацию данных отношений.

Таким образом, формально задачу разработки отмеченной модели данных можно сформулировать следующим образом:

требуется создать реляционную схему  $R = \{R_1, \dots, R_p\}$ , удовлетворяющую следующим требованиям:

1. Каждая схема отношения  $R_i$  из  $R$  находится в 3НФ относительно  $F$  (множества функциональных зависимостей  $R$ ).
2. Не существует схемы отношений с меньшим числом схем чем  $R$ , удовлетворяющей требованию 1.



3. Для любого отношения  $r(R)$ , удовлетворяющего  $F$ ,
- $$r = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie \dots \bowtie \pi_{R_p}(r)$$

#### 1.4 Выводы

1. Проведен обзор современных подходов к созданию информационных моделей и выявлена тенденция к построению интегрированных моделей.
2. На основе обзора в качестве базового средства моделирования выбран ER подход к проектированию моделей данных.
3. Проведен анализ аналогов разрабатываемой модели и определены требования накладываемые на модель.
4. Сформулирована формальная задача разработки модели.

## **2 РАЗРАБОТКА И НОРМАЛИЗАЦИЯ КОНЦЕПТУАЛЬНОЙ МОДЕЛИ**

Один из эффективных методов разработки концептуальной модели предметной области - введение ряда понятий и концепций реляционной модели данных. Эти понятия и концепции применяются в процессе анализа информации о данных и их структуре, полученной у конечных пользователей. Концептуальная модель служит основой для создания физической модели, которая может быть реализована средствами системы управления базами данных.

Немаловажным этапом разработки концептуальной модели является анализ предметной области задачи, т.е. анализ информационной базы предприятия или организации для которой разрабатывается данная модель.

### **2.1 Разработка прототипа концептуальной модели**

Как уже упоминалось нашей задачей является построение модели базы данных, которая бы автоматизировала работу отдела кадров, деканатов, учебной части и социально-бытового отдела ВУЗа. В этой БД должна находиться информация обо всех студентах, их успеваемости, индивидуальной работе, их личные дела, медицинская информация и др. Также должна быть возможность создания всех необходимых отчетов, списков, ведомостей и др. документов необходимых для работы вышеперечисленных подразделений ВУЗа.

### 2.1.1 Анализ предметной области

Рассмотрим основные объекты модели данных, их атрибуты и связи.

При анализе предметной области можно выделить несколько основных сущностей модели данных: “Личные дела студентов”, “Успеваемость”, “Социально бытовая информация”, “Бухгалтерская информация”, “Система справочников”.

Рассмотрим взаимодействие этих объектов на самом высоком уровне абстракции. (см. Рисунок 2.3 )

Личное дело студента содержит общую информацию о студенте а также ссылки на различные справочники, успеваемость, социально бытовую информацию и др. Каждому личному делу студента соответствуют по одному объекту Успеваемости, Социально бытовой информации, Бухгалтерской информации. Каждому справочнику соответствует множество Личных дел студентов.

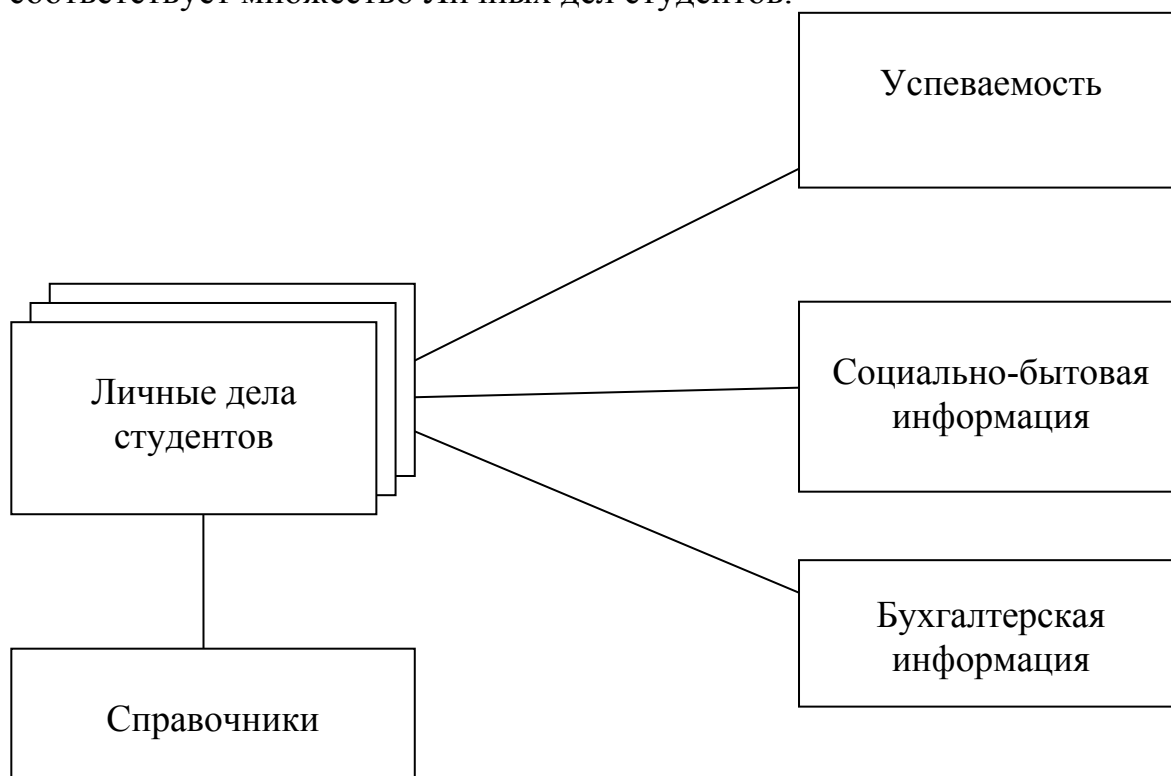


Рисунок 2.3

### 2.1.2 Определение элементов прототипа

Опишем теперь каждый из объектов подробнее.

Личное дело студента.

Как уже было сказано выше, Личное дело студента содержит информацию обо всех обучающихся студентах. Обычно данная информация находится в отделе кадров. Таким образом к определяющим свойствам этой сущности следует отнести атрибуты “Специальность, Место обучения, Форма обучения, Факультет, Курс, Номер группы, Год поступления, Номер зачетной книжки, Пол, Ф.И.О, Гражданство, Дата рождения, Дом. Адрес, Специальность, Вид обучения, Национальность, Семейное положение, Характер жительства, Место окончания уч. Зав., Год окончания уч. зав., Оконченное уч. Зав., Служба в армии, Трудовой стаж, Специальность до вуза, Социальное положение”

Успеваемость.

Объект “Успеваемость” содержит информацию обычно содержащуюся в деканатах, т.е успеваемость, рейтинг, олимпиады и т.д. В данном объекте содержатся следующие атрибуты “Тип академического отпуска, Дата начала Академического Отпуска, Дата конца Академического отпуска, Тема индивидуальной работы, Руководитель, Название предмета, Дата олимпиады, Занятое место, Номер семестра обучения, Название предмета, Количество часов, Преподаватель, Оценка, Номер семестра рейтинга, Номер контрольной, Предмет рейтинга, Балл, Передача предмета, Дата экзамена, Тип особой отметки, Текст особой отметки”

Социально бытовая информация.

Данный объект содержит социально бытовую информацию о студенте: “Служба в армии, Трудовой стаж, Специальность до вуза, Социальное положение, Проживание в общежитии, Информация о профсоюзе, Год поступления в профсоюз, Дата проживания в профилактории ”

Бухгалтерская информация.

В данном объекте находится информация об оплате студентом обучения:

“Номер семестра Оплаты, Название предприятия, Необходимая сумма, оплаченная сумма, Вид обучения”

Справочники

В справочниках находится различная расширенная информация по атрибутам, и фактически справочники служат для избавления модели от информационной избыточности данных.

## 2.2 Нормализация модели

Основное понятие, вводимое при разработке концептуальной модели, - это *нормализация отношений*. В процессе нормализации элементы данных группируются в таблицы, представляющие объекты и их взаимосвязи. Теория нормализации основана на том, что определенный набор отношений обладает лучшими свойствами при включении, модификации и удалении данных, чем все остальных наборы отношений, с помощью которых могут быть представлены данные.

Введение нормализации отношений при разработке концептуальной модели обеспечивает ее работоспособность. Это вовсе не означает что ненормализованная концептуальная модель обязательно окажется неработоспособной. Дело в том, что ненормализованная модель может вызвать определенные трудности реализации прикладных программ, модифицирующих базу данных.

### 2.2.1 Построение схемы отношения

Объединим описанные выше объекты в одну схему отношений:

R = Специальность, Место обучения, Форма обучения, Факультет, Курс, Номер группы, Год поступления, Номер зачетной книжки, Пол, Ф.И.О, Гражданство, Дата рождения, Дом. Адрес, Специальность, Тип академического отпуска, Дата начала Академического Отпуска, Дата конца Академического отпуска, Номер приказа о зачислении, Дата приказа о зачислении, Номер семестра Оплаты, Название предприятия, Необходимая сумма, оплаченная сумма, Вид обучения, Национальность, Семейное положение, Характер жительства, Место окончания уч. Зав., Год окончания уч. зав., Оконченное уч. Зав., Служба в армии, Трудовой стаж, Специальность до вуза, Социальное положение, Тема работы, Руководитель, Дата прохождения медосмотра, Проживание в общежитии, Информация о профсоюзе, Год поступления в профсоюз, Дата проживания в профилактории, Название предмета, Дата олимпиады, Занятое место, Номер семестра обучения, Название предмета, Количество часов, Преподаватель, Оценка, Номер семестра рейтинга, Номер контрольной, Предмет рейтинга, Балл, Передача предмета, Дата экзамена, Тип особой отметки, Текст особой отметки

Опишем множество функциональных зависимостей для данного отношения F:

$F = \{ \text{Номер зачетной книжки} \rightarrow \text{Специальность} \quad \text{Место обучения}$   
 $\text{Форма обучения} \quad \text{Факультет} \quad \text{Курс} \quad \text{Номер группы} \quad \text{Год}$   
 $\text{поступления} \quad \text{Пол} \quad \text{Ф.И.О} \quad \text{Гражданство} \quad \text{Дата рождения} \quad \text{Дом.}$   
 $\text{Адрес} \quad \text{Специальность} \quad \text{Номер приказа о зачислении} \quad \text{Дата приказа о}$   
 $\text{зачислении} \quad \text{Вид обучения} \quad \text{Национальность} \quad \text{Семейное положение}$

Характер жительства Место окончания уч. зав. Год окончания уч. зав. Оконченное уч. зав. Служба в армии Трудовой стаж  
 Специальность до вуза Социальное положение Проживание в общежитии,

Номер зачетной книжки -> Номер семестра оплаты Название предприятия Необходимая сумма Оплаченная сумма,

Номер зачетной книжки -> Тип академического отпуска Дата начала Академического Отпуска Дата конца Академического отпуска,

Номер зачетной книжки -> Тема работы Руководитель,

Номер зачетной книжки -> Дата прохождения медосмотра ,

Номер зачетной книжки -> Дата проживания в профилактории,

Номер зачетной книжки -> Название предмета Дата олимпиады Занятое место,

Номер зачетной книжки -> Номер семестра обучения Название предмета Количество часов Преподаватель Оценка Передача предмета Дата экзамена,

Номер зачетной книжки -> Тип особой отметки Текст особой отметки,

Номер зачетной книжки -> Номер семестра рейтинга Номер контрольной Предмет рейтинга Балл }



### 2.2.2 Нормализация посредством синтеза

Основная задача сводится к нахождению схемы базы данных в 3НФ, представляющую схему отношения, не находящуюся в 3НФ.

На входе задана схема отношения  $R$  и множество  $F$  - зависимостей  $F$  над  $R$ . С их помощью требуется создать схему базы данных  $R = \{R_1, \dots, R_p\}$  над  $R$ , удовлетворяющую следующим четырем требованиям:

1. Множество  $F$  полностью характеризуется с помощью  $R$ , т.е.  $F \equiv \{K \rightarrow R_i \mid R_i \text{ входит в } R, K - \text{выделенный ключ } R_i\}$ .

2. Каждая схема отношения  $R_i$  из  $R$  находится в 3НФ относительно  $F$ .

3. Не существует схемы базы данных с меньшим числом схем чем  $R$ , удовлетворяющей свойствам 1 и 2.

4. Для любого отношения  $r(R)$ , удовлетворяющего  $F$ ,  $r = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie \dots \bowtie \pi_{R_p}(r)$

Схему базы данных  $R$ , удовлетворяющую первым трем требованиям, называют полной схемой базы данных для  $F$ .

Выполнение метода начинается с построения кольцевого покрытия, порождающего для  $F$  полную схему базы данных. Поскольку метод строит схему отношений непосредственно из  $F$  - зависимостей  $F$ , его называют методом синтеза.

Поскольку самой трудной частью метода являются нахождение редуцированного, минимального, кольцевого покрытия для входного множества  $F$  - зависимостей, данный метод будет иметь полиномиальную временную сложность.

Синтез, использующий кольцевые покрытия дается следующим методом

## Метод Синтеза

**Вход:** множество  $F$  - зависимостей  $F$  на  $U$

**Выход:** полная схема базы данных для  $F$

1. Найти для  $F$  редуцированное минимальное покрытие  $G$ .
2. Для каждой  $CF$  зависимости  $(X_1, X_2, \dots, X_R) \rightarrow Y$  из  $G$  построить схему отношений  $R = X_1 X_2 \dots X_R Y$  с выделенным ключом  $K = \{X_1, X_2, \dots, X_R\}$
3. Вернуться к множеству схем п. 2

Используя данный метод проведем нормализацию отношения  $R$ .

Воспользовавшись определением 9 находим редуцированное минимальное покрытие  $F$ :

$G = \{$  ( Специальность Место обучения Форма обучения Факультет  
Курс Номер группы Год поступления Пол Ф.И.О Гражданство  
Дата рождения Дом. Адрес Специальность Номер приказа о  
зачислении Дата приказа о зачислении Вид обучения  
Национальность Семейное положение Характер жительства Место  
окончания уч. зав. Год окончания уч. зав. Оконченное уч. зав.  
Служба в армии Трудовой стаж Специальность до вуза  
Социальное положение Проживание в общежитии)  $\rightarrow$  Номер зачетной  
книжки,

( Номер семестра оплаты Название предприятия Необходимая сумма  
Оплаченная сумма)  $\rightarrow$  Номер зачетной книжки,

(Тип академического отпуска Дата начала Академического Отпуска  
Дата конца Академического отпуска)  $\rightarrow$  Номер зачетной книжки,

( Тема работы Руководитель)  $\rightarrow$  Номер зачетной книжки,

(Дата прохождения медосмотра )  $\rightarrow$  Номер зачетной книжки,

(Дата проживания в профилактории)  $\rightarrow$  Номер зачетной книжки,

(Название предмета Дата олимпиады Занятое место) -> Номер зачетной книжки,

(Номер семестра обучения Название предмета Количество часов Преподаватель Оценка Пересдача предмета Дата экзамена) -> Номер зачетной книжки,

(Тип особой отметки Текст особой отметки) -> Номер зачетной книжки,

( Номер семестра рейтинга Номер контрольной Предмет рейтинга Балл ) -> Номер зачетной книжки }

Из определения 10 следует, что данное редуцированное покрытие минимально.

Построим схемы отношений используя полученные данные комплексные функциональные зависимости:

R1 = Специальность, Место обучения, Форма обучения, Факультет, Курс, Номер группы, Год поступления, Номер зачетной книжки, Пол, Ф.И.О, Гражданство, Дата рождения, Дом. Адрес, Специальность, Номер приказа о зачислении, Дата приказа о зачислении, Вид обучения, Национальность, Семейное положение, Характер жительства, Место окончания уч. Зав., Год окончания уч. зав., Оконченное уч. Зав., Служба в армии, Трудовой стаж, Специальность до вуза, Социальное положение, Проживание в общежитии, Информация о профсоюзе, Год поступления в профсоюз

R2 = Номер зачетной книжки, Тип академического отпуска, Дата начала Академического Отпуска, Дата конца Академического отпуска

R3 = Номер зачетной книжки, Номер семестра Оплаты, Название предприятия, Необходимая сумма, оплаченная сумма

R4 = Номер зачетной книжки, Тема работы, Руководитель

R5 = Номер зачетной книжки, Прохождение профосмотра, Наличие прививок

R6 = Номер зачетной книжки, Дата проживания в профилактории

R7 = Номер зачетной книжки, Название предмета, Дата олимпиады, Занятое место

R8 = Номер зачетной книжки, Номер семестра обучения, Название предмета, Количество часов, Преподаватель, Оценка, Передача предмета, Дата экзамена

R9 = Номер зачетной книжки, Номер семестра рейтинга, Номер контрольной, Предмет рейтинга, Балл

R10 = Номер зачетной книжки, Тип особой отметки, Текст особой отметки

С выделенными ключами:

K1 = Специальность, Место обучения, Форма обучения, Факультет, Курс, Номер группы, Номер зачетной книжки, Ф.И.О, Вид обучения,

K2 = Номер зачетной книжки

K3 = Номер зачетной книжки, Номер семестра Оплаты

K4 = Номер зачетной книжки

K5 = Номер зачетной книжки

K6 = Номер зачетной книжки

K7 = Номер зачетной книжки, Название предмета

K8 = Номер зачетной книжки, Номер семестра обучения, Название предмета

К9 = Номер зачетной книжки, Номер семестра рейтинга, Номер контрольной, Предмет рейтинга

К10 = Номер зачетной книжки, Тип особой отметки

На этом нормализацию можно считать завершенной, у нас появилось 10 отношений находящихся в третьей нормальной форме удовлетворяющих определению 6.

В результате нормализации мы получили дополнительные объекты.

Полученная схема отношений схематично показана на Рисунке 2.4.



Рисунок 2.4

На Рисунке 2.5 изображена детальная схема отношений с изображением отдельных атрибутов.

Таким образом в результате нормализации мы получили схему отношений находящуюся в третьей нормальной форме и удовлетворяющую следующим требованиям:

1. Каждая схема отношения  $R_i$  из  $R$  находится в ЗНФ относительно  $F$  (множества функциональных зависимостей  $R$ ).
2. Не существует схемы базы данных с меньшим числом схем чем  $R$ , удовлетворяющей требованию 1.

3. Для любого отношения  $r(R)$ , удовлетворяющего  $F$ ,  
 $r = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie \dots \bowtie \pi_{R_p}(r)$

Таким образом разработку концептуальной модели можно считать завершенной.

### 2.3 Выводы.

1. Построена концептуальная модель содержащая информацию о студентах взятую из отдела кадров, деканатов, учебного отдела.
2. Данная модель является интегрированной для описанных выше подразделений вуза.
3. В данной модели исключено семантическое дублирование данных.

Разработка физической реализации не входит в задачу данной дипломной работы. Однако чтобы упростить разработку приложений основанных на данной концептуальной модели необходимо указать систему управления базами данных наиболее подходящую для физической реализации модели.

Таким образом задача третьей главы провести сравнительный анализ современных систем управления базами данных и выбрать оптимальную для физической реализации разработанной концептуальной модели.



Рисунок 2.5



### **3 ВЫБОР СУБД ДЛЯ ФИЗИЧЕСКОЙ РЕАЛИЗАЦИИ КОНЦЕПТУАЛЬНОЙ МОДЕЛИ**

Опыт применения ЭВМ для построения прикладных систем обработки данных показывает, что самым эффективным инструментом здесь являются не универсальные алгоритмические языки высокого уровня, а специализированные языки для создания систем управления данными. Такие средства обычно включаются в состав систем управления базами данных (СУБД), но они могут существовать и отдельно. СУБД дают возможность пользователям осуществлять непосредственное управление данными, а программистам быстро разрабатывать более совершенные программные средства их обработки. Характеристики готовых прикладных пакетов определяются прежде всего принятой в СУБД организацией данных и типом используемого транслятора.

Прежде чем перейти к рассмотрению конкретных пакетов, уместно уточнить само понятие системы управления базами данных. В наиболее полном варианте такой пакет может иметь следующие компоненты [18]:

Средства поддержки модели данных:

- средства поддержки структур данных;
- примитивы манипулирования данными;
- средства обеспечения целостности данных

Средства администратора БД:

- средства ведения справочника данных;
- средства сбора и анализа статистики функционирования БД;
- средства реорганизации БД;
- Средства контроля целостности и восстановления БД;
- средства конвертирования данных.

Средства разработки приложений;

- средства проектирования БД;
- интерфейс с языками программирования;
- средства генерации программного кода, форм ввода вывода, пользовательских интерфейсов;
- генератор отчетов.

Средства конечного пользователя:

- интерфейс конечного пользователя;
- язык запросов;
- средства обучения и подсказки.

Интерфейсы с другими СУБД.

СУБД делятся на три группы: сетевые реляционные и иерархические.

Нас будут интересовать реляционные СУБД представленные на рынке программных продуктов очень широко. Это, например, такие системы, как Paradox, Microsoft Access, FoxPro, Clipper и др.

Рассмотрим более подробно описанные выше СУБД.

### 3.1 Обзор и анализ современных СУБД

#### СУБД Microsoft Access.

СУБД Access [19] - один из многих программных продуктов фирмы Microsoft. Как и большинство других продуктов данной фирмы данная СУБД полностью ориентирована на пользователя и не требует очень больших познаний в программировании. Рассмотрим общую схему базы данных Access.

Microsoft Access называет объектами все, что может иметь имя. В базе данных Access основными объектами являются таблицы, запросы, формы, отчеты, макросы и модули (Рисунок 3.6) .

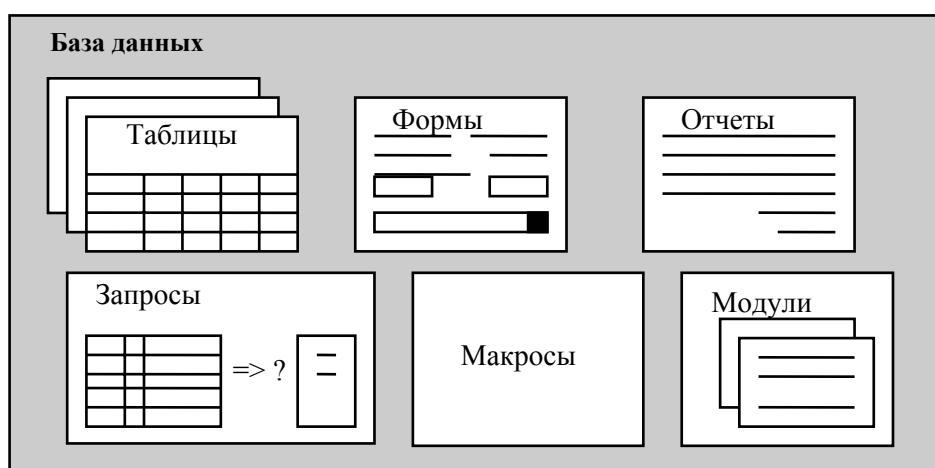


Рисунок 3.6

Если посмотреть на другие СУБД, то можно заметить, что термин база данных обычно относится только к файлам, в которых хранятся данные. В Microsoft Access база данных включает в себя все объекты, связанные с хранимыми данными, в том числе и те, которые вы определяете для автоматизации работы с ними.

В таблицах хранятся данные, которые можно извлекать при помощи запросов. Используя формы можно выводить данные на экран или изменять их. Заметим, что формы и отчеты могут использовать данные непосредственно из таблиц или через запросы. Для выполнения нужных вычислений и преобразования данных запросы могут использовать встроенные функции или функции написанные в Microsoft Access Basic.

Из макросов и модулей можно изменять ход выполнения приложения; открывать, фильтровать и изменять данные в формах и отчетах; выполнять запросы и создавать новые таблицы. Из модуля также можно вызывать библиотечные процедуры Microsoft Windows (DLL), чтобы использовать в приложении не только встроенные в Access функции, но и возможности Windows.

#### Paradox.

СУБД Paradox [20] фирмы Borland предназначен в основном для профессиональных программистов, которые имеют опыт работы с различными языками программирования. В отличие от Access проект на СУБД Paradox не находится в одном файле и для каждого модуля есть свой файл. Проект состоит почти из тех же компонент что и у Access, но есть и некоторые отличия. Основные объекты Paradox - это таблицы, формы, отчеты, запросы, SQL, сценарии, библиотеки.

#### FoxPro.

СУБД FoxPro [21] обладают исключительно высокими скоростными характеристиками и в этом отношении заметно выделяется среди интерпретирующих систем. Набор команд и функций, предлагаемых разработчикам программных продуктов к среде FoxPro достаточно широк. Здесь может быть реализован довольно удобный,

гибкий и эффектный пользовательский интерфейс. В FoxPro поддерживаются разнообразные всплывающие и многоуровневые меню, работа с мышью, реализованы функции низкоуровневого доступа к файлам, управление цветами, настройка принтера, данные могут быть представлены в виде, похожем на электронные таблицы, и т.п. Система также обладает средствами быстрой генерации экранов, отчетов и меню. В пакете имеется компилятор, позволяющий при желании сформировать EXE - файлы готовых программ.

### Clipper.

Первые версии системы Clipper [22] были просто компиляторами к СУБД dBASE. Затем она вместе со своим программным окружением превратилась в полноценную и независимую языковую среду для построения систем обработки данных. После компиляции программ, созданных с помощью Clipper, формируются загрузочные модули типа EXE, которые далее могут запускаться самостоятельно без поддержки их "родительской" СУБД, как это имеет место в системах-интерпретаторах. Недостатком систем-компиляторов являются большие суммарные затраты времени на многократную компиляцию и сборку ("линковку") исходных модулей программы при ее отладке, что очень несколько труд разработчика.

3.2 Определение требований к СУБД для физической реализации концептуальной модели.

Определим ряд требований к СУБД для физической реализации концептуальной модели.

#### 1. Поддержка архитектуры клиент-сервер

Данное условие накладывается так как Архитектура клиент-сервер обладает полезными свойствами перечисленными ниже.

Клиентская и серверная часть системы работают обычно на разных компьютерах. Во-первых, каждый компьютер в системе можно выбрать таким образом, чтобы он лучше всего отвечал требованиям каждого компонента. Таким образом, организация может при минимальных затратах предоставить пользователям простое в применении инструментальное средство для ввода и анализа данных.

Во-вторых, такая система обладает хорошей адаптируемостью и гибкостью в случае неизбежных изменений в программном и аппаратном обеспечении. Предположим, например, что появился новый тип компьютера, дающего при вдвое меньшей цене удвоенную по сравнению с имеющимся сервером производительность. В системах клиент/сервер легко заменить старый сервер на новый, не нарушая функциональности клиентских приложений и продуктивности работы пользователей.

В-третьих, легко масштабировать систему, приспособив ее к изменениям в рабочей группе. Например, если в отделе появляются новые сотрудники, их можно с помощью новых клиентских рабочих станций сразу подключить к сетевой системе.

2. Наличие визуальных инструментов для разработки пользовательского интерфейса.

Данное условие накладывается для облегчения разработки приложения и обеспечения удобного пользовательского интерфейса.

3. Поддержка средств отображения модели данных:

- средства поддержки структур данных: описания отношений и атрибутов, описания ключей, поддержка связей между отношениями;
- примитивы манипулирования данными: обеспечение выполнения каскадных операций обновления и удаления над связанными отношениями
- средства обеспечения целостности данных: обеспечение ссылочной целостности

Данное требование необходимо для отображения разработанной концептуальной модели.

Результаты сравнительного анализа рассмотренных СУБД на основе описанных выше критериев поддержки модели данных, а также дополнительного сервиса разработчика, сведены в следующую таблицу. Характеристики описанных СУБД оценены исходя из пятибалльной системы.

СУБД	Средства поддержки структур данных	Примитивы манипулирования данными	Средства обеспечения целостности данных	Поддержка архитектуры клиент-сервер	Наличие визуальных инструментов для разработки пользовательского интерфейса
<b>Paradox</b>	Полная поддержка, интерфейс недостаточно удобен 4	Все кроме обеспечения каскадных модификаций вычисляемых атрибутов 4	Полностью удовлетворяются реляционные ограничения целостности 4	Поддержка клиент-сервер, запросов SQL, работа с ODBC 3	Визуальные инструменты есть не достаточно выразительны 3
<b>Microsoft Access</b>	Полная поддержка, графический интерфейс разработки 5	Все кроме обеспечения каскадных модификаций вычисляемых атрибутов 4	Полностью удовлетворяются реляционные ограничения целостности 4	Поддержка клиент-сервер, запросов SQL, работа с ODBC 3	Большой набор визуальных инструментов с полуавтоматическими средствами проектирования 5
<b>FoxPro</b>	Полная поддержка, интерфейс недостаточно удобен 3	Поддержка отсутствует 1	Частичные ограничения целостности 3	Поддержка частичная 1	Визуальные инструменты есть не достаточно выразительны 3
<b>Clipper</b>	частичная поддержка, интерфейс отсутствует 2	Поддержка отсутствует 1	Частичные ограничения целостности 3	Поддержка частичная 2	инструменты разработки отсутствуют 0

### 3.3 Выводы.

Результаты сравнительного анализа показывают, что из рассмотренных СУБД наиболее полно выбранным критериям удовлетворяет СУБД Microsoft Access.

Очевидно что ее и необходимо выбрать для физической реализации разработанной концептуальной модели.



На этом задачу поставленную в третьей главе можно считать решенной.

## ЗАКЛЮЧЕНИЕ

В процессе дипломной работы была разработана концептуальная модель клиентской части информационной интегрированной системы “Студент”, и были выполнены следующие задачи:

- анализированы современные подходы к реализации концептуальных моделей баз данных (в результате анализа выбран ER подход к проектированию данных)
- рассмотрены аналоги разрабатываемой модели (при разработке модели учтены сильные и слабые стороны аналогов)
- была проведена разработка и нормализация концептуальной модели (построена интегрированная модель предметной области и при помощи метода синтеза проведена нормализация)
- был проведен сравнительный анализ современных систем управления базами данных (в качестве СУБД для физической реализации предложена СУБД Microsoft Access)

Таким образом в данной дипломной работе создана интегрированная концептуальная модель предметной области при помощи которой на СУБД предложенной в данной работе можно разработать приложение для автоматизации деятельности отделов ВУЗа работающих с информацией о студентах.

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Ермолаев В. А., Нода Е. Г., Толоч В. А., Опыт и перспективы применения интегрированных сетевых технологий в запорожском государственном университете - Доклад на конференции EDUNET '97.
2. Филобок А. П., Ермолаев В.А. Проект модели данных для информационной системы “АБИТУРИЕНТ” / Тезисы докладов научной конференции студентов - Том 6 - часть 1 - Запорожский государственный университет, Запорожье, Украина, 1996.
3. Мейер Д., Теория реляционных баз данных: Пер. С англ. - М.: Мир, 1987, 608 с.
4. Codd E. F. A Relational Model of Data for Large Shared Data Banks. //Comm. ACM, V 13, No 6, 1970, pp. 377-387.
5. Rosenthal A. and Reiner D. Tools and Transformations - Rigorous and Otherwise - for Practical Database Design, //ACM TODS, Vol. 19, No 2 (June 1994), pp. 167-211
6. Fraternali P., Tanca L.. A Structured Approach for the Definition of the Semantics of Active Databases, ACM TODS, Vol. 20, No 4 (Dec. 1995), pp. 414-471
7. Aiken A., Hellerstein J. M., Widom J.. Static Analysis Techniques for Predicting the Behavior of Active Database Rules, ACM TODS, Vol. 20, No 1 (Mar. 1995), pp. 3-41
8. Haskin R. L., Lorie R. A.. On Extending the Functions of a Relational Database System, Proc. ACM SIGMOD Conf., Orlando, 1982
9. Roth M. A., Korth H. F., Silberschatz A.. Extended Algebra and Calculus for Nested Relational Databases, ACM TODS Vol. 13, No. 4 (Dec. 1988), pp. 389-417

10. Clifford J., Croker A., Tuzhilin A.. On Completeness of Historical Relational Query Languages // ACM TODS, Vol. 19, No. 1(March 1994), pp. 64-116
11. Parent C., Rolin H., Yetongnon K., Spaccapietra S.. An ER Calculus for the Entity Relationship Complex Model, / Proc. 8th International Conference on Entity - Relationship Approach, Oct. 18-20, 1989
12. Chen P. P.. The Entity-Relationship Model: Towards a Unified View of Data, ACM TODS, Vol. 1, No. 1, 1976.
13. Dey, D., Sarkar, S. A Probabilistic Relational Model and Algebra
14. Атре Ш., Структурный подход к организации баз данных: Пер с англ. - М. “Финансы и статистика”, 1983, 317 с.
15. Ермолаев В. А. Разработка системы словаря данных для проектирования гибких информационных приложений / Автореферат диссертации на соискание ученой кандидата физико-математических наук, Запорожский государственный университет, Запорожье, Украина, 1994
16. Мухин В. В., Ермолаев В. А. Принципы проектирования системы словаря данных для информационных приложений. В: Тезисы сообщений научной конференции студентов и лекторов. Том 3, Часть 1, Запорожский государственный университет, Запорожье, Украина, 1993, гг. 62-63.
17. Ермолаев В. А. Развитие реляционной модели данных с помощью аппарата вычисляемых атрибутов. В: Тезисы сообщений научной конференции студентов и лекторов. Том 5, Часть 1, Запорожский государственный университет, Запорожье, Украина, 1995, г. 69.
18. Системы управления базами данных для ЕС ЭВМ: справочник / Под общей редакцией В. М. Савинкова. - М.: Финансы и Статистика, 1984. - 224 с.

19. Эффективная работа с Microsoft Access 2 / Перевод с английского - СПб: Питер, 1995 - 350 с.
20. Paradox. User's Guide. Release 3.0. ANSA Software 1989.
21. Foxbase +. Relational Database Management System. Revision 2.10. Reference Manual. Fox Software, Inc. May 1988.
22. Clipper 5.0. Reference - Nantucket Corp. - 1991.