

Проектирование баз данных

Быстренина Ирина Евгеньевна,
кандидат педагогических наук, доцент
кафедры прикладной информатики
РГАУ-МСХА имени К. А. Тимирязева



Задачи и основные этапы проектирования баз данных





Основные задачи проектирования базы данных:

- Обеспечение хранения в базе данных всей необходимой информации;
- Обеспечение возможности получения данных по всем необходимым запросам;
- Сокращение избыточности и дублирования данных;
- Обеспечение целостности базы данных.



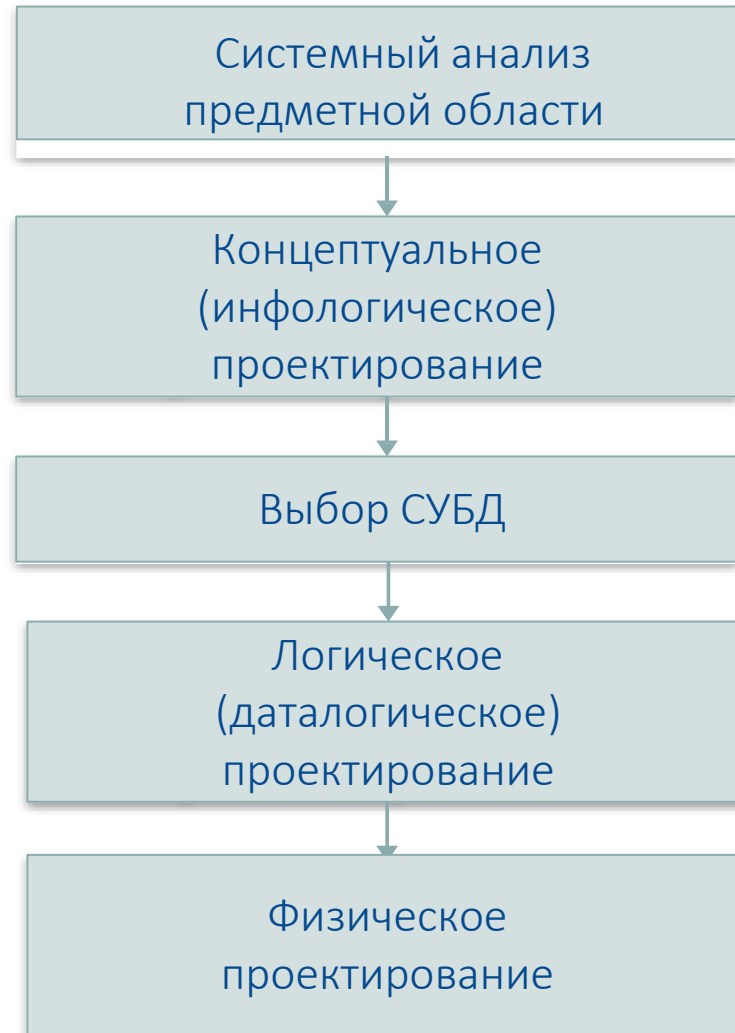
Этапы проектирования БД

Процесс проектирования БД включает:

- Системный анализ и словесное описание информационных объектов предметной области;
- Проектирование концептуальной модели предметной области;
- Выбор СУБД;
- Логическое проектирование;
- Физическое проектирование.



Этапы проектирования БД



Анализ предметной области





Изучение предметной области включает:

- Непосредственное наблюдение протекающих в ней процессов;
- Изучение документов, циркулирующих в системе;
- Интервьюирование участников этих процессов.



Предметная область – часть реального мира, данные о которой нужно отобразить в базе данных. Любая предметная область имеет свои границы рассмотрения и при проектировании базы данных, необходимо выделить информационные объекты внутри границ предметной области и абстрагироваться от информации вне границ предметной области. Например: для предметной области «Учет товаров на складе», понятие «накладная» является существенным, но количество детей сотрудника, принимающего накладные, является несущественным.



Модель предметной области — это формализованные знания о предметной области, выраженные при помощи каких-либо средств. В качестве таких средств могут выступать:

- **Текстовые описания предметной области** (должностные инструкции, описание документооборота, примеры первичных документов, выходных отчетов и т.п.);
- **Специализированные графические нотации** – более информативные и полезные при разработке баз данных.



Важным инструментом является использование единого языка представления проектных решений — языка моделирования.

Язык моделирования — это набор графических нотаций, которые используются для описания моделей в процессе проектирования.

Нотация представляет собой совокупность графических объектов, используемых в модели, и является синтаксисом языка моделирования.

Концептуальное моделирование





Концептуальное (инфологическое) проектирование – построение концептуальной модели предметной области. Такая модель создается без ориентации на какую-либо конкретную СУБД.

Термины «концептуальная модель», «семантическая модель» и «инфологическая модель» являются синонимами.

Концептуальная модель – это определенное множество понятий и связей между ними, являющихся смысловой структурой рассматриваемой предметной области.

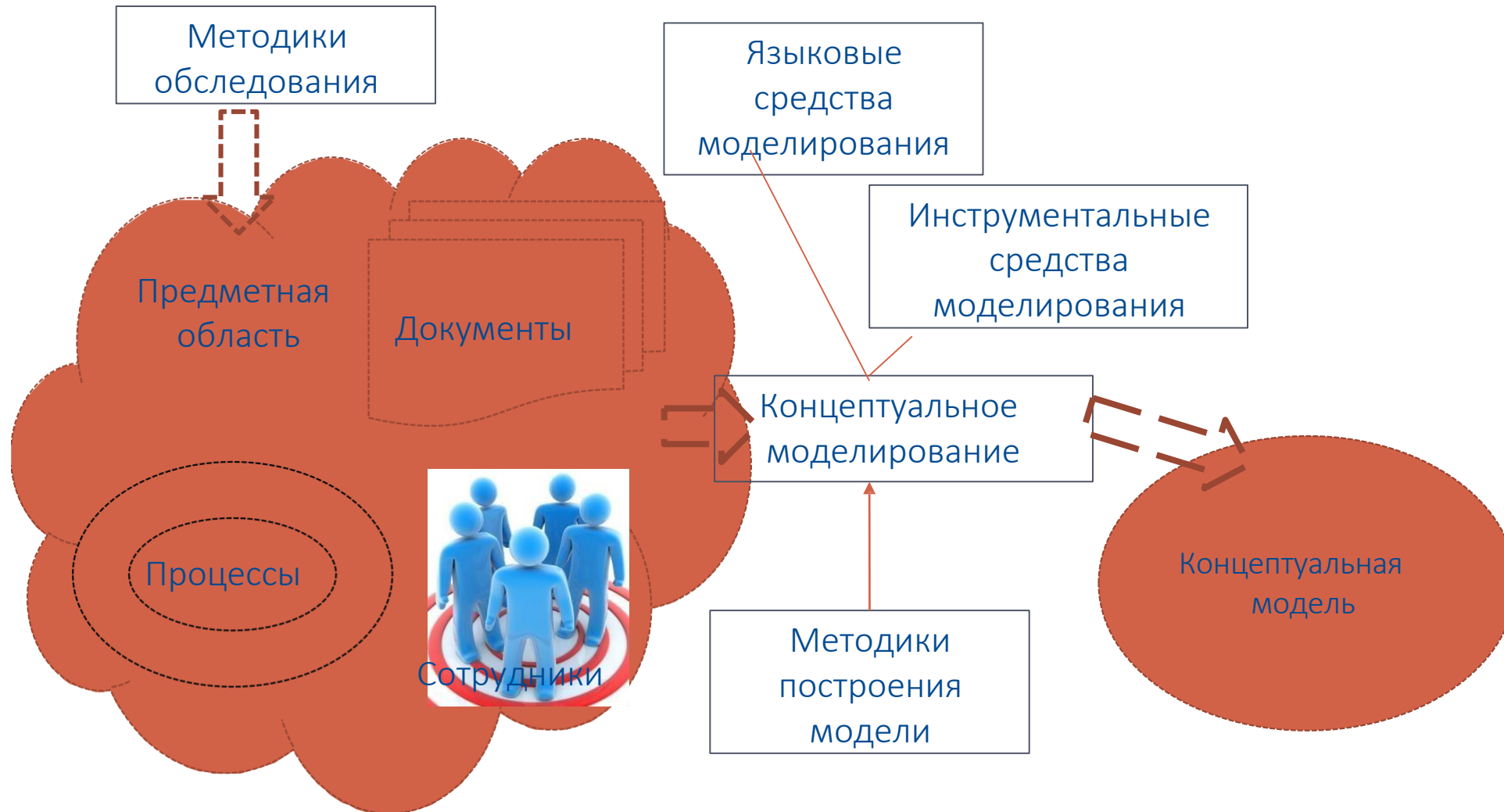


К концептуальной модели предметной области предъявляются следующие требования:

- **формализация**, обеспечивающая однозначное описание структуры предметной области;
- **понятность для заказчиков** и разработчиков на основе применения графических средств отображения модели;
- **реализуемость**, подразумевающая наличие средств физической реализации модели предметной области в ИС;
- **легкость модификации** (в модель по разным причинам часто приходится вводить новые объекты или модифицировать существующие; модель должна обеспечивать возможность ввода новых данных без изменения ранее определенных);
- **обеспечение возможности оценки эффективности реализации модели предметной области** на основе определенных методов и вычисляемых показателей.



Схема построения концептуальной модели





Концептуальная модель базы данных включает в себя:

- **описание информационных объектов**, или понятий предметной области и связей между ними;
- **описание ограничений целостности**, т.е. требований к допустимым значениям данных и к связям между ними.



В 1976 г. Питером Ченом была предложена модель «сущность-связь» (entity-relationship).

Она включает сущность и взаимосвязи, отражающие основные бизнес-правила предметной области.

Сокращенное название – **ER-модель**.

ER-модель – стандарт при концептуальном моделировании баз данных.



Семантическую основу ER-модели составляют следующие предположения.

- **часть реального мира**, т.е. совокупность взаимосвязанных объектов может быть представлена как совокупность сущностей;
- **сущности можно классифицировать по типам сущностей**: каждый экземпляр сущности (представляющий некоторый объект) может быть отнесен к определенному классу.



Класс — это совокупность объектов с одинаковым набором свойств, задаваемых в виде наборов атрибутов.

ER-диаграмма изображается с помощью трех конструктивных элементов: сущность, атрибут и связь.

Сущность — класс однотипных объектов, информация о которых имеет существенное значение для рассматриваемой предметной области. Каждая сущность должна иметь имя, выраженное существительным в единственном числе. Имя сущности должно быть уникальным в пределах модели.

Сущность представляет собой множество экземпляров реальных или абстрактных объектов (людей, событий, состояний, предметов и т.п.).



Экземпляр сущности — конкретный представитель данной сущности. Каждый экземпляр сущности должен быть отличим от любого другого экземпляра той же сущности (это требование аналогично требованию отсутствия кортежей-дубликатов в реляционных таблицах).

Каждая сущность обладает одним или несколькими атрибутами.

Атрибут сущности — это именованная характеристика, являющаяся некоторым свойством сущности. Наименование атрибута должно быть выражено существительным в единственном числе (возможно с характеризующими прилагательными).



Виды атрибутов

- Множественные или единичные;
- Простые или составные;
- Постоянные (статическими) или динамические, например, свойство «ИНН» является статическим, а «Адрес» — динамическим.



Первичный ключ сущности – не избыточный набор атрибутов, значение которых в совокупности является уникальными для каждого экземпляра сущности.

Атрибуты делятся на **ключевые** и **описательные**.

- **Ключевые атрибуты** — атрибуты, входящие в состав первичного ключа. Ключевые атрибуты помещают в начало списка и помечают символом #.
- Прочие атрибуты называются **описательными**. Описательные атрибуты могут быть обязательными или необязательными. Обязательные атрибуты для каждой сущности всегда имеют конкретное значение, необязательные — могут быть не определены.



Связь — отношение одной сущности к другой или к самой себе.

Обязательная связь — если любой экземпляр одной сущности связан хотя бы с одним экземпляром другой сущности.

Необязательная связь — экземпляр одной сущности может быть связан с одним или несколькими экземплярами другой сущности, а может быть и не связан ни с одним экземпляром.

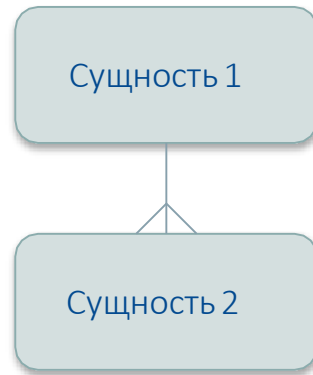




Обозначения разных типов связи в нотации Баркера



ОДИН-К-ОДНОМУ



ОДИН-КО-МНОГИМ



МНОГИЕ-КО-МНОГИМ

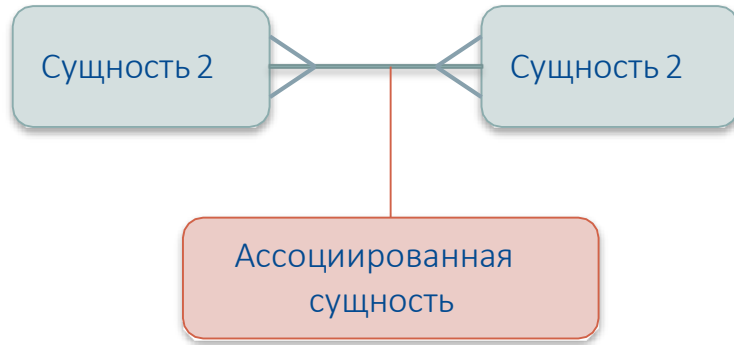


Все сущности можно подразделить на 3 вида:

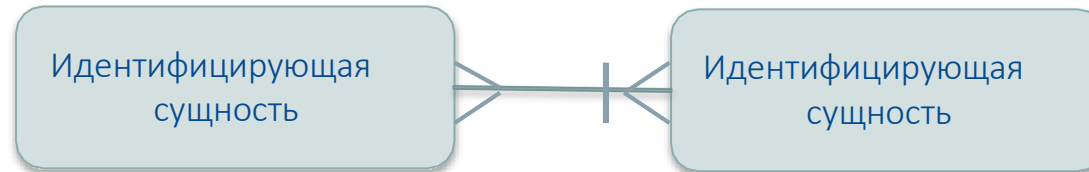
Независимая сущность представляет независимые данные, которые всегда присутствуют в системе. Они могут быть как связаны с другими сущностями, так и нет.

Зависимая сущность представляет данные, зависящие от других сущностей системы, поэтому она всегда должна быть связана с другими сущностями.

Ассоциированная сущность представляет данные, которые ассоциируются с отношениями между двумя и более сущностями. Обычно данный вид сущностей появляется в модели для разрешения отношения «многие-ко-многим».



Обозначение ассоциированной сущности в нотации Баркера



Обозначение идентификации посредством другой сущности в нотации Баркера



Пример ER-модели БД «Учет поставки и продажи товаров»





Построение концептуальной модели может выполняться как «вручную», так и с использованием автоматизированных средств проектирования - CASE-средств.

Логическое проектирование и физическая модель базы данных





Критерии выбора СУБД

- тип модели данных, которую поддерживает данная СУБД, адекватность модели данных структуре рассматриваемого ПО;
- характеристики производительности СУБД;
- запас функциональных возможностей для дальнейшего развития информационной системы;
- степень оснащённости СУБД инструментарием для персонала администрирования данными;
- удобство и надёжность СУБД в эксплуатации;
- стоимость СУБД и дополнительного программного обеспечения.



Логическое проектирование

Логическое (дatalogическое) проектирование – создание схемы базы данных на основе конкретной модели данных (например реляционной модели данных).



Логическое проектирование

На этапе логического проектирования учитывается специфика конкретной модели данных, но может не учитываться специфика конкретной СУБД.



Логическое проектирование

Спроектировать логическую структуру базы данных означает:

- Определить все информационные единицы и связи между ними, задать их имена;
- Определить их тип, задать некоторые количественные характеристики (например, длину поля).



Логическое проектирование

В концептуальной модели должна быть отображена вся информация, циркулирующая в информационной системе. Прежде чем строить логическую модель, необходимо определить состав БД, т. е. перечень показателей, которые целесообразно хранить в базе данных.



Логическое проектирование

Также не все виды связей, существующие в предметной области, могут быть непосредственно отображены в конкретной логической модели. Так, многие СУБД не поддерживают непосредственно отношение «многие-ко-многим» между элементами. В этом случае в логическую модель вводится дополнительный вспомогательный элемент, отображающий эту связь (отношение «многие-ко-многим» условно разбивается на два отношения «один-ко-многим» между этим вновь введенным элементом и исходными элементами).

Типовая пошаговая процедура преобразования диаграммы «сущность-связь» в реляционную схему базы данных



- Каждая сущность превращается в таблицу. Имя сущности становится именем таблицы. Экземплярам типа сущности соответствуют строки соответствующей таблицы.
- Каждый атрибут становится столбцом таблицы с тем же именем; может выбираться более точный формат представления данных. Столбцы, соответствующие необязательным атрибутам, могут содержать неопределенные значения; столбцы, соответствующие обязательным атрибутам, - не могут.

Типовая процедура преобразования диаграммы «сущность-связь» в реляционную схему базы данных



Компоненты уникального идентификатора сущности превращается в первичный ключ таблицы. Если имеется несколько возможных уникальных идентификаторов, для первичного ключа выбирается наиболее характерный. Если в состав уникального идентификатора входят связи, к числу столбцов первичного ключа добавляется копия уникального идентификатора сущности, находящийся на другом конце связи. Для именованя этих столбцов используются имена концов связей и/или имена парных сущностей.

Типовая пошаговая процедура преобразования диаграммы «сущность-связь» в реляционную схему базы данных



Связи «один к одному» становятся внешними ключами. Необязательные связи соответствуют столбцам внешнего ключа, допускающим наличие неопределенных значений; обязательные связи — столбцам, не допускающим неопределенных значений. Если между двумя сущности А и В имеется связь «один к одному», то соответствующий внешний ключ по желанию проектировщика может быть объявлен как в таблице А, так и в таблице В.

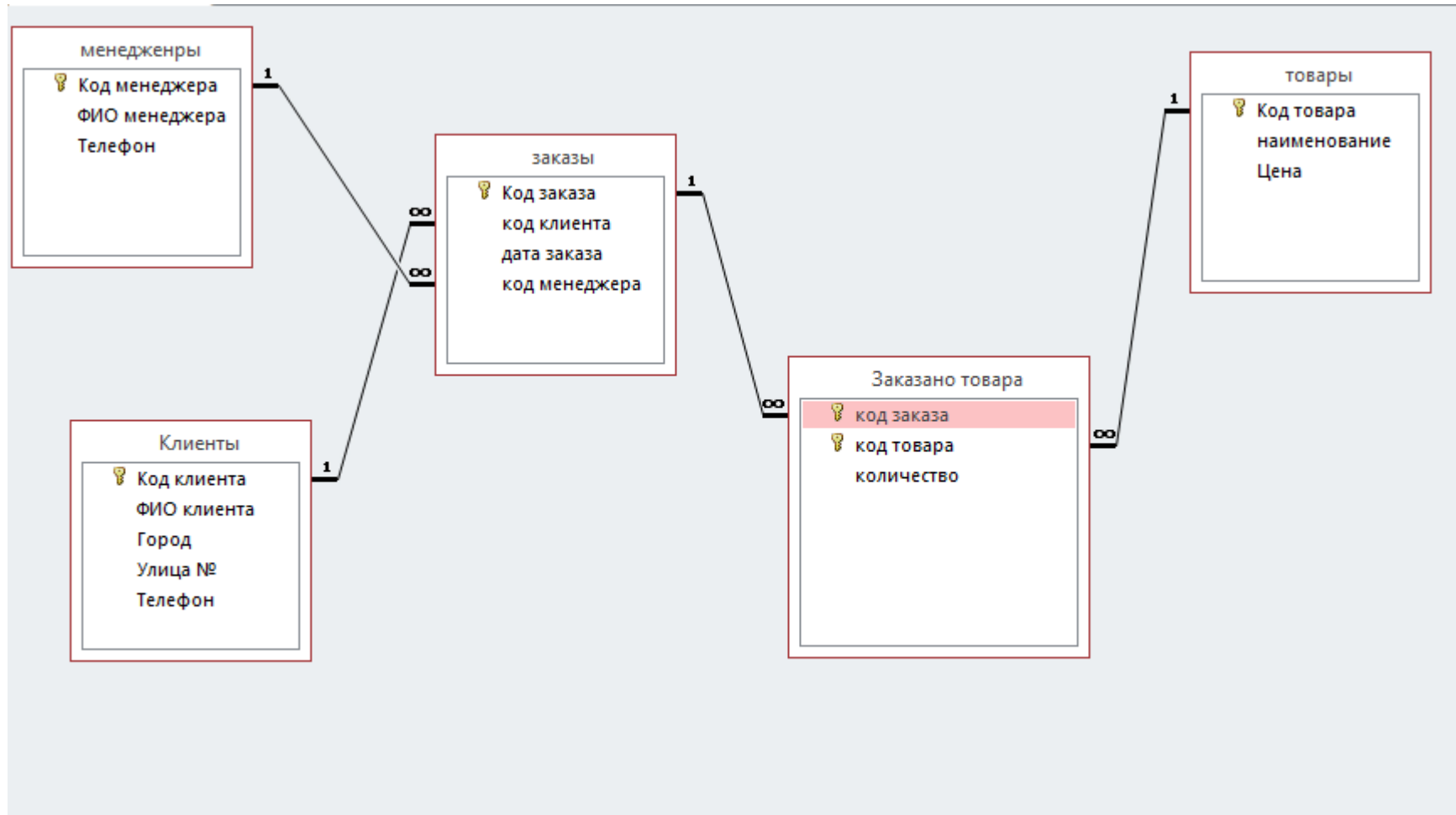
Типовая пошаговая процедура преобразования диаграммы «сущность-связь» в реляционную схему базы данных



Для поддержки связи «многие ко многим» между типами сущности А и В создается дополнительная таблица С с двумя обязательными столбцами, один из которых содержит уникальные идентификаторы экземпляров сущности А, а другой – уникальные идентификаторы экземпляров сущности В.



Пример логической модели данных в СУБД Access





Физическое проектирование

Для привязки логической модели данных к среде хранения используется модель данных физического уровня – физическая модель – определяем способы физической организации данных в среде хранения. Физическая модель строится с учетом возможностей, предоставляемых СУБД. Описание физической структуры базы данных называется схемой хранения.



Физическое проектирование

Физическое проектирование — создание схемы базы данных для конкретной системы управления базами данных.

На этапе физического проектирования учитывается специфика конкретной модели данных и специфика конкретной СУБД.

Отношения, разработанные в логической модели, преобразуются в таблицы; атрибуты становятся столбцами таблиц, для ключевых атрибутов создаются уникальные индексы, домены преобразуются в типы данных, принятые в используемой СУБД.



Физическое проектирование

Физическое проектирование БД предполагает:

- выбор эффективного размещения БД на внешних носителях для обеспечения наиболее эффективной работы приложения;
- выбор вычислительных ресурсов, необходимых для функционирования системы (типа и конфигурации ЭВМ, операционной системы), который зависит от следующих показателей:



Физическое проектирование

- Примерный объем данных БД;
- Динамика роста объема данных;
- Характер запросов к данным (извлечение и обновление отдельных записей, обработка групп записей, обработка отдельных отношений, соединение отношений);
- Интенсивность запросов к данным по типам запросов;
- Требования к времени отклика системы по типам запросов.

Основные этапы и задачи процесса проектирования базы данных



Уровень	Основные задачи	Основные шаги	Базовые понятия
Концептуальный	Сбор, анализ и редактирование требований к данным	Обследование предметной области, изучение ее информационной структуры. Выявление всех фрагментов предметной области, каждый из которых характеризуется пользовательским представлением, информационными объектами и связями между ними, процессами над информационными объектами. По окончании данного этапа получаем концептуальную модель — ER-диаграмму, инвариантную к структуре базы	Сущности. Атрибуты. Связи

Основные этапы и задачи процесса проектирования базы данных



Уровень	Основные задачи	Основные шаги	Базовые понятия
Логический	Преобразование требований к данным в структуры данных	ER-диаграмма преобразуется в набор таблиц, при этом проводится их нормализация. На выходе получаем СУБД ориентированную структуру базы данных и спецификации прикладных программ. На этом этапе часто моделируют базы данных применительно к различным СУБД и проводят сравнительный анализ моделей	Таблицы. Записи. Элементы данных. Связи между записями

Основные этапы и задачи процесса проектирования базы данных



Уровень	Основные задачи	Основные шаги	Базовые понятия
Физический	Определение особенностей хранения данных, методов доступа к данным и пр.	Адаптация логической модели к выбранной программной платформе. Выбор и построение индексов. Организация средств протоколирования и пр.	Группировка данных. Индексы. Методы доступа

Проектирование баз данных на основе принципов нормализации



Аномалии

- **Аномалии** – это проблемы, возникающие в данных из-за дефектов проектирования БД.
- **Аномалия обновления** – появление в базе данных несогласованности данных при выполнении операций вставки, удаления, модификации записей.
- **Аномалии вставки (ввода)** – добавление лишней информации или возникновение противоречащих значений в некоторых столбцах при вставке новой записи.



Аномалии вставки (ввода)

Номер зачетной книжки		Номер группы
203911	Песочников В.Д.	852
203912	Баранова П.Т.	851
230910	Песочникова В.Д.	851
234576	Скuryякова А.П.	852
294950	Григорьев Б.Н.	853
322343	Галактионов А.В.	852
394052	Рухлов В.П.	851
402622	Мамаева Г.А.	853
403291	Проклова Е.П.	852
404657	Водопьянов Г.И.	853
405090	Олегов И.И.	852
409221	Васильева А.Д.	851
431290	Цыпкин В.М.	242
431301	Ершов В.А.	242

Добавление названия новой группы повлечет обязательное определение Фамилии и Номера зачетной книжки, в то время как эти данные могут быть пока не известны.



Аномалии удаления

Аномалии удаления – удаление лишней информации при удалении записи.

Номер зачетной книжки	Фамилия	Номер группы
203911	Песочников В.Д.	852
203912	Баранова П.Т.	851
230910	Песочникова В.Д.	851
245576	Скuryякова А.П.	852
294950	Григорьев Б.Н.	853
322343	Галактионов А.В.	852
394052	Рухлов В.П.	851

Удаление студента или нескольких студентов может привести к удалению из БД и номера группы.



Аномалии модификации

Аномалии модификации – появление записей с противоречащими значениями в некоторых столбцах при изменении значений соответствующих полей одной записи.

Номер зачетной книжки	Фамилия	Номер группы
203911	Песочников В.Д.	852
203912	Баранова П.Т.	851
230910	Песочникова В.Д.	851
245576	Скuryякова А.П.	852
294950	Григорьев Б.Н.	853
322343	Галактионов А.В.	852
394052	Рухлов В.П.	851

При изменении номера группы (например, при переходе на следующий курс) его нужно изменить для всех студентов, обучающихся в этой группе. Если для какого-либо студента этого не сделать, то возникнет несоответствие данных.



Избыточность данных

Под **избыточностью данных** понимают дублирование данных, содержащихся в базах данных. При этом различают простое (неизбыточное) дублирование и избыточное дублирование.

Неизбыточное дублирование является естественным и допустимым, возникает из необходимости хранить идентичные данные, поскольку важен сам факт их идентичности, и удаления хотя бы одного представителя идентичных данных приведет к невозполнимой потере информации.



Избыточность данных

Номер зачетной книжки	Фамилия	Номер группы
203911	Песочников В.Д.	852
203912	Баранова П.Т.	851
230910	Песочникова В.Д.	851
245576	Скuryякова А.П.	852
294950	Григорьев Б.Н.	853
322343	Галактионов А.В.	852
394052	Рухлов В.П.	851

Пример избыточного дублирования – дублирование значений атрибута Номер группы. При удалении одного из повторяющихся значений группы будет потеряна информация о принадлежности одного студента к данной группе.



Избыточность данных

Избыточное дублирование (избыточность) обычно связано с необходимостью задания значения всех атрибутов отношения, при этом дублируемые данные не являются необходимыми, и в случае потери (удаления) могут быть восстановлены по данным одного или нескольких отношений БД.



Избыточность данных

Номер зачетной книжки	Фамилия	Номер группы	Куратор
203911	Песочников В.Д.	852	Ольхов Е.В.
203912	Баранова П.Т.	851	Поспелова А.Г.
230910	Песочникова В.Д.	851	Поспелова А.Г.
245576	Скuryякова А.П.	852	Ольхов Е.В.
294950	Григорьев Б.Н.	853	Горшев Н.С.
322343	Галактионов А.В.	852	Ольхов Е.В.
394052	Рухлов В.П.	851	Поспелова А.Г.
402622	Мамаева Г.А.	853	Горшев Н.С.

Пример избыточного дублирования – дублирование значений атрибута Куратор. Для каждого студента некоторой группы значение атрибута Куратор может быть восстановлено по значениям данного атрибута других студентов рассматриваемой группы.



Декомпозиция таблицы

Декомпозиция (разбиение) таблицы — процесс деления таблицы на несколько таблиц для поддержания целостности данных. т.е. устранения избыточности данных и аномалий.

Номер зачетной книжки	Фамилия	Номер группы
203911	Песочников В.Д.	852
203912	Баранова П.Т.	851
230910	Песочникова В.Д.	851
245576	Скурякова А.П.	852
294950	Григорьев Б.Н.	853
322343	Галактионов А.В.	852
394052	Рухлов В.П.	851
431290	Цыпкин В.М.	242
431301	Ершов В.А.	242

Номер группы	Куратор
852	Ольхов Е.В.
851	Поспелова А.Г.
853	Горшев Н.С.
242	Соловьев А.А.

Таблицы связаны между собой по номеру группы. Для получения информации о кураторе студента из первой таблицы считывают номер его группы, после чего из второй таблицы по номеру группы определяют фамилию куратора.



Нормализация

Нормализация таблиц базы данных – это процесс организации данных в базе данных, включающий создание таблиц и установление отношений между ними в соответствии с правилами, которые обеспечивают защиту данных и делают базу данных более гибкой, устраняя избыточность и несогласованные зависимости.

Главная цель нормализации базы данных – это устранение (или, по крайней мере, серьезное сокращение) избыточности, дублирования данных. Как следствие, значительно сокращается вероятность появления противоречивых данных, облегчается администрирование базы и обновление информации в ней, сокращается объем дискового пространства.



Нормализация

Процесс проектирования базы данных с использованием метода нормальных форм является пошаговым и заключается в последовательном переводе по определенным правилам отношений из первой нормальной формы в нормальные формы более высокого уровня.

В теории реляционных баз данных обычно выделяется следующая последовательность нормальных форм:

- первая нормальная форма (1 нф, 1 Normal Form, 1NF);
- вторая нормальная форма (2 нф, 2 NF);
- третья нормальная форма (3 нф, 3 NF);
- нормальная форма Бойса—Кодда (BCNF);
- четвертая нормальная форма (4 нф, 4 NF);
- пятая нормальная форма, или нормальная форма проекции- соединения (5 нф, 5 NF, или PJ/NF).



Нормализация

Проектирование начинается с определения всех объектов, информация о которых должна содержаться в базе данных, определения атрибутов этих объектов. Атрибуты всех объектов сводятся в одну исходную таблицу. Эта таблица приводится к первой нормальной форме в соответствии с ее требованиями. Затем эта таблица декомпозируется на две или несколько таблиц, те в свою очередь тоже могут быть преобразованы в другие таблицы. Так последовательно создается совокупность взаимосвязанных таблиц, отвечающих требованиям нормальных форм. Обычно на практике применение находят только первые три нормальные формы.



Нормализация

Каждой нормальной форме соответствует некоторый определенный набор ограничений. Отношение находится в нормальной форме, если удовлетворяет свойственному ей набору ограничений.

Первая нормальная форма

- поля таблицы должны содержать неделимую (атомарную) информацию.

Вторая нормальная форма

- таблица должна удовлетворять требованиям первой нормальной формы;
- любое неключевое поле должно однозначно идентифицироваться ключевыми полями.

Третья нормальная форма

- Таблица должна удовлетворять требованиям второй нормальной формы;
- Неключевые поля не зависят друг от друга.



Разработка базы данных для хранения сведений о студентах

ФИО	Дата рождения	Специальность	Шифр	Группа, Куратор
Песочников В.Д.	15.11.1980	Информационные системы	09.02.04	852 Ольхов Е.В.
Баранова П.Т.	20.06.1981	Компьютерные сети	09.02.02	851 Поспелова А.Г.
Песочникова В.Д.	12.10.1981	Компьютерные сети	09.02.02	851 Поспелова А.Г.
Скuryкова А.П.	10.02.1980	Информационные системы	09.02.04	852 Ольхов Е.В.
Григорьев Б.Н.	02.08.1980	Информационные системы	09.02.04	853 Горшев Н.С.
Галактионов А.В.	09.10.1981	Информационные системы	09.02.04	852 Ольхов Е.В.
Рухлов В.П.	18.02.1982	Компьютерные сети	09.02.02	851 Поспелова А.Г.
Мамаева Г.А.	30.05.1982	Информационные системы	09.02.04	853 Горшев Н.С.
Проклова Е.П.	28.09.1981	Информационные системы	09.02.04	852 Ольхов Е.В.



Первая нормальная форма

ФИО	Дата рождения	Специальность	Шифр	Группа	Группа, Куратор
Песочников В.Д.	15.11.1980	Информационные системы	09.02.04	852	Ольхов Е.В.
Баранова П.Т.	20.06.1981	Компьютерные сети	09.02.02	851	Поспелова А.Г.
Песочникова В.Д.	12.10.1981	Компьютерные сети	09.02.02	851	Поспелова А.Г.
Скuryакова А.П.	10.02.1980	Информационные системы	09.02.04	852	Ольхов Е.В.
Григорьев Б.Н.	02.08.1980	Информационные системы	09.02.04	853	Горшев Н.С.
Галактионов А.В.	09.10.1981	Информационные системы	09.02.04	852	Ольхов Е.В.
Рухлов В.П.	18.02.1982	Компьютерные сети	09.02.02	851	Поспелова А.Г.
Мамаева Г.А.	30.05.1982	Информационные системы	09.02.04	853	Горшев Н.С.
Проклова Е.П.	28.09.1981	Информационные системы	09.02.04	852	Ольхов Е.В.



Вторая нормальная форма

Номер зачетной книжки	ФИО	Дата рождения	Специальность	Шифр	Группа	Группа, Куратор
322343	Песочников В.Д.	15.11.1980	Информационные системы	09.02.04	852	Ольхов Е.В.
394052	Баранова П.Т.	20.06.1981	Компьютерные сети	09.02.02	851	Поспелова А.Г.
402622	Песочникова В.Д.	12.10.1981	Компьютерные сети	09.02.02	851	Поспелова А.Г.
403291	Скuryякова А.П.	10.02.1980	Информационные системы	09.02.04	852	Ольхов Е.В.
404657	Григорьев Б.Н.	02.08.1980	Информационные системы	09.02.04	853	Горшев Н.С.
405090	Галактионов А.В.	09.10.1981	Информационные системы	09.02.04	852	Ольхов Е.В.
409221	Рухлов В.П.	18.02.1982	Компьютерные сети	09.02.02	851	Поспелова А.Г.
431290	Мамаева Г.А.	30.05.1982	Информационные системы	09.02.04	853	Горшев Н.С.
431301	Проклова Е.П.	28.09.1981	Информационные системы	09.02.04	852	Ольхов Е.В.



Третья нормальная форма

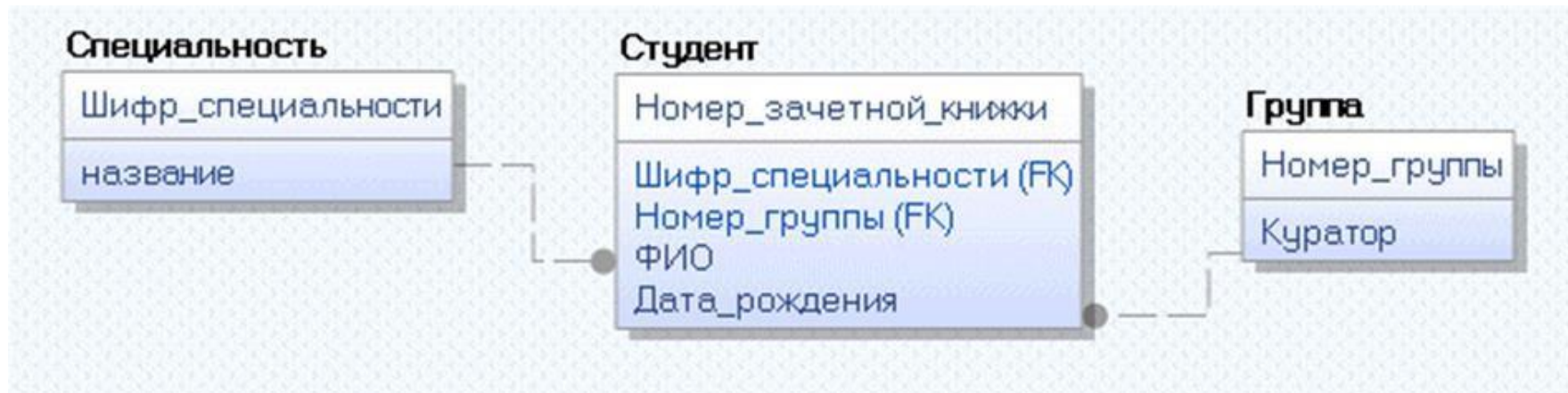


Схема базы данных



Третья нормальная форма

Таблица Студент

Номер зачетной книжки	ФИО	Дата рождения	Специальность	Шифр специальности	Номер группы
203911	Песочников В.Д.	15.11.1980	Информационные системы	09.02.04	852
203912	Баранова П.Т.	20.06.1981	Компьютерные сети	09.02.02	851
230910	Песочникова В.Д.	12.10.1981	Компьютерные сети	09.02.02	851
234576	Скурякова А.П.	10.02.1980	Информационные системы	09.02.04	852
294950	Григорьев Б.Н.	02.08.1980	Информационные системы	09.02.04	853
322343	Галактионов А.В.	09.10.1981	Информационные системы	09.02.04	852
394052	Рухлов В.П.	18.02.1982	Компьютерные сети	09.02.02	851
402622	Мамаева Г.А.	30.05.1982	Информационные системы	09.02.04	853
403291	Проклова Е.П.	28.09.1981	Информационные системы	09.02.04	852
404657	Водопьянов Г.И.	10.02.1982	Информационные системы	09.02.02	853
405090	Олегов И.И.	20.10.1982	Информационные системы	09.02.02	852
409221	Васильева А.Д.	23.08.1982	Компьютерные сети	09.02.02	851
431290	Цыпкин В.М.	24.07.1981	Коммерция	38.02.04	242
431301	Курпатов И.В.	10.02.1982	Коммерция	38.02.04	242

Таблица Специальность

Шифр специальности	Название
09.02.04	Информационные системы
09.02.02	Компьютерные сети
38.02.04	Коммерция

Таблица Группа

851	Поспелова А.Г.
852	Ольхов Е.В.
852	Горшев Н.С.
242	Соловьев А.А.



Спасибо за внимание!