

Министерство образования Республики Беларусь
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Кафедра интеллектуально-информационных технологий

**ОРГАНИЗАЦИЯ ОБРАБОТКИ ДАННЫХ СРЕДСТВАМИ СУБД.
Основные понятия теории баз данных**

Методическое пособие для студентов специальности 53 01 02
«Автоматизированные системы обработки информации»

УДК 681.3 (075.8)

ББК с57

М 91

Учреждение образования "Брестский государственный технический университет"
Кафедра интеллектуальных информационных технологий

**Рекомендовано к изданию редакционно-издательским советом
Брестского государственного технического университета**

Рецензенты:

доцент кафедры математического моделирования БрГУ им. А.С. Пушкина,
к.т.н., доцент Пролиско Е.Е.; доцент кафедры экономики и управления
БрГУ им. А.С. Пушкина, к.т.н., доцент Матюшков Л.П.

Муравьев Г.Л., Хвещук В.И.

М 91 ОРГАНИЗАЦИЯ ОБРАБОТКИ ДАННЫХ СРЕДСТВАМИ СУБД. Основные понятия
теории баз данных. – Брест.: БрГТУ, 2006. - 22 с.

Целью пособия является знакомство студентов с базовыми понятиями информационной технологии баз и банков данных.

1. Общие сведения

Предметная (прикладная) область (ПрО) - часть реального мира (совокупность процессов, объектов, сущностей), представляющая интерес в качестве источников информации для удовлетворения информационных потребностей пользователей, организации управления и подлежащая изучению с целью автоматизации. Например, предприятие, магазин и т.п.

Информационный объект, сущность (ИО) – собирательное понятие некоторого повторяющегося объекта прикладной области, элемент ПрО (основное содержание явления, процесса, предмета, события), о котором нужно хранить информацию для решения пользовательских задач.

ИО задается структурой - совокупностью логически взаимосвязанных свойств (атрибутов), представляющих его качественные и количественные характеристики. Информационный объект может быть реальным (например, СТУДЕНТ, ЗАКАЗЧИК), абстрактным (например, ФОРМУЛЯР читателя).

Атрибут, реквизит (А) - поименованная смысловая характеристика объекта, логическая единица информации, подлежащая хранению. Конкретный экземпляр информационного объекта задается набором атрибутов. Например, объект СТУДЕНТ может иметь атрибуты - номер группы, ФИО, адрес. Все экземпляры ИО одного типа задаются множеством однотипных наборов атрибутов.

Автоматизированная информационная система (АИС) - система, реализующая на базе ЭВМ автоматизированное хранение и обработку данных об объектах предметной области. Включает сами данные (например, в виде баз данных), технические и программные средства их обработки и персонал.

Ранние АИС проектировались от задач (от алгоритмов обработки данных), информация в них хранилась в виде последовательных файлов (на магнитных лентах), а затем в виде файлов с прямым, произвольным доступом, индексно-последовательных файлов (на магнитных дисках), что повысило скорость обработки. Однако данные и программы зависели друг от друга и изменение структуры данных приводило к перепрограммированию АИС. Пользователь управлял данными такой АИС в терминах файловой системы при помощи команд ввода-вывода.

В АИС, основанных на базах данных (с конца 60-х), проектирование ведется от модели предметной области (модели данных), построенной по результатам ее обследования. По модели ПрО автоматически генерируется база данных, обеспечивается независимость данных и программ. Необходимая степень специализированности обработки данных достигается применением языков запросов и языков программирования.

Уровни представления данных - немашинный (концептуальный, пользовательский), внутримашинный (внешний, логический и внутренний, физический) уровни.

Концептуальный уровень (уровень администратора, управленческого персонала предприятия) - описание всей базы данных в абстрактной форме в виде инфологических: концептуальных моделей (КМ), представляющих ПрО в виде сущностей (наборов атрибутов) и связей между ними.

Внешний уровень (уровень пользователей АИС, базы данных и разработчиков) - представление данных в виде схем и подсхем датологических (data) моделей (синонимы - модели данных МД или модели реализации), описывающих хранимые в базе данные и связи между ними в терминах элемент (поле), агрегат данных, запись, набор записей, ключ и т.п.

Физический уровень (уровень программистов, аналитиков базы данных) - описание всей базы данных физическими моделями (ФМ) в терминах блок данных, указатель, кластер и т.п., воспринимаемых программами системы управления базами данных и операционной системы.

База данных, информационная база (англ. **database, БзД**) – данные, совместно используемые задачами, пользователями в рамках АИС (в частности, для принятия управленческих решений), и метаданные - описания самих данных. База данных - информационная модель предметной области, реализованная на основе датологических моделей. Это унифицированная, структурированная совокупность поименованных данных различных типов и связей между ними, отражающая состояние объектов и их отношений в рассматриваемой ПрО. Ее особенности: - интегрированность и взаимосвязанность; - отсутствие дублирования, избыточность; - независимость от программ обработки; - модельность.

БзД обеспечивает однократный ввод первичных, хранимых данных и их многократное, многоаспектное использование. Хранится на машинных носителях в виде специальным образом организованных файлов. Создание и использование БзД реализуется с помощью языков программирования либо централизованно с помощью программ систем управления базами данных (СУБД).

БзД подразделяются: по технологии обработки данных на централизованные и распределенные; по способу доступа к данным на БзД с локальным и удаленным (сетевым) доступом; по типу (форме представления) хранимой информации на документальные и фактографические.

Документальные БзД хранят неструктурированные данные, в них наименьший элемент данных - документ и его реквизиты (например, ключевые слова, позволяющие ускорить поиск). В фактографических БзД наименьший элемент данных – поле, т.е. информация структурирована и хранится по правилам выбранной модели данных (например, хранимые атрибуты объектов задаются парой <название_атрибута, значение_атрибута>).

Структурирование данных - преобразование данных с целью их представления в соответствии с правилами и соглашениями, принятыми в выбранном типе модели данных, который отображается в базе данных. Модель данных - совокупность разрешенных типов структур данных, записей, групповых отношений (связей) между экземплярами записей и операций по их обработке.

Структурные элементы логической организации БзД могут быть классифицированы по терминологии международной ассоциации по языкам систем обработки данных CODASYL.

Данное - некоторый факт (атрибут информационного объекта). **Элемент данных** (поле, ЭД) – наименьшая неделимая поименованная (адресуемая в СУБД) семантически значимая единица логической организации данных, хранящая значение одного атрибута.

Простой или составной **агрегат данных** (АД) – поименованная совокупность разнотипных элементов данных и (или) других АД внутри записи, включая множественные (повторяющиеся) агрегаты данных.

Запись (экземпляр записи) - основной объект модели данных, составной агрегат в виде поименованной совокупности элементов и агрегатов данных, не входящий в состав других агрегатов данных. Запись задается структурой - совокупностью логически связанных ЭД, хранящих конкретные сведения об атрибутах одного экземпляра объекта и рассматриваемых в обработке как целое. Запись может иметь иерархическую структуру, записи одинаковой структуры образуют тип записи.

Набор записей (при хранении файлов) – однотипные записи, хранящие атрибуты всех экземпляров объекта одного типа.

Первичный **ключ** – элемент данных или их комбинация (составной ключ), однозначно определяющие каждую запись в наборе однотипных. Вторичный ключ - один или несколько ЭД, выполняющих роль поисковых или группировочных признаков.

Система управления базами данных (СУБД) - универсальный инструмент для управления БД, поддерживающий один из типов моделей данных и обеспечивающий интерфейс между пользователями и базой данных. Пользователи, обрабатывающие программы, обращаются к данным посредством информационных запросов к СУБД, которая генерирует команды файловой системе ЭВМ на соответствующую обработку данных в файлах БД.

По составу СУБД - комплекс программных и языковых средств для создания БД (описания структуры базы, ее инициализации, загрузки данными), поддержания базы в актуальном состоянии (обеспечения логической и физической целостности данных), совместного эффективного доступа пользователей к данным в рамках предоставленных полномочий.

Языковые средства СУБД: языки описания структуры БД (моделей данных); языки описания запросов для выборки данных и их обработки; языки (макросов и программирования) описания нестандартных алгоритмов обработки данных, событий, создания графического интерфейса пользователя, позволяющие объединить разные операции с БД в единый технологический процесс.

Программные средства СУБД: процессор базы данных, обеспечивающий предобработку (трансляцию) запросов и ядро системы - обработчик запросов; средства генерации базы данных по модели данных; средства конструирования интерфейса БД, экранных форм, предназначенных для ввода, просмотра, обработки данных в диалоговом режиме; генераторы форм и отчетов для вывода данных и результатов обработки на печать; сервисные программы.

По степени универсальности СУБД подразделяют на СУБД общего назначения (FoxPro, Paradox, Access, Oracle) и специализированные СУБД.

Информационные запросы (англ. query) - запросы на выборку информации из базы данных в соответствии с заданными критериями и запросы на действия над выбранной информацией. Описываются в терминах языковых средств СУБД. Существует стандарт описания запросов - язык SQL и упрощенные средства формирования запросов. Например, в Excel - таблица описания критериев отбора записей, в Access - шаблон описания запросов QBE (query by example).

Язык SQL (англ. Structured Queries Language) - международный стандарт, реляционный язык структурированных запросов, реализованный во многих СУБД. Позволяет напрямую управлять общей базой данных, установленной на сервере и поддерживающей концепцию "клиент-сервер", и передавать сквозные SQL-запросы совместимым со спецификацией ODBC SQL-базам данных (таким, как MS SQL Server и Oracle). Не является полноценным языком программирования и при необходимости команды SQL встраиваются в конструкции языка программирования СУБД.

Банк данных (англ. databank, БД) - разновидность АИС, включающая методы и средства поддержания динамической информационной модели ПРО и обеспечивающая хранение и обработку данных средствами технологии баз данных в целях обеспечения информационных потребностей пользователей.

Основные составляющие: - информация, организованная и хранимая в виде баз данных; - программное обеспечение (СУБД); - техническое обеспечение (ЭВМ, сеть); - пользователи, разделенные на группы в зависимости от полномочий доступа к БД; - обслуживающий персонал (администратор БД и системные программисты), обеспечивающий качественное функционирование БД и надлежащий порядок его использования, разработчики и программисты.

Типы БД: централизованные (базы данных хранятся на одной ЭВМ), распределенные (база данных состоит из отдельных или дублируемых частей, хранимых на разных

ЭВМ сети); смешанные; одно и многопользовательские, с локальным и с удаленным (сетевым) способом доступа к данным. Одновременный доступ пользователей к общей БзД предполагает наличие сети и многопользовательской СУБД.

Архитектуры централизованных баз данных с сетевым доступом: файл-сервер (одна ЭВМ сети - файл-сервер хранит централизованную БзД, из которой файлы данных в соответствии с пользовательскими запросами передаются на другие ЭВМ, рабочие станции для их обработки); клиент-сервер (одна ЭВМ сети - клиент-сервер, кроме функций файл-сервера обеспечивает по SQL-запросам поиск, извлечение данных из файлов базы данных сервера и передачу их пользователям).

БД разделяют в зависимости от: сфер применения (медицинские; экономические и т.д.); задач, решаемых в конкретной области (складские; биржевые и т.д.); от характера информационных запросов (системы оперативной обработки информации, автоматизированные информационные системы); от типа хранимой информации (документальные, фактографические) и т.д.

Характеристики баз и банков данных. Пользовательские характеристики: производительность системы; уровень потребляемых ресурсов, размер базы данных; достоверность, актуальность, корректность информации; безопасность, защищенность, восстанавливаемость данных, возможность разграничения полномочий в доступе к данным, шифрования программ и данных; возможность работы в многопользовательских средах; наличие средств блокировки базы данных, файла, записи, поля, средств обновления информации после ее модификации.

Внутренние характеристики (на примере реляционных баз данных): - целостность, точность и непротиворечивость данных; - ссылочная целостность. Целостность – это свойство записей одного типа (ИО), когда каждому значению первичного ключа соответствует одна запись. Ссылочная целостность – это свойство групп записей двух связанных ИО, когда в связанном информационном объекте каждому значению ключа соответствует одна запись.

Производительность баз и БД оценивается временем выполнения запросов, скоростью поиска информации, максимальным числом параллельных обращений к данным в многопользовательском режиме, временем генерации отчетов, временем выполнения операций импортирования базы данных из других форматов, скоростью выполнения таких массовых операций, как обновление, вставка, удаление данных и т.п.

Жизненный цикл базы данных включает фазы анализа предметной области, проектирования, реализации базы данных и ее эксплуатации.

2. Модели предметной области

Инфологические, информационно-логические, концептуальные модели (ИЛМ, КМ) - описательные неформальные семантические модели структуры предметной области. Описывают объекты, представляющие интерес в рассматриваемой ПрО и выявляемые путем анализа данных о ПрО, и связи между объектами. КМ является представлением точки зрения пользователя на ПрО и не зависит ни от программного обеспечения СУБД, ни от технических решений. КМ описывает данные, подлежащие хранению в БзД, и в процессе проектирования БзД трансформируется в модель данных, совместимую с выбранной СУБД.

Основные виды КМ хорошо структурированной информации: ER-модели, диаграммы потоков данных, IDEF-модели. Широко используются ER-модели ("сущность-связь"). П. Чена и их модификации, где сущности, атрибуты, связи изображаются графически соответственно прямоугольниками, овалами и ромбами. В прямоугольники сущностей вписывают их названия и названия первичных ключей. Основные виды КМ плохо структу-

рированной информации: семантические сети, фреймы.

Связь в КМ - соответствие, отношение, возникающие между информационными объектами и определяемое их природой. Связь есть, если логически взаимосвязаны экземпляры объектов. Связь двух информационных объектов – бинарные, n информационных объектов – n -арные. Существуют бинарные связи разных типов.

Связь **один к одному (1:1)**, когда в каждый момент времени одному экземпляру первого ИО соответствует не более одного экземпляра второго ИО и наоборот (например, связь между объектами СТУДЕНТ и СЕССИЯ). Такие ИО можно объединять путем объединения их атрибутов и выбором в качестве ключевого любого из ключей исходных объектов.

Связь **один ко многим (1:M)**, когда одному экземпляру первого объекта соответствует 0, 1 или более экземпляров второго объекта, но каждый экземпляр второго объекта связан не более чем с одним экземпляром первого объекта (например, связь между объектами СОТРУДНИК и ДОЛЖНОСТЬ).

Связь **многие ко многим (M:M)**, когда одному экземпляру первого объекта соответствует 0, 1 или более экземпляров второго объекта и наоборот (например, связь между объектами ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ).

Датологические, логические модели, модели данных (МД) – формальные модели структуры предметной области, разновидность КМ, которая может быть реализована средствами конкретной СУБД. Модели данных описывают данные и логические связи между элементами данных вне зависимости от их содержания и среды хранения и в соответствии с набором принципов и правил, определяющих организацию логической структуры данных при хранении в базе данных. Соответственно тип модели данных определяет допустимые структуры данных, связи между ними, операции над данными и правила ограничения целостности данных. При создании базы данных МД отображает в физическую модель.

Основные МД фактографической информации: объектно-ориентированные, графовые (сетевые, иерархические), теоретико-множественные (реляционные, бинарных отношений).

Основные МД документальной информации: модели инвертированной организации, схемно- и контекстно-определяемой структуры.

В простых АИС применяют файловые модели (наборы независимых файлов однотипных записей с одноуровневой структурой), в СУБД широко используют реляционные модели.

Иерархическая МД организует данные в виде древовидной иерархии узлов, где узел - набор записей, описывающих объекты одного типа. Допустимы связи типа 1: n , где каждый узел подчиняется только одному узлу вышестоящего уровня. Основные элементы такой МД - атрибут, агрегат, запись (иерархическая или линейная), набор записей. Число уровней ограничено, информация дублируется, поиск записей не от корневого узла затруднен.

Сетевая МД (расширение иерархической) организует данные в виде сетевой структуры, допуская для каждого узла существование произвольного числа предшествующих узлов и наличие связей любого типа. Основные элементы - атрибут, агрегат, запись (чаще линейная), набор записей. Поиск записей может вестись от любого узла. Достоинства указанных МД – компактность, быстрое действие. Недостатки – зависимость от данных (слабая универсальность), отсутствие строгой математической основы, что при модификации БД в процессе эксплуатации может потребовать ее перепроектирование.

Реляционная модель данных (англ. relation - отношение) – организует данные в виде реляционных таблиц (РТ), связанных через общие поля данных (ссылки, ссылочные

ключи), если в них есть одинаковые (по смыслу, типу данных и иногда по имени) поля. Каждая таблица описывает информационный объект одного типа и при хранении образует файл. Типы связей любые, кроме т:п. Основные элементы – поле (атрибут), запись (линейная), таблица.

Достоинства: простота структур данных, представленных в виде привычных пользователю таблиц; применение для обработки данных алгебры отношений, реляционного исчисления, основанного на работах Е.Кодда, что обеспечивает живучесть базы в процессе эксплуатации, сохранение ее целостности и ссылочной целостности. База данных, созданная на основе реляционной модели данных, называется реляционной (relational database, РБД).

Реляционная таблица (РТ) - двумерная таблица из описаний полей и однотипных строк (записей, кортежей), содержащих информацию - отношения атрибутов каждого экземпляра объекта одного типа. Столбцы РТ соответствуют полям (атрибутам отношений, доменам). РТ имеет первичный и ссылочные ключи. В параграфе 4 приведена схема реляционной базы данных в виде таблиц (отношений) СОТРУДНИК, ОТДЕЛ и т.п. У таблиц есть совпадающие ключи, т.е. между ними существуют связи (например, у таблицы СОТРУДНИК есть первичный ключ – НомерСотрудника и внешний ключ - НомерОтдела, обеспечивающий ее связь с таблицей ОТДЕЛ).

Каждая реляционная таблица должна отвечать требованиям нормализации – соответствовать нормальной форме (НФ). Требование первой нормальной формы - иметь уникальный ключ и атомарные поля (агрегаты данных запрещены). Требование второй НФ – не иметь частичных зависимостей. Т.е. между описательными атрибутами как и между ключевыми не допустимы функциональные зависимости, а каждый описательный атрибут должен функционально полностью зависеть от значения всего ключа (если он составной). Требование третьей НФ - описательные поля не должны зависеть от ключа транзитивно.

Соблюдение этих требований обеспечивает целостность РТ - поддержание безыбыточности, корректности данных при внесении изменений в базу данных. Связная целостность РТ обеспечивается запретом на: - добавление записи в подчиненную таблицу с несуществующим в главной таблице значением ключа связи; - изменение значения ключа связи в записи главной таблицы или удаление записи, с которой связаны записи подчиненной таблицы. Соответственно в процессе разработки таблицы данных анализируются на соответствие требованиям нормализации и при необходимости нормализуются (подвергаются последовательному преобразованию в 1НФ, 2НФ, 3НФ и т.д.).

Физические модели – модели реализации базы данных в реальной компьютерной системе. Это, например, модели файловой организации, страничной организации БзД (в них данные перемещаются между внешней и оперативной памятью соответственно физическими записями или страницами фиксированной длины) и др.

3. Понятие о проектировании баз данных

Проектирование – процесс разработки структуры базы данных в соответствии с пользовательскими требованиями. Фазы проектирования - инфологическое (информационно-логическое) и датологическое проектирование (проектирование реализации).

Инфологическое проектирование.

1. Этап анализа предметной области и планирования разработки базы данных. **Результат** - техническое задание на разработку, фиксирующее объем, стоимость, целесообразность и технологическую осуществимость работ, концептуальные требования к БзД, к содержанию и процессу обработки данных в базе данных, типовые запросы и

процедуры обработки, решаемые задачи, словари данных. Подход - выявление путем анализа сведений о предметной области и ее документооборота информационных потребностей пользователей, например, в виде описаний запросов с указанием их интенсивностей и объемов потребляемых данных, решаемых задач, источников информации для их удовлетворения (ИО).

2. Этап концептуального проектирования. Результат – концептуальная схема (КМ) информационной структуры предметной области, представляющая видение пользователей и не зависящая от СУБД и технической среды реализации базы данных.

Основные подходы представлены ниже.

А. Восходящий (интуитивный) подход к выделению типовых информационных объектов для БЗД с малым числом объектов и атрибутов. Здесь типовые информационные объекты, соответствующие реальным объектам, выявляются интуитивно.

Формальный подход к выделению информационных объектов предполагает шаги: - анализ структуры и состава реквизитов документов; - определение функциональных зависимостей реквизитов и выделение групп зависимых (описательных) и независимых (ключевых) реквизитов (при транзитивной зависимости реквизит одновременно попадает в обе группы); - группирование всех описательных реквизитов (зависимых от одних и тех же ключевых) в отношении (информационные объекты) с последующей нормализацией полученных таблиц.

Б. Нисходящий подход. Базируется на объектном представлении предметной области в терминах понятий объектно-ориентированного проектирования (абстракция и иерархия абстракций, агрегация, обобщение); на анализе задач и моделировании сущностей в терминах семантических ER-моделей "сущность-связь" и их расширения EER-моделей. Подход: выделение высокоуровневых сущностей и их связей с последующим нисходящим уточнением. Шаги: - выделение задач, для решения которых строится база данных, и их потребностей в данных; выделение локальных предметных областей для отдельных задач; - построение локальных представлений предметных областей (их концептуальных моделей) путем выделения информационных объектов и их связей; - описание атрибутов информационных объектов, выделение ключевых и описательных атрибутов; - объединение локальных представлений в единую модель.

Датологическое проектирование (на примере РБД).

1. Этап выбора типа модели данных и СУБД (здесь выбрана реляционная модель данных, описывающая объекты записями линейной структуры с разрешенными бинарными связями).

2. Этап логического проектирования – преобразование КМ в логическую МД (в структуры данных и форматы, поддерживаемые СУБД). Результат – СУБД-ориентированное описание данных (схема данных в терминах выбранного типа МД и языка описания данных).

Основные подходы представлены ниже.

А. Традиционный подход. Практически означает модификацию восходящего подхода концептуального проектирования (подход Кодда) и состоит в создании сразу реляционной модели данных за счет нормализации выявленных отношений (таблиц).

Б. Нисходящий подход - "механическое" преобразование КМ (с нормализацией таблиц в упрощенном варианте для проверки корректности полученной МД). Здесь шаги: а) устранение рекурсивных, избыточных, не бинарных связей и связей "многие к многим"; б) проверка модели на корректность объединения атрибутов в каждой реляционной таблице (последовательной нормализацией), на реализуемость транзакций пользователей, на обеспечение целостности.

3. Этап проектирования программ, приложений (ведется параллельно с проектированием базы данных). Результат – описание запросов, интерфейса базы данных (форм,

отчетов), программных модулей, пользовательского интерфейса банка данных, например, графического интерфейса пользователя (ГИП). Шаги: а) проектирование объектов базы данных (запросов, форм, отчетов); б) программирование задач (макросов, модулей); в) проектирование ГИП.

4. Этап физического проектирования. Результат – выбор способов размещения и хранения данных на физических носителях и методов доступа к ним, средств защиты, техники индексирования и т.п. Подход – преобразование МД в физическую модель (физические средства ЭВМ, носители информации).

Реализация базы данных. Описание на языке описания данных СУБД таблиц, схем данных, фиксирующих связи между таблицами, параметров целостности, каскадного обновления и удаления записей. Загрузка данных, тестирование, эксплуатация и сопровождение базы данных.

4. Примеры проектирования баз данных

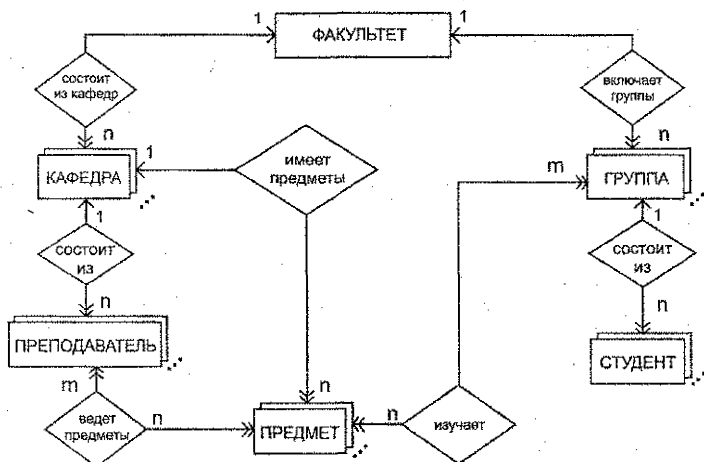


Рис. 1. Факультет (закрепление предметов). Модель "сущность-связь"

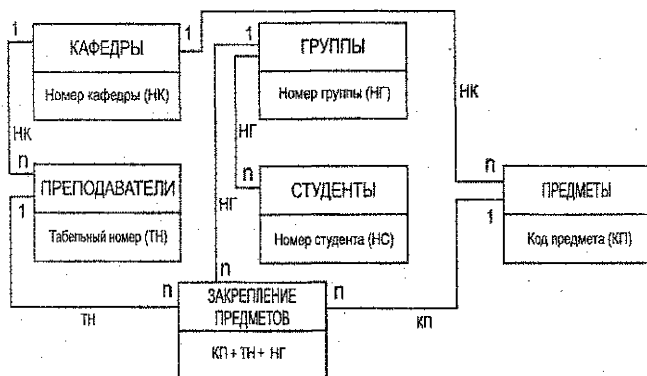


Рис. 2. Факультет (закрепление предметов). Реляционная модель данных

Некоторые типовые этапы проектирования реляционных баз данных иллюстрируются на примере трех предметных областей ("ФАКУЛЬТЕТ", "БИБЛИОТЕКА", "КАДРЫ"), соответствующие модели (концептуальные и логические) приведены ниже.

Предметная область "ФАКУЛЬТЕТ" представляет учебный процесс в части распределения и закрепления предметов текущего семестра между преподавателями и группами студентов. Соответственно на рис. 1 приведена ER-модель предметной области, а на рис. 2 схема реляционной модели данных.

Предметная область "БИБЛИОТЕКА" представлена задачами по обработке информации о книгах библиотечного фонда, данных о читателях, а также обслуживанию читателей (например, выдаче книг и т.п.). На рис. 3 приведена ER-модель предметной области, а на рис. 4-12 результаты проектирования соответствующих схем реляционной модели данных с использованием разных подходов.

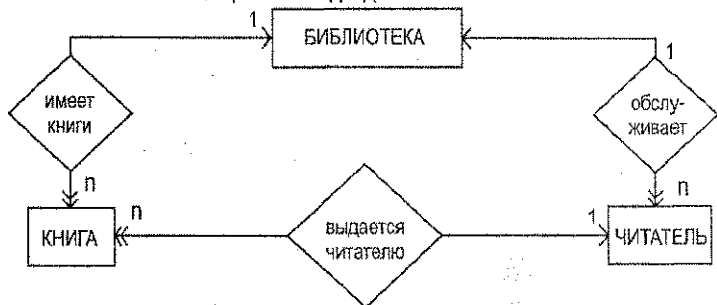


Рис. 3. Библиотека. Модель "сущность-связь"

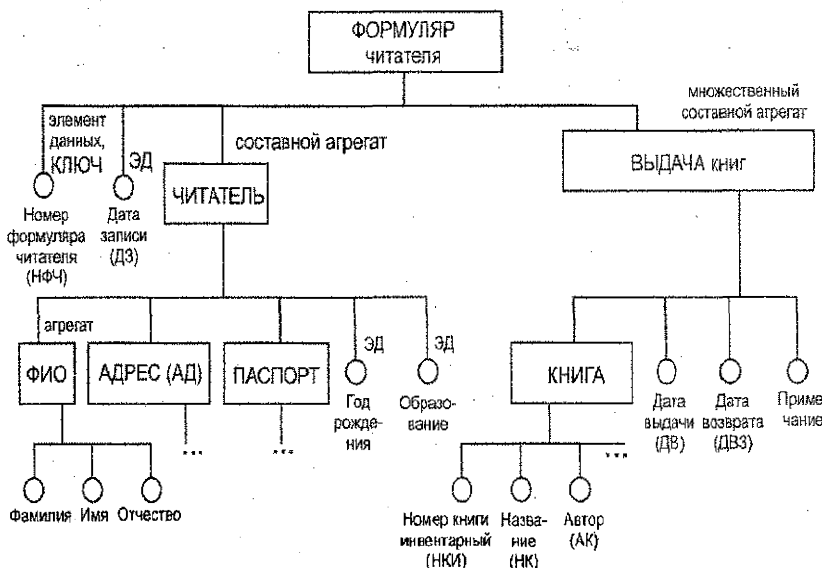


Рис. 4. Библиотека. Документ "Формуляр читателя" (структура записи)

ключи), если в них есть одинаковые (по смыслу, типу данных и иногда по имени) поля. Каждая таблица описывает информационный объект одного типа и при хранении образует файл. Типы связей любые, кроме м:п. Основные элементы – поле (атрибут), запись (линейная), таблица.

Достоинства: простота структур данных, представленных в виде привычных пользователю таблиц; применение для обработки данных алгебры отношений, реляционного исчисления, основанного на работах Е.Кодда, что обеспечивает живучесть базы в процессе эксплуатации, сохранение ее целостности и ссылочной целостности. База данных, созданная на основе реляционной модели данных, называется реляционной (relational database, РБД).

Реляционная таблица (РТ) - двумерная таблица из описаний полей и однотипных строк (записей, кортежей), содержащих информацию - отношения атрибутов каждого экземпляра объекта одного типа. Столбцы РТ соответствуют полям (атрибутам отношений, доменам). РТ имеет первичный и ссылочные ключи. В параграфе 4 приведена схема реляционной базы данных в виде таблиц (отношений) СОТРУДНИК, ОТДЕЛ и т.п. У таблиц есть совпадающие ключи, т.е. между ними существуют связи (например, у таблицы СОТРУДНИК есть первичный ключ – НомерСотрудника и внешний ключ - НомерОтдела, обеспечивающий ее связь с таблицей ОТДЕЛ).

Каждая реляционная таблица должна отвечать требованиям нормализации – соответствовать нормальной форме (НФ). Требование первой нормальной формы - иметь уникальный ключ и атомарные поля (агрегаты данных запрещены). Требование второй НФ – не иметь частичных зависимостей. Т.е. между описательными атрибутами как и между ключевыми не допустимы функциональные зависимости, а каждый описательный атрибут должен функционально полно зависеть от значения всего ключа (если он составной). Требование третьей НФ - описательные поля не должны зависеть от ключа транзитивно.

Соблюдение этих требований обеспечивает целостность РТ - поддержание безыбыточности, корректности данных при внесении изменений в базу данных. Связная целостность РТ обеспечивается запретом на: - добавление записи в подчиненную таблицу с несуществующим в главной таблице значением ключа связи; - изменение значения ключа связи в записи главной таблицы или удаление записи, с которой связаны записи подчиненной таблицы. Соответственно в процессе разработки таблицы данных анализируются на соответствие требованиям нормализации и при необходимости нормализуются (подвергаются последовательному преобразованию в 1НФ, 2НФ, 3НФ и т.д.).

Физические модели – модели реализации базы данных в реальной компьютерной системе. Это, например, модели файловой организации, страничной организации БзД (в них данные перемещаются между внешней и оперативной памятью соответственно физическими записями или страницами фиксированной длины) и др.

3. Понятие о проектировании баз данных

Проектирование – процесс разработки структуры базы данных в соответствии с пользовательскими требованиями. Фазы проектирования - инфологическое (информационно-логическое) и датологическое проектирование (проектирование реализации).

Инфологическое проектирование.

1. Этап анализа предметной области и планирования разработки базы данных. Результат - техническое задание на разработку, фиксирующее объем, стоимость, целесообразность и технологическую осуществимость работ, концептуальные требования к БзД, к содержанию и процессу обработки данных в базы данных, типовые запросы и

процедуры обработки, решаемые задачи, словари данных. Подход - выявление путем анализа сведений о предметной области и ее документооборота информационных потребностей пользователей, например, в виде описаний запросов с указанием их интенсивностей и объемов потребляемых данных, решаемых задач, источников информации для их удовлетворения (ИО).

2. Этап концептуального проектирования. Результат – концептуальная схема (КМ) информационной структуры предметной области, представляющая видение пользователей и не зависящая от СУБД и технической среды реализации базы данных.

Основные подходы представлены ниже.

А. Восходящий (интуитивный) подход к выделению типовых информационных объектов для БД с малым числом объектов и атрибутов. Здесь типовые информационные объекты, соответствующие реальным объектам, выявляются интуитивно.

Формальный подход к выделению информационных объектов предполагает шаги: - анализ структуры и состава реквизитов документов; - определение функциональных зависимостей реквизитов и выделение групп зависимых (описательных) и независимых (ключевых) реквизитов (при транзитивной зависимости реквизит одновременно попадает в обе группы); - группирование всех описательных реквизитов (зависимых от одних и тех же ключевых) в отношении (информационные объекты) с последующей нормализацией полученных таблиц.

Б. Нисходящий подход. Базируется на объектном представлении предметной области в терминах понятий объектно-ориентированного проектирования (абстракция и иерархия абстракций, агрегация, обобщение); на анализе задач и моделировании сущностей в терминах семантических ER-моделей "сущность-связь" и их расширения EER-моделей. Подход: выделение высокоровневых сущностей и их связей с последующим нисходящим уточнением. Шаги: - выделение задач, для решения которых строится база данных, и их потребностей в данных; выделение локальных предметных областей для отдельных задач; - построение локальных представлений предметных областей (их концептуальных моделей) путем выделения информационных объектов и их связей; - описание атрибутов информационных объектов, выделение ключевых и описательных атрибутов; - объединение локальных представлений в единую модель.

Датологическое проектирование (на примере РБД).

1. Этап выбора типа модели данных и СУБД (здесь выбрана реляционная модель данных, описывающая объекты записями линейной структуры с разрешенными бинарными связями).

2. Этап логического проектирования – преобразование КМ в логическую МД (в структуры данных и форматы, поддерживаемые СУБД). Результат – СУБД-ориентированное описание данных (схема данных в терминах выбранного типа МД и языка описания данных).

Основные подходы представлены ниже.

А. Традиционный подход. Практически означает модификацию восходящего подхода концептуального проектирования (подход Кодда) и состоит в создании сразу реляционной модели данных за счет нормализации выявленных отношений (таблиц).

Б. Нисходящий подход - "механическое" преобразование КМ (с нормализацией таблиц в упрощенном варианте для проверки корректности полученной МД). Здесь шаги: а) устранение рекурсивных, избыточных, не бинарных связей и связей "многие к многим"; б) проверка модели на корректность объединения атрибутов в каждой реляционной таблице (последовательной нормализацией), на реализуемость транзакций пользователей, на обеспечение целостности.

3. Этап проектирования программ, приложений (ведется параллельно с проектированием базы данных). Результат – описание запросов, интерфейса базы данных (форм,

отчетов), программных модулей, пользовательского интерфейса банка данных, например, графического интерфейса пользователя (ГИП). Шаги: а) проектирование объектов базы данных (запросов, форм, отчетов); б) программирование задач (макросов, модулей); в) проектирование ГИП.

4. Этап физического проектирования. Результат – выбор способов размещения и хранения данных на физических носителях и методов доступа к ним, средств защиты, техники индексирования и т.п. Подход – преобразование МД в физическую модель (физические средства ЭВМ, носители информации).

Реализация базы данных. Описание на языке описания данных СУБД таблиц, схем данных, фиксирующих связи между таблицами, параметров целостности, каскадного обновления и удаления записей. Загрузка данных, тестирование, эксплуатация и сопровождение базы данных.

4. Примеры проектирования баз данных

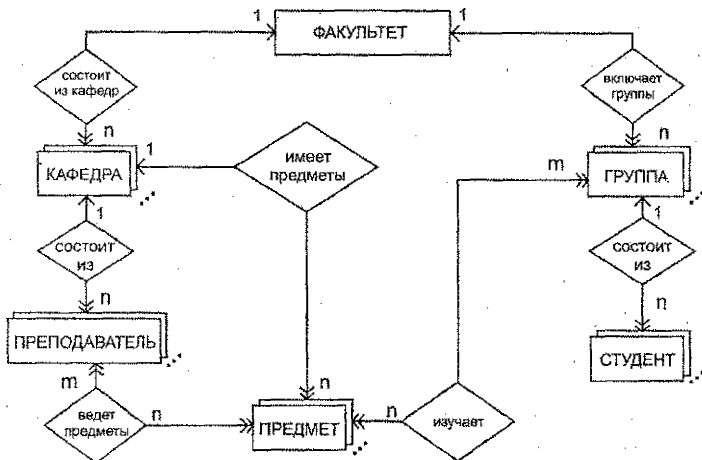


Рис. 1. Факультет (закрепление предметов). Модель "сущность-связь"

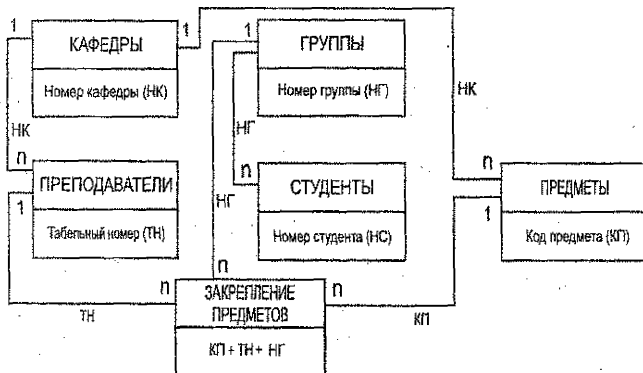


Рис. 2. Факультет (закрепление предметов). Реляционная модель данных

Некоторые типовые этапы проектирования реляционных баз данных иллюстрируются на примере трех предметных областей ("ФАКУЛЬТЕТ", "БИБЛИОТЕКА", "КАДРЫ"), соответствующие модели (концептуальные и логические) приведены ниже.

Предметная область "ФАКУЛЬТЕТ" представляет учебный процесс в части распределения и закрепления предметов текущего семестра между преподавателями и группами студентов. Соответственно на рис. 1 приведена ER-модель предметной области, а на рис. 2 схема реляционной модели данных.

Предметная область "БИБЛИОТЕКА" представлена задачами по обработке информации о книгах библиотечного фонда, данных о читателях, а также обслуживанию читателей (например, выдаче книг и т.п.). На рис. 3 приведена ER-модель предметной области, а на рис. 4-12 результаты проектирования соответствующих схем реляционной модели данных с использованием разных подходов.

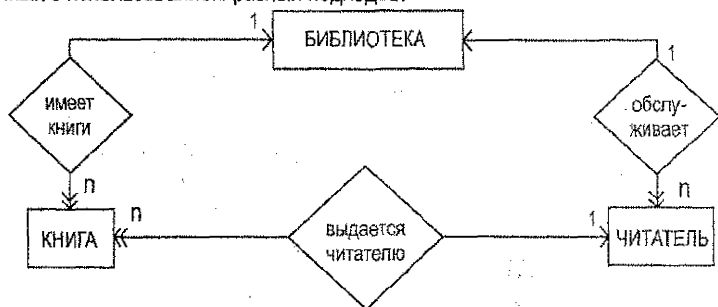


Рис. 3. Библиотека. Модель "сущность-связь"

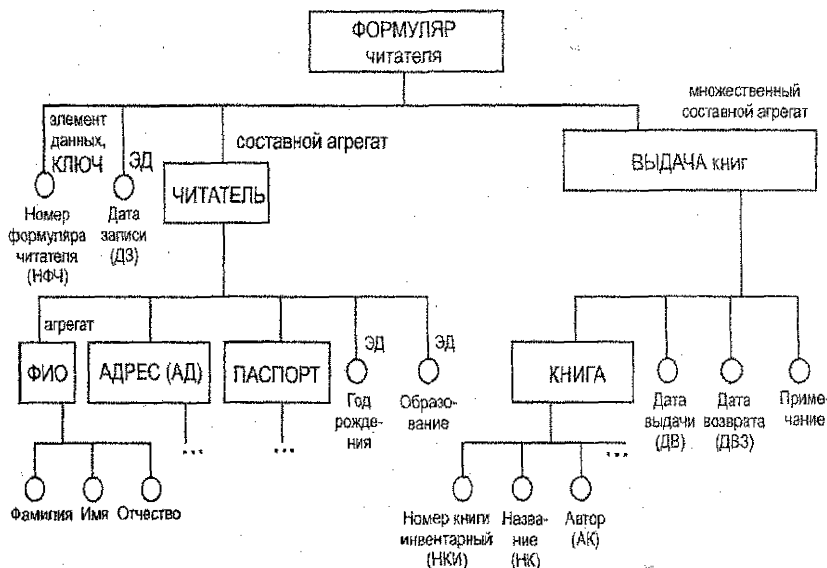


Рис. 4. Библиотека. Документ "Формуляр читателя" (структура записи)

Результаты восходящего проектирования на базе выделения информационных объектов (ИО) путем анализа первичных документов (формуляра читателя, инвентарной книги, книжного каталога и др.), их реквизитов (атрибутов) и зависимостей реквизитов представлены на рис. 4-6. Проектирование проведено на примере анализа одного документа - формуляра читателя (ФЧ).

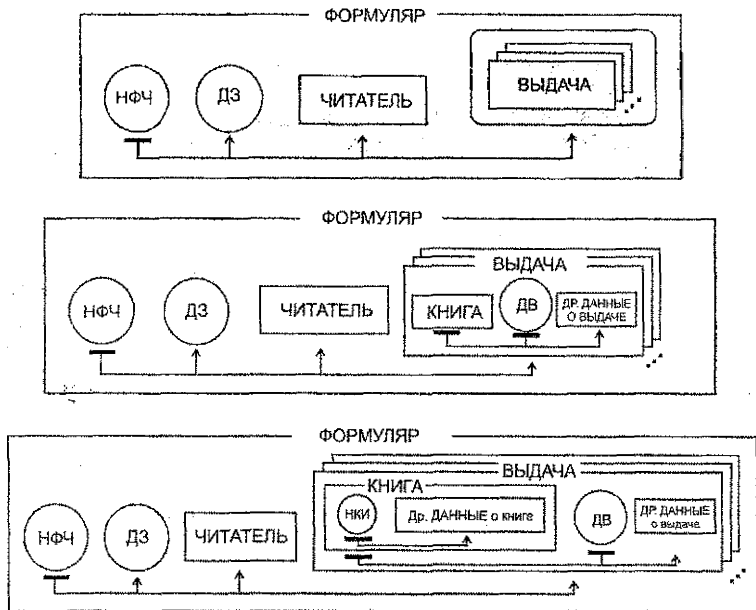


Рис. 5. Библиотека. Формуляр читателя (анализ зависимостей элементов данных)

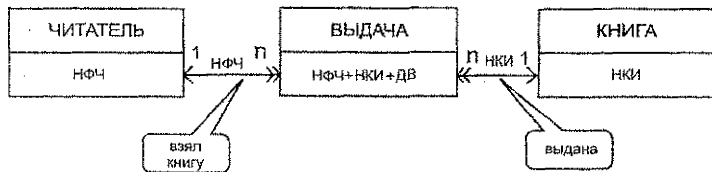


Рис. 6. Библиотека. Формуляр читателя. Модель "сущность-связь"

Структура записей формуляра читателя показана на рис. 4. Она содержит агрегаты и множественные агрегаты, не допустимые в реляционных моделях данных.

Результаты последовательного анализа реквизитов формуляра читателя и их зависимостей приведены на рис. 5. Анализ позволил выделить три группы информационных объектов - КНИГА, ЧИТАТЕЛЬ, ВЫДАЧА книг. Они определяются ключевыми полями НомерКнигиИнвентарный (НКИ), НомерФормуляраЧитателя (НФЧ) и составным ключом (НКИ+НФЧ+ДВ) соответственно. Здесь ДВ - ДатаВыдачи книги. Модель данных для формуляра читателя показана на рис. 6. В дальнейшем следует выполнить аналогичный анализ и других документов, представляющих предметную область, и при необходимости скорректировать итоговую модель данных.

Результаты восходящего проектирования на базе выделения информационных объектов путем анализа и нормализации таблиц первичных документов показаны на рис. 7 (также на примере формуляра читателя).

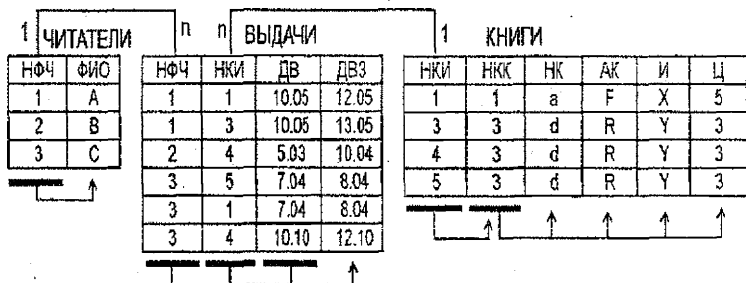
Представим, что выдача-возврат книг фиксируется в виде табличных данных (по типу формуляра читателя, например, в электронной таблице Excel), содержащих поля, указанные на рис. 7а (с добавлением к полю - НомерКнигиИнвентарный дополнительный поля - НомерКнигиКаталожный). Так как в библиотеке может быть произвольное число книг одного типа, то конкретный экземпляр книги будет идентифицироваться полем НКИ в инвентарной книге. А тип книги будет определяться полем НКК в каталоге книг, где и приводится описание книги каждого типа.

КИНОВЫДАЧИ

Номер формуляра читателя (НФЧ)	ФИО читателя (ФИО)	Номер книги инвентарный (НКИ)	Номер книги каталожный (НКК)	Название книги (НК)	Автор книги (АК)	Издательство (И)	Цена (Ц)	Дата выдачи (ДВ)	Дата возврата (ДВЗ)
1	A	1	1	a	F	X	5	10.05	12.05
1	A	3	3	d	R	Y	3	10.05	13.05
2	B	4	3	d	R	Y	3	5.03	10.04
3	C	5	3	d	R	Y	3	7.04	8.04
3	C	1	1	a	F	X	5	7.04	8.04
3	C	4	3	d	R	Y	3	10.10	12.10



а)



б)



в)

Рис. 7. Библиотека. Реляционные модели, нормальные формы: а) первая (таблица Книговывадачи - аналог формуляра читателя); б) вторая; в) третья

Анализ таблицы показывает наличие частичных зависимостей (рис. 7,а) ее полей от полей НКИ и НФЧ, входящих в составной ключ таблицы. Поэтому устранение частичных зависимостей приводит к новой схеме базы данных (рис. 7,б). Последующий анализ схемы показывает, что в таблице КНИГИ присутствует транзитивная зависимость некоторых реквизитов книги от ее ключевого поля НомерКнигиКаталожный. Устранение транзитивной зависимости приводит к получению нормализованной модели данных (рис. 7,в) для представления формуляра читателя ФЧ (приведенные на рис. 7,в таблицы соответствуют требованиям третьей нормальной формы).

Результаты нисходящего проектирования, который ведется от анализа задач, решаемых базой данных, приведены на рис. 8 - 12. Здесь (рис. 8) первоначально получены локальные концептуальные модели однотипных групп задач в терминах "сущность-связь". Например, модель задач обслуживания читателей представлена на рис.8, в, обработки книг - на рис. 8, а, а ведения данных о читателях - на рис. 8, б. Затем они объединены в обобщенную концептуальную модель (рис. 9) и переведены в реляционную модель данных (рис. 10 - 12).

Схема реляционной модели данных частного случая - для решения задач текущего времени, т.е. всех, кроме ведения формуляра читателя ("истории" выдач и возвратов книг), представлена на рис. 10.

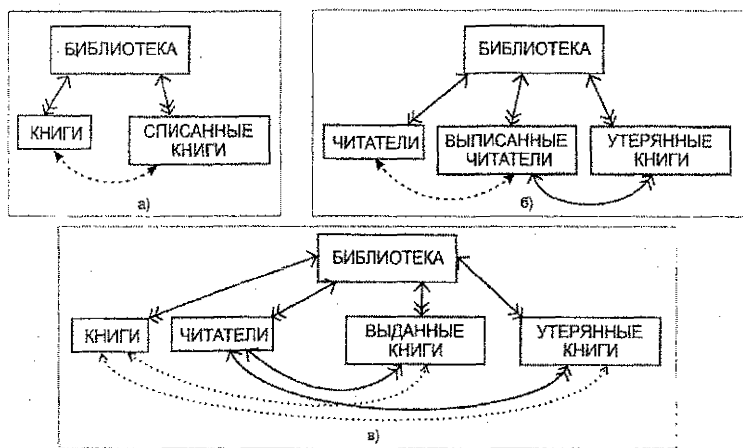


Рис. 8. Библиотека. Локальные концептуальные модели задач: а) работа с книгой; б) работа с читателем; в) обслуживание читателей

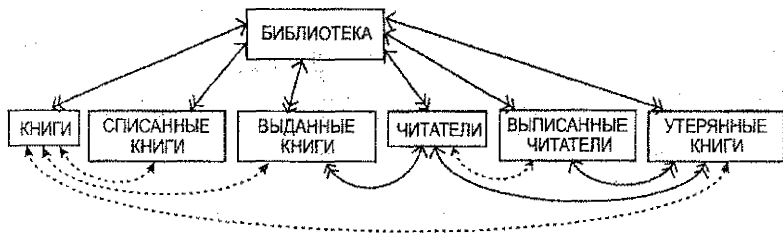


Рис. 9. Библиотека. Концептуальная модель



Рис. 10. Библиотека. Реляционная модель (без учета ведения формуляра читателя)

Попытка информационной поддержки в этой схеме всех задач, включая ведение "истории" выдач и возвратов книг, приводит к многозначной зависимости таблиц **КНИГИ** и **ЧИТАТЕЛИ**. В результате ее устранения путем введения новой таблицы для объектной связи **ВЫДАЧИ** книг может быть получена новая схема модели данных (рис. 11).

Такая же схема, но для случая введения новых таблиц (например, **СПИСАННЫЕ_КНИГИ**, **УТЕРЯННЫЕ_КНИГИ**, **ВЫПИСАННЫЕ_ЧИТАТЕЛИ**) для хранения соответствующих экземпляров объектов представлена на рис. 12. При этом, если экземпляры объектов дублируются в основных таблицах (например, **КНИГИ**, **ЧИТАТЕЛИ**), то новые и основные таблицы связаны по типу один к одному (эти связи показаны пунктиром), а если нет, то новые таблицы автономны по отношению к основным (и связей между ними нет).

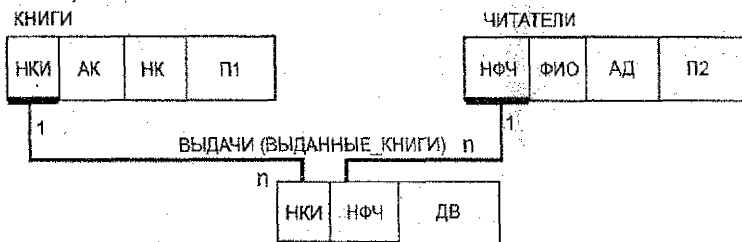


Рис. 11. Библиотека. Реляционная модель (с учетом ведения формуляра читателя)

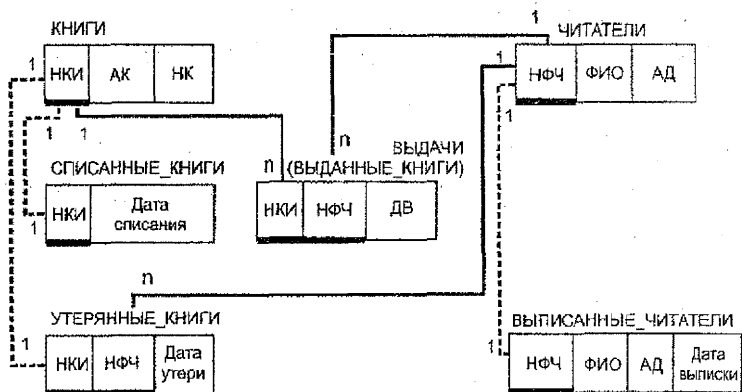


Рис. 12. Библиотека. Реляционная модель (с выделением дополнительных сущностей)

Предметная область "КАДРЫ" представлена в части хранения и обработки кадровых данных о сотрудниках, отделах, в которых они работают, и проектах, в которых они могут участвовать. Соответствующая ER-модель приведена на рис. 13.

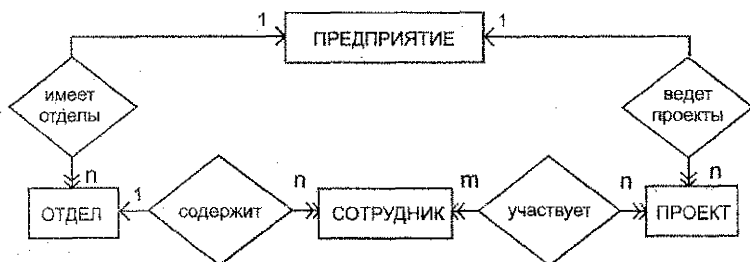


Рис. 13. Кадры. Модель "сущность-связь"

Возможная схема реляционной модели данных этой предметной области приведена на рис. 14.

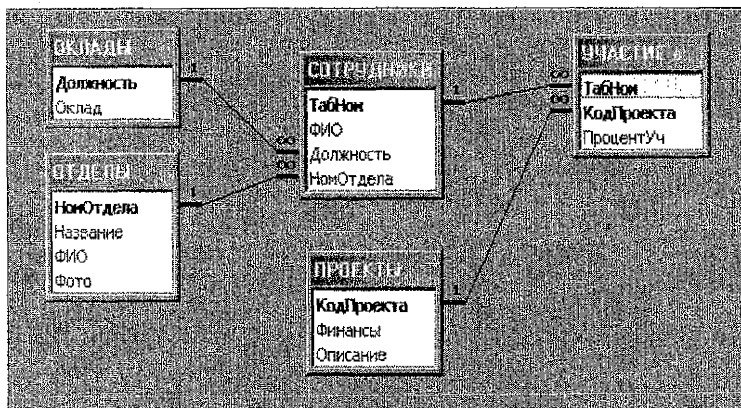


Рис. 14. Кадры. Реляционная модель

Она может быть получена, например, анализом документов (штатного расписания, расстановки кадров, списков сотрудников отделов, списков отделов, проектов, участия в проектах и др.) или задач (ведение данных о сотрудниках; отделах; проектах; ведение штатного расписания), возложенных на систему.

ПРИЛОЖЕНИЕ 1. Задания для самостоятельного выполнения

1. Нормализовать заданную таблицу (определить ключи, устранить транзитивные и частичные зависимости)

сведения о результатах экзаменационных сессий								
Факультет	Курс	Группа	ФИО	Семестр	Предмет	Оценка	Дата	Преподаватель
юридич.	1	1	Авдеев	1	информатика	5	12.12.1994	Мухов
юридич.	1	1	Орлов	1	латынь	5	16.12.1994	Валитов
юридич.	2	1	Авдеев	3	угол.право	5	12.12.1995	Крот
юридич.	1	2	Мороз	1	информатика	5	12.12.1995	Мухов
эконом.	1	5	Леснов	1	делопр.	5	12.12.1995	Матюшков
эконом.	1	5	Леснов	1	математика	3	16.12.1995	Махист
эконом.	1	4	Василенко	2	экономика	5	10.06.1995	Братухин
эконом.	2	4	Мясникова	3	немецкий	4	23.12.1995	Афонин
эконом.	1	3	Ротайко	1	делопр.	5	12.12.1993	Саута
эконом.	3	3	Шуга	5	практика	4	22.12.1995	Хохлова

2. Для исходных данных о заказах, приведенных ниже в табличном виде, определить межтабличные зависимости и спроектировать схему данных (описать таблицы, поля таблиц, ключи, связи между таблицами):

Таблица "Клиенты"

Код клиента	Предприятие	Кредит
1	Белан	8 754 565 476 р.
2	СП СМУ «Брестторгстрой»	5 654 367 р.

Описание полей

Имя поля	Тип данных
Код клиента	Счетчик
Предприятие	Текстовый
Кредит	Денежный

Таблица "Заказы"

Код клиента	Код товара	Дата заказа	Заказано	Имя поля	Тип данных
1	1	02.09.97	3	Код клиента	Числовой
2	2	07.09.98	79	Код товара	Числовой
				Дата заказа	Дата
				Заказано	Числовой

Таблица "Товары"

Код товара	Наименование товара	Цена	Имя поля	Тип данных
1	Одеяло	6 700 р.	Код товара	Числовой
2	Подушка	8 000 р.	Наименование товара	Текстовый
			Цена	Денежный

3. Построить схемы данных РБД для заданных предметных областей (склад; гостиница; телеателье; библиотека; абитуриенты; кассы; приемное отделение больницы, поликлиники).

ПРИЛОЖЕНИЕ 2. Примерный перечень изучаемых вопросов

Тема 1. Информация в автоматизированных информационных системах (АИС) [3, 4]. Виды, особенности, структура. Классификация документов, форма и реквизиты документов. Классификация и кодирование информации.

Тема 2. Организация данных в базах данных. СУБД [3, 4]. Назначение и состав баз и банков данных. Эволюция систем обработки данных, независимость данных от программ, работа с данными по ключам. Модели данных (сетевые, иерархические, реляционные). Сущности, экземпляры сущностей, описательные и ключевые атрибуты (реквизиты). Особенности реляционных баз данных (РБД) [3, 4, 7], элементы (отношение, таблица, запись, поле-атрибут, ключ). Типы зависимостей атрибутов (функционально полная, транзитивная, частичная). Требования целостности. Функции и структура СУБД. Запросы и языки их описания. Типы и характеристики баз и БД. Системы на основе БД. Жизненный цикл и экономические аспекты БД.

Тема 3. Общая характеристика СУБД [1, 2, 8]. Возможности СУБД (на примере Access) по управлению одно и многопользовательскими РБД и взаимодействию с клиент-серверными БД. Интерфейсы доступа ODBC и OLE DB. Возможности по созданию приложений-клиентов баз данных SQL-сервера. Объекты (таблицы, формы, запросы, отчеты, страницы, макросы и модули). Интерфейс, инструменты (мастера и конструкторы), режим конструктора [1, с.13-39]. Типы данных. Свойства таблиц, полей, ключи [1, с.43-73]. Манипулирование данными в объектах, фильтры [1, с.197-205].

Тема 4. Формирование запросов. Введение в SQL [1, 2, 8]. Запросы по образцу QBE. Запросы на выборку. Задание условий отбора, группирование записей, вычисляемые поля и параметры. Запросы на модификацию БД. Запросы на основе запросов. Характеристика языка SQL, формирование запросов [1, с.207-245].

Тема 5. Проектирование пользовательского интерфейса баз данных [1]. Одно и многотабличные, одиночные и составные формы: подсхема, структура (заголовок, область данных и примечаний, элементы управления). Средства создания, панель элементов конструктора форм. Ограничения доступа к данным [1, с.122-194]. Отчеты: типы, под-схемы, структура (разделы и элементы разделов), средства создания и панель инструментов, просмотр и печать [1, с.257-267].

Тема 6. Элементы проектирования и создание баз данных (на примере Access) [1, с.41-77, 2 с.35-57, 7]. Предметная область, задачи, этапы проектирования. Техническое задание (описание задач, словарей данных, требований к характеристикам БД). Концептуальное проектирование (описание моделей сущностей, ER-модели). Логическое проектирование (нормализация таблиц). Физическое проектирование. Реализация РБД (описание таблиц, схем данных, параметров целостности, каскадного обновления и удаления записей).

Тема 7. Средства разработки прикладных программ [1, 2]. События. Языки макросов: формат макрокоманд, задание условий и формирование ссылок, средства конструирования [1, с.295-311]. Создание приложений (определение задач, конструирование объектов, алгоритмов, макросов, модулей, интерфейса) [1, с.291-295]. Возможности VBA [1, с.321-354]: классы, объекты, свойства, методы, модули, процедуры, команды. Редактор VBA. Графический интерфейс пользователя, диспетчер кнопочных форм [1, с.377-381].

Тема 8. Управление базами данных [1, с.399-448; 2]. Резервирование, оптимизация, восстановление. Разрешение конфликтов, блокировка страниц и записей. Распределенные базы данных. Репликация и синхронизация баз-реплик, конфликты. Публикация баз данных в Internet, конструирование Web-страниц.

ЛИТЕРАТУРА

1. Бекаревич Ю.Б., Пушкина Н.В. Самоучитель Microsoft ACCESS 2000. - СПб.: БХВ - Санкт-Петербург, 1999. - 480 с., ил.
2. Оскерко В.С. Практикум по технологиям баз данных: Учеб. пособие. - Мн.: БГЭУ, 2004. - 170 с.
3. Григорьев Ю.А., Ревунков Г.И. Базы данных: Учеб. для вузов. - М.: Изд-во МГТУ им. Н.Э.Баумана, 2002. - с. 320.
4. Малыгина М.П. Базы данных: основы, проектирование, использование: Учеб. пособие. - СПб.: БХВ - Петербург, 2004. - 512 с.
5. Технологии организации, хранения и обработки данных. Учебная программа для экономических специальностей. Морозевич А.Н., Оскерко В.С. - Минск, БГЭУ, 2000.
6. Microsoft ACCESS 97. Шаг за шагом: Практ. пособ./ Пер. с англ. - М.: Издательство ЭКОМ, 1999. - 328 с.: илл.
7. Дубнов П.Ю. Access 2000. Проектирование баз данных. - М.: ДМК, 2000. - 272 с.: ил.
8. Вейскас Дж. Эффективная работа с Microsoft ACCESS 2000. - СПб.: Издательство "Питер" - Санкт-Петербург, 2000. - 1040 с., ил.
9. Бекаревич Ю.Б., Пушкина Н.В. MS ACCESS 2000 за 30 занятий. - СПб.: БХВ - Санкт-Петербург, 2000. - 512 с., ил.
10. Михеева В., Харитоновна И. Microsoft ACCESS 2000: разработка приложений. - СПб.: БХВ - Санкт-Петербург, 1999. - 832 с., ил.
11. Дейт К. Дж. Введение в системы баз данных. - Киев-Москва: "Диалектика", 2001.
12. Уткин В.Б. Информационные системы в экономике: Учебник для студ. высш. учеб. заведений. - М.: Издательский центр "Академия", 2004. - 288 с.
13. Хансен Г., Хансен Дж. Базы данных: разработка и управление. - М.: БИНОМ, 1999. - 704 с.
14. Рынок информационных технологий в 2002-2003 годах. Шляхтина С., Прохоров А. Компьютер Пресс, N 1, 2003, с. 8-15.
15. Рынок информационных технологий в 2003-2004 годах. Прохоров А. Компьютер Пресс, N 1, 2004, с. 8-14.
16. Брич В.Г., Хвещук В.И., Горбашко Л.А., Муравьев Г.Л. Проектирование структур баз данных. Методические указания. - Брест: БРПИ, 1994. - 43 с.
17. Муравьев Г.Л. Работа в системе управления базами данных ACCESS (Часть 1). Типовые приемы создания и манипулирования объектами и данными. Методические указания к лабораторным работам. - Брест: БФ УО ИСЗ, 2002. - 73 с.

Учебное издание

Муравьев Геннадий Леонидович
Хвещук Владимир Иванович

**ОРГАНИЗАЦИЯ ОБРАБОТКИ ДАННЫХ СРЕДСТВАМИ СУБД.
Основные понятия теории баз данных**

Методическое пособие для студентов специальности 53 01 02
«Автоматизированные системы обработки информации»

Ответственный за выпуск: **Муравьев Г.Л.**

Редактор: **Строкач Т.В.**

Компьютерный набор и верстка: **Муравьев Г.Л., Кармаш Е.Л.**

Корректор: **Никитчик Е.В.**

Подписано в печать 02.10.2006 г. Формат 60x84^{1/16}.
Бумага писч. Усл. п.л. 1,16. Усл. изд. л. 1,25. Тираж 100 экз.

Заказ №944

Отпечатано на ризографе УО «Брестский государственный технический университет»
224017, Брест, ул. Московская, 267