

Глава 1

1. Укажите отличительные особенности технологии БД, выделяющие ее среди технологий вообще и информационных технологий, в частности.

Технология баз данных – это технология, регламентирующая процесс создания и эксплуатации информационных систем, в которых для долговременного хранения информации используется база данных, а управляет ею некоторая система управления базами данных (СУБД).

Информационную технологию баз данных отличает использование СУБД той или иной модели данных (МД) – концепции представления в ней объектов мира и их взаимосвязей. Развитые модели, обеспечивающие адекватное представление данных, относящихся к различным предметным областям, в конечном итоге определяют степень соответствия СУБД целям их создания.

2. Каковы предпосылки использования технологии БД?

1. **Долговременное хранение информации** о предметной области
2. **Для удовлетворения информационных потребностей нескольких людей**, в том числе тех, кто в силу разных причин не в состоянии воспринимать этот мир с помощью своих органов чувств.
3. **Хорошо структурированное хранилище**, в котором каждый бит информации лежит на своей полке и непротиворечиво связан с остальными.
4. **Пользователи должны вносить изменения** в его информационное описание.
5. Большая часть пользователей будет удовлетворять свои потребности в информации об этом мире, **обращаясь к информационной системе, в которой должны обеспечиваться специальные механизмы, способствующие удобному извлечению и преобразованию информации.**
6. **Производная информация должна получаться из первичной, введенной непосредственно людьми, с помощью простых универсальных преобразований.**
7. **Информация вводится, хранится и предъявляется преимущественно в алфавитно-цифровой форме.**

3. В чем особенности термина модель в технологии БД?

Информационную технологию баз данных отличает использование СУБД той или иной модели данных (МД) – концепции представления в ней объектов мира и их взаимосвязей. **Термин модель в технологии БД обозначает теорию моделирования, а результат моделирования есть не что иное, как БД.**

4. Каково основное назначение моделей данных?

Основное назначение МД – обеспечение процесса интерпретации данных и превращения их в информацию.

5. Перечислите составные части любой модели данных.

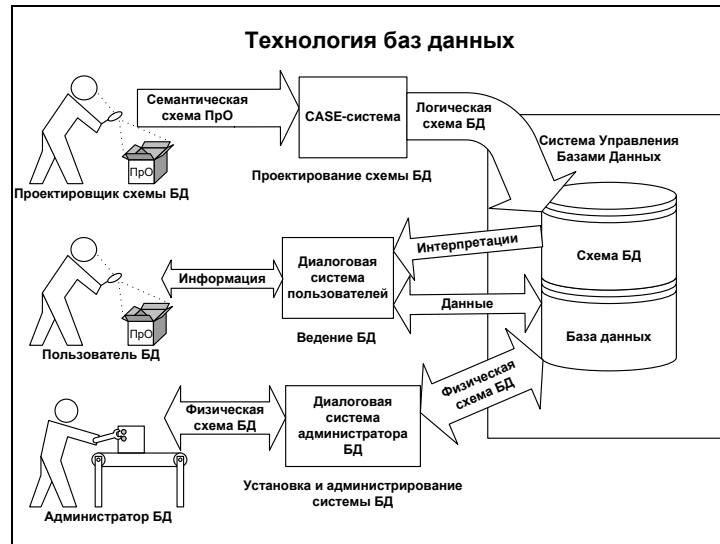
В каждой модели данных имеются следующие компоненты:

- правила порождения допустимых структур данных;
- правила порождения ограничений целостности данных;
- набор операций над данными.

6. Каково назначение структур данных, ограничений целостности и операций над данными?

На основании правил определяется схема БД (основные понятия предметной области и их взаимосвязи), а операции необходимы, чтобы первоначально наполнить схему непротиворечивыми данными и поддерживать их в дальнейшем в адекватном предметной области состоянии, а также обеспечивать получение необходимой информации.

7. Укажите основные процессы в системах БД, кто и какие задачи в них реализует.



В реальной системе возможны ситуации, когда одну и ту же роль исполняет несколько человек, образуя соответствующую функциональную группу (например, отдел администрирования БД). С другой стороны, в небольшой автономной системе для личных нужд все роли играет один человек.

Самый первый, важный процесс – проектирование схемы БД. От того, насколько качественной будет схема БД, существенно зависят простота и легкость осуществления всех остальных процессов. Проектировщик схемы БД изучает самостоятельно и с помощью экспертов ПрО, и формализует полученные знания в виде семантической схемы ПрО. Эта схема строится с использованием той или иной семантической модели данных, концепция которой максимально близка человеческому восприятию мира.

При наличии специализированного программного обеспечения в лице CASE-системы можно, во-первых, просто и естественно для проектировщика осуществить ввод семантической схемы, а, во-вторых, автоматически перевести ее на язык СУБД.

При отсутствии подходящего CASE-инструмента необходимую трансляцию схемы ПрО проектировщик осуществляет вручную, на бумаге. Для этого он должен знать методику этого перевода – набор правил преобразования структур и ограничений целостности из семантической модели в логическую модель СУБД.

После разработки схемы БД и диалоговой системы пользователей администратор системы осуществляет ее установку на оборудовании организации-заказчика. Он администрирует систему на уровне физической схемы БД – схемы хранения данных и их интерпретаций на диске. Основной обязанностью администратора БД является обеспечение эффективной бесперебойной работы БД, не взирая на возможный выход из строя аппаратуры и программ.

Применение пользователями. Пользователи могут выступать в двух ролях – писателей и читателей. Первые обязаны отражать в БД изменения ПрО. Вторые удовлетворяют свои информационные потребности, обращаясь к системе БД.

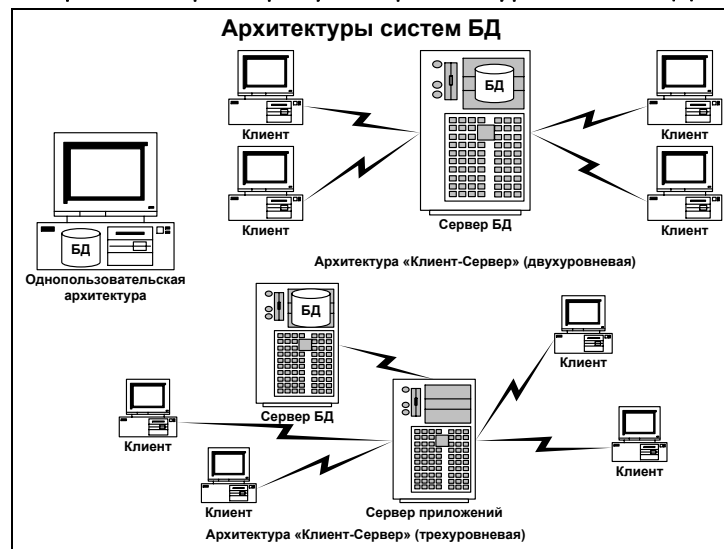
8. Для чего предназначены CASE-системы?

Правильный выбор подходящей семантической модели может существенно облегчить задачу проектировщика.

-поддержка модели CASE-средством (CASE – Computer Aided Software/System Engineering – **программная система для проектирования и реализации других программных систем**), обеспечивающим автоматическую трансляцию схемы ПрО на язык СУБД-ориентированной модели данных или даже на язык логической модели конкретной СУБД.

При наличии специализированного программного обеспечения в лице CASE-системы можно, во-первых, просто и естественно для проектировщика осуществить ввод семантической схемы, а, во-вторых, автоматически перевести ее на язык СУБД.

9. Перечислите и кратко охарактеризуйте архитектуры систем БД.



В однопользовательской архитектуре (автономная) вся система БД управляется одним компьютером, на котором расположены и БД, и СУБД, и диалоговая система пользователя.

В клиент-серверной архитектуре БД (двухуровневая) система состоит из множества компьютеров, объединенных в сеть. Одни компьютеры, называемые клиентами, занимаются обработкой прикладных программ, которые ведут диалоги с пользователями. Другие компьютеры, называемые серверами, занимаются обработкой БД.

Трехуровневая архитектура, включающая сервер БД, web-сервер и браузер. Каждый из этих уровней может работать под управлением своей операционной системы на отдельном компьютере. Так же как и двухуровневую, трехуровневую архитектуру программных продуктов технологии БД можно реализовать и на одном компьютере. Через интерфейс между web-сервером и сервером БД передаются SQL-операторы и реляционные данные. Через интерфейс между web-сервером и браузером передаются web-страницы, клиентский код и данные.

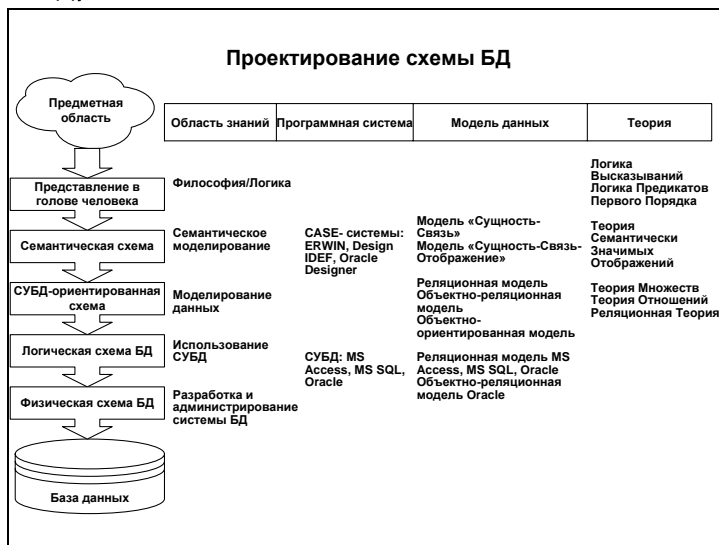
10. В чем основные различия OLTP- и OLAP-систем БД?

Транзакция – логически неделимая единица работы с данными (обычно включающая несколько операций над БД), которая должна выполняться атомарным образом.

OLTP (система оперативной обработки транзакций) – система управления больших потоков транзакций, вносящих незначительные изменения в БД. Размер БД для OLTP-систем может варьироваться от совсем малого, всего в несколько мегабайт, до среднего, порядка нескольких гигабайт, и дальше вплоть до очень большого, на уровне нескольких терабайт или даже петабайт.

OLAP (инструменты оперативной аналитической обработки) – инструменты для анализа огромного объема данных, представленных через многомерные кубы.

11. Какие формы представлений о ПрО выделяют в технологии БД? Охарактеризуйте каждую из них.



Первоначальные представления о ПрО возникают **в голове у проектировщика БД** на основании его личного знакомства с ней. Эти исключительно неформализованные представления составляют зрительные, слуховые, обонятельные, осязательные образы. Его задачей является перевод своего неформального восприятия ПрО на язык конкретной модели данных, используя ее формальный понятийный набор.

В результате получается **семантическая схема** ПрО – ее первое формальное представление.

Модели данных конкретных СУБД, хотя и отличаются друг от друга, часто имеют много общих особенностей, которые образуют так называемые СУБД-ориентированные модели данных. В частности, к ним относится самая популярная в настоящее время реляционная модель. Если в качестве СУБД вы выбрали одну из реляционных СУБД, вам предстоит построить реляционную схему своей ПрО, которая является одной из разновидностей СУБД-ориентированных схем.

Две модели данных – **логическую и физическую**. Первая из них обращена в сторону человека (проектировщика или пользователя БД) и предлагает языковые и диалоговые инструменты общения именно для него. Вторую модель СУБД (физическую) редко кто, кроме разработчиков этой СУБД, и знает, поскольку обращена она к средствам хранения данных в оперативной и внешней памяти.

Глава 2.1

1. Объясните, почему человека интересуют не данные, а информация.

Определение 2.1.1. Данные - это факты реального мира и идеи, представленные в формализованном виде, позволяющем передавать или обрабатывать их при помощи некоторого процесса и соответствующих технических средств.

Определение 2.1.2. Информация – приращение знаний человека, которое может быть получено на основе данных.

Сами по себе данные ничего для человека не значат, но на их основе человек может получить информацию, которая его интересует.

Пример данных: 120 и 30

Пример информации: Некий человек имеет значение характеристики Рост, равное 120, и значение характеристики Возраст, равное 30.

2. Что кроме данных необходимо для получения информации?

Интерпретация и знание ПрО

Данные без интерпретаций мертвы, необходимы дополнительные сведения. Пример: такие данные, как 120 и 30 ничего не говорят человеку. Его интересует информация; Если эти данные снабдить интерпретацией первое число – рост человека в сантиметрах, второе – возраст этого же человека в годах, можно образовать содержание (смысл) сообщения в целом.

3. Перечислите и охарактеризуйте три этапа процесса образования информации из данных.

Определение 2.1.3. Семиотика – комплекс научных учений, изучающих свойства семиотических (знаковых) систем, которые выражают некоторое содержание.

Процесс получения информации в знаковых системах проходит в три этапа, каждому из которых соответствует свой раздел семиотики.

Определение 2.1.4. Синтактика – раздел семиотики, изучающий внутренние свойства систем знаков безотносительно к интерпретации.

Определение 2.1.5. Семантика – раздел семиотики, рассматривающий отношение знаков к обозначаемому (содержание знаков) или, что то же, соотношения между знаками и их интерпретациями, независимо от того, кто служит адресатом (интерпретатором).

Термин семантически значимый предполагает значимость с точки зрения семантики, изучающей семантические отношения, которые образуются между объектами и знаками, представляющими эти объекты в знаковой системе.

Определение 2.1.6. Прагматика – раздел семиотики, изучающий восприятие выражений знаковой системы в соответствии с разрешающими способностями воспринимающего.

Прагматика исследует связь знаков с адресатом, т. е. проблемы интерпретации знаков теми, кто их использует, их полезность и ценность для интерпретатора.

4. Какие разделы семиотики изучают эти этапы?

Определение 2.1.4. Синтактика – раздел семиотики, изучающий внутренние свойства систем знаков безотносительно к интерпретации.

Определение 2.1.5. Семантика – раздел семиотики, рассматривающий отношение знаков к обозначаемому (содержание знаков) или, что то же, соотношения между знаками и их интерпретациями, независимо от того, кто служит адресатом (интерпретатором).

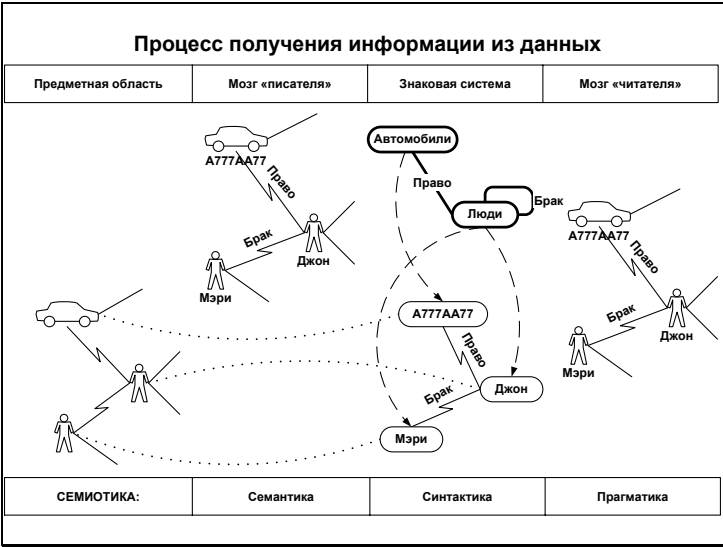
Термин семантически значимый, который мы в дальнейшем будем часто использовать, предполагает значимость с точки зрения семантики, изучающей семантические отношения, которые образуются между объектами и знаками, представляющими эти объекты в знаковой системе.

Определение 2.1.6. Прагматика – раздел семиотики, изучающий восприятие выражений знаковой системы в соответствии с разрешающими способностями воспринимающего. Прагматика исследует связь знаков с адресатом, т. е. проблемы интерпретации знаков теми, кто их использует, их полезность и ценность для интерпретатора.

5. К какому классу информационных систем относятся системы БД?

Системы БД относятся к классу информационных систем, интерпретации которых хранятся в программе и на диске в виде специальным образом организованных данных. В системах БД интерпретации не фиксируются в программах, а хранятся на диске, как и сами данные. Именно их мы задаем на этапе проектирования схемы БД. Заданные однократно интерпретации ассоциируются с данными при их вводе, и в дальнейшем при всех манипуляциях с данными эта связь остается неразрывной, обеспечивая, таким образом, получение информации.

6. Опишите, как протекает процесс передачи информации о ПрО с использованием систем БД.



Как осуществляется передача информации о ПрО в технологии БД:

В ПрО существуют некоторые объекты, обладающие характеристиками и вступающие в связи с другими объектами. Человек, чьей обязанностью является наблюдение за ПрО и отражение всех изменений в БД (назовем его писателем), узнал о существовании этих объектов и связей. При этом он произвел у себя в голове первичную формализацию ситуации, в частности, определил знаки для идентификации объектов и связей

После этого писатель обратился к знаковой системе – системе БД. В ее схеме он обнаружил подходящую подсхему и создал в БД новые объекты, а также связи.

Теперь по мере необходимости любой читатель может воссоздать представления писателя о ПрО, не прибегая к ее непосредственному исследованию. Вместо этого он обращается к системе.

7. В чем заключается основное назначение модели данных?

Определение 2.1.7. (Функциональное определение модели данных). Модель данных (МД) – это интеллектуальное средство, позволяющее реализовать интерпретацию данных и таким образом способствующее получению информации.

8. Из каких компонентов состоит атомарная единица информации (АЕИ)?

Основными объектами исследований в моделировании данных являются данные и их интерпретации. Эта область человеческих знаний, как и многие другие, имеет свой **элементарный объект – атомарную единицу информации (АЕИ)**. Она определяется четверкой – **<Идентификатор объекта, Наименование признака, Значение признака, [Время]>**.

Каждая АЕИ задает истинность следующего факта: объект, на который указывает идентификатор, имеет определенное значение признака, заданного именем, в конкретное время.

9. Объясните, почему первые три компонента АЕИ являются обязательными.

Последний элемент четверки может быть опущен. **Все остальные элементы являются обязательными. Если опустить хотя бы один из них, информация не образуется.** Что же касается времени, то не существует моделей данных, в которых механизмы работы с данными были бы доведены до идеала. Большинство моделей БД предполагают хранение информации об одном (текущем) состоянии ПрО. Такие БД еще называют оперативными, подчеркивая тот факт, что они меняются синхронно с изменением состояния ПрО. Предыдущие состояния данных в них не сохраняются. Поэтому в дальнейшем под АЕИ будем понимать тройку – **<Идентификатор объекта, Наименование признака, Значение признака>**.

10. С чем ассоциируются понятия **схема БД и база данных при табличном представлении данных?**

Определение 2.1.8. Совокупность именованных категорий и их признаков, а также ограничений на допустимые данные называется **схемой БД**.

Определение 2.1.9. Совокупность данных, структура и значения которых соответствуют конкретной схеме, называется **базой данных (БД)**.

Часто база данных ассоциируется с одной системой. Клиенты подключаются к веб-сайту, который обращается к базе данных для получения информации. БД временных рядов созданы под множество клиентов (программ). Здесь нет простого сервера, обращающегося к БД, но есть куча разных сенсоров (к примеру), выполняющих вставку данных одновременно. Как следствие, её инструменты были разработаны для того, чтобы предоставить эффективные способы потребления или производства данных.

По поводу этих определений следует сделать несколько замечаний.

Во-первых, каждая модель данных предполагает свой набор понятий, используя которые мы структурируем ПрО. Наши определения даны для категориальной модели, поэтому в них используются понятия категория и признак.

Во-вторых, в этих определениях указано основное содержание понятий **схема** и **база данных**. Иногда это содержание так или иначе расширяют. Так в **схему БД** могут быть включены программы обработки данных, определения запросов, диалоговых форм и т.д. БД тоже может представляться как все, что хранится на диске помимо собственно СУБД. Мы и сами иногда будем поступать так.

11. Синтезируйте категориальную модель.

Понятие категории является основным структурным понятием одноименной модели – **категориальной модели**. Она предполагает разбиение всех объектов ПрО по категориям. Для каждой категории определяется набор признаков, значения которых характеризуют объекты данной категории.

//Последовательно избавимся от этой избыточности. Сначала построим таблицы признаков (Рост и Вес), а затем соберем в одну таблицу значения всех признаков однотипных объектов. Полученная таблица описывает все объекты одной категории – ЧЕЛОВЕК. //

12. Дайте структурное определение модели данных.

Определение 2.1.10. (Структурное определение модели данных). Модель данных (МД)

определяется двумя множествами G и O . G – множество правил порождения схем, O – множество операций над данными. В свою очередь во множестве G выделяются два подмножества – G_s (правила порождения структур данных) и G_c (правила порождения ограничений целостности).

13. Проиллюстрируйте компоненты модели данных на примере категориальной модели.

1-я модель данных: правила порождения допустимых структур данных.

Для категориальной модели правила множества порождения структур данных выглядят следующим образом:

- БД – это совокупность таблиц.
- Каждая таблица предназначена для хранения информации об объектах одной категории. Имя таблицы – это имя категории.
- Для каждой категории определяется набор признаков, представляющих интерес для объектов этой категории. Имена признаков составляют шапку соответствующей таблицы.
- Каждый объект категории представляется в виде строки таблицы, в столбце признака указывается его значение для данного объекта.

2-я модель данных: правила порождения ограничений целостности данных.

Допустимые значения признаков можно ограничить:

- указанием их типа (символьные, числовые, даты и т.д.),
- перечислением этих значений,
- сравнением значений с константой.

3-я модель данных: набор операций над данными.

Множество операций над данными может выглядеть для нашей модели так

- операция INSERT для добавления новой строки в таблицу,
- операция UPDATE для изменения значений одного или нескольких признаков в строке таблицы,

- операция DELETE для удаления строки из таблицы,
- операция SELECT для поиска строк таблицы, удовлетворяющих определенному условию.

целостности:

12. Проиллюстрируйте компоненты модели данных на примере категориальной модели.

Категория человек

- ЧЕЛОВЕК (Фамилия, Имя, Отчество, Возраст, Пол, Рост, Вес, ...)

Используя указанные правила, зададим следующие ограничения целостности:

- Значения признаков Фамилия, Имя, Отчество, Пол имеют тип СТРОКИ.
- Значения признаков Возраст, Рост, Вес имеют тип ЦЕЛЫЕ ЧИСЛА.
- Признак Пол имеет два допустимых значения 'мужской', 'женский'.
- Признак Возраст принимает значения, большие 0 и меньше 150.

Состояние БД может быть следующим.

ЧЕЛОВЕК

Фамилия	Имя	Отчество	Возраст	Пол	Рост	Вес	...
Иванов	Иван	Иванович	50	мужской	180	80	...
Петров	Петр	Петрович	30	мужской	185	85	...
Попова	Мария	Олеговна	40	женский	165	70	...

14. Что представляет собой СУБД? Какие классы операций необходимо реализовать.

Определение 2.1.11. Управление БД на ЭВМ осуществляется специализированными программными средствами – **системами управления базами данных (СУБД)**, каждая из которых предлагает свои языковые и диалоговые формы для множеств *G* и *O*: язык определения данных (ЯОД) и язык манипулирования данными (ЯМД). Иногда выделяют отдельный язык определения ограничений целостности (ЯООЦ), но чаще ограничения целостности задаются вместе со структурой в командах ЯОД.

СУБД (Система управления базами данных) — комплекс специализированных программ, которые позволяют создать базу данных (БД) и манипулировать данными (вставлять, обновлять, удалять и выбирать). Система обеспечивает безопасность, надёжность хранения и целостность данных, а также предоставляет средства для администрирования БД.

Классы операций, которые необходимо реализовать в любой СУБД:

- операция INSERT для добавления новой строки в таблицу,
- операция UPDATE для изменения значений одного или нескольких признаков в строке таблицы,
- операция DELETE для удаления строки из таблицы,
- операция SELECT для поиска строк таблицы, удовлетворяющих определенному условию.

Глава 2.2

1. Что представляют собой элементарные единицы данных и интерпретаций. Для чего они используются в технологии БД?

Определение 2.2.1. **Знак** – это данное (строка символов, число, дата или их агрегат), определяющее конкретный объект, связь объектов или значение характеристики объекта.

Определение 2.2.2. Тип – это именованный класс подобных знаков. Имя типа определяет интерпретацию принадлежащих типу знаков.

В БД они используются для структуризации, которая в свою очередь нужна для построения такой системы типов с использованием понятий, предлагаемых правилами структуризации данных этой модели.

2. В каких частях системы БД представлены элементарные единицы данных и интерпретаций?

Определение 2.2.4. Обобщение – это абстракция, которая позволяет соотнести множество знаков или типов с одним общим типом в соответствии с отношением есть некоторый. В зависимости от вида объектов обобщения рассматривают два частных случая обобщения – классификацию и генерализацию. **Классификация** – это обобщение знаков до типа (отношение экземпляр-класс), а **генерализация** – это обобщение типов до типа (отношение подкласс-суперкласс).

Определение 2.2.5. Агрегация – это абстракция, посредством которой объект конструируется из других, базовых объектов в соответствии с отношением есть часть. Агрегация возможна как на уровне знаков (сложный знак собирается из более простых знаков), так и на уровне типов (сложный тип собирается из более простых типов). Более того, агрегация на уровне типов предполагает множество агрегаций на уровне знаков.

2. В каких частях системы БД представлены знаки и типы?

Структурный компонент схемы БД представляет собой определения типов (типов объектов ПрО, типов их связей и их характеристик). Типы в схеме не изолированы друг от друга, а образуют взаимосвязанную систему типов. Задачей структуризации ПрО в конкретной модели данных и является построение такой системы типов с использованием понятий, предлагаемых правилами структуризации данных этой модели.

3. Какие мыслительные процедуры используются при структуризации данных? Уточните, когда применяется каждая из них.

Абстракция - метод исследования, основанный на том, что при изучении какого-то явления не учитываются его несущественные стороны и признаки; это позволяет упрощать картину изучаемого явления и рассматривать его в чистом виде.

Абстракция предполагает, что несущественные частные детали должны быть опущены, а внимание должно быть сосредоточено на важных общих особенностях.

Абстракция используется для придания множествам объектов некоторой семантики.

В моделировании данных абстракция используется для образования категорий данных. Кроме того, абстракция позволяет на основе одних категорий построить другие, более общие.

В технологии БД выделяют два вида абстракции: обобщение и агрегацию.

- Обобщение – это абстракция, которая позволяет соотнести множество знаков или типов с одним общим типом с отношением есть некоторый. В зависимости от вида объектов обобщения рассматривают два частных случая обобщения – классификацию и генерализацию. Можно использовать для строительства многоуровневых иерархий.

- Классификация – это обобщение знаков до типа (отношение экземпляр-класс).

- Генерализация – это обобщение типов до типа (отношение подкласс-суперкласс). Все признаки обобщенного типа могут быть унаследованы базовыми типами.

- Агрегация – это абстракция, при которой объект конструируется из других, базовых объектов с отношением есть часть. Агрегация возможна на уровне знаков (сложный знак собирается из более простых знаков) и на уровне типов (сложный тип собирается из более простых типов). Более того, агрегация на уровне типов предполагает множество агрегаций на уровне знаков. Так же как и обобщение, агрегацию можно использовать для строительства многоуровневых иерархий.

Агрегация и обобщение применяются взаимодополняющим образом и выражают структурные и классификационные аспекты типизации. Структуру типа можно представить как агрегат базовых типов, а сам агрегат будет объектом обобщения. Классификация типов может быть выражена иерархией обобщения, а структура – иерархией агрегации.

4. Поясните, как абстракции используются для интерпретации данных (знаков).

Абстракция, как предмет, представляет собой совокупность деталей конкретного явления или группы явлений, которая может быть соответствующим образом рассмотрена, как самостоятельное целое. Абстракция, как деятельность, предполагает, что несущественные частные детали должны быть опущены, а внимание должно быть сосредоточено на важных общих особенностях.

В моделировании данных абстракция используется для образования категорий данных. Кроме того, абстракция позволяет на основе одних категорий построить другие, более общие.

В технологии БД выделяют два вида абстракции: обобщение и агрегацию.

Определение 2.2.4. Обобщение – это абстракция, которая позволяет соотнести множество знаков или типов с одним общим типом в соответствии с отношением есть некоторый. В зависимости от вида объектов обобщения рассматривают два частных случая обобщения – классификацию и генерализацию. **Классификация** – это обобщение знаков до типа (отношение экземпляра-класс), а **генерализация** – это обобщение типов до типа (отношение подкласс-суперкласс).

Определение 2.2.5. Агрегация – это абстракция, посредством которой объект конструируется из других, базовых объектов в соответствии с отношением есть часть. Агрегация возможна как на уровне знаков (сложный знак собирается из более простых знаков), так и на уровне типов (сложный тип собирается из более простых типов). Более того, агрегация на уровне типов предполагает множество агрегаций на уровне знаков.

5. Перечислите традиционно используемые в моделях данных формы данных.

Множество - это собрание правильно идентифицированных объектов, удовлетворяющих условию принадлежности.

Множество – это комплекс, все элементы которого находятся в одной позиции.

Комплекс - это собрание правильно идентифицированных объектов, удовлетворяющих условию i -принадлежности. Отношение i -принадлежности между элементом x и комплексом Y наблюдается тогда и только тогда, когда x находится в i -ой позиции комплекса Y .

В отличие от множества элементы комплекса распределены по позициям, определяемым целыми положительными числами. Причем в каждой позиции может быть любое число различных элементов.

Кортеж – это комплекс, который имеет по одному элементу в каждой позиции от 1 до n .

Домен - множества, элементы которых синтаксически однородны, например, множество целых чисел от 60 до 70, строки букв длиной до 10 символов и т.д.

Атрибуты - именованные домены, представляющие семантически значимые объекты. Атрибуты являются интерпретацией объектов реального мира и их характеристик.

6. Укажите свойства множеств.

Определение 2.2.6. Множество – это собрание правильно идентифицированных объектов, удовлетворяющих условию принадлежности.

В классической теории множеств эта форма характеризуется следующими свойствами:

- множества необязательно должны иметь фиксированное количество элементов, они могут быть и бесконечными;
- на элементах множества отсутствует какой-либо порядок;
- элементы-дубликаты во множестве не имеют смысла.

7. Дайте определения интенционала и экстенционала множества. Почему в моделировании данных уместно говорить о нескольких реализациях множества?

Интенционал множества определяет свойства, общие для всех элементов всех реализаций множества. (студент, сидящий в 104 аудитории)

Экстенционал множества определяет актуальную реализацию множества путем явного указания его элементов. (Иванов, Петров)

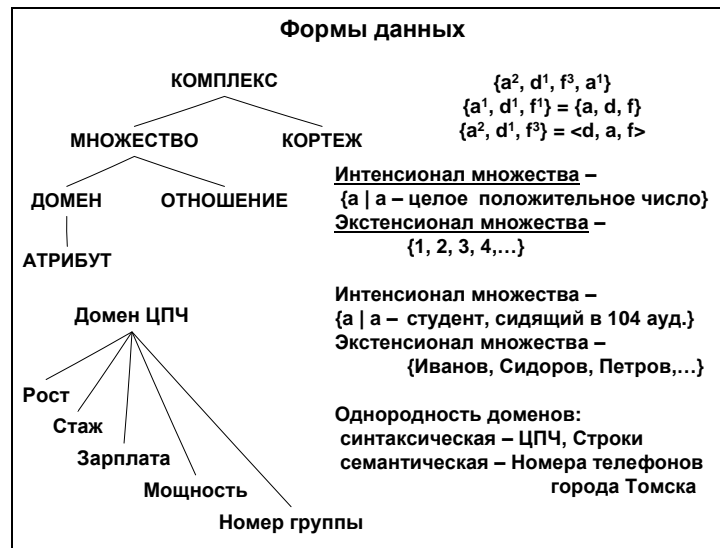
При моделировании в БД динамичных ПрО одному и тому же множеству в различные моменты времени могут соответствовать различные реализации. В качестве примера можно рассмотреть множество студентов, сидящих в конкретной аудитории. Его экстенционал будет меняться чуть ли не на каждой паре занятий при неизменном интенционале.

8. Чем отличается комплекс от множества?

Определение 2.2.7. Комплекс – это собрание правильно идентифицированных объектов, удовлетворяющих условию i -принадлежности. Отношение i -принадлежности между элементом x и комплексом Y (записывается $x \in_i Y$) наблюдается тогда и только тогда, когда x находится в i -ой позиции комплекса Y .

В отличие от множества элементы комплекса распределены по позициям, определяемым целыми положительными числами. Причем в каждой позиции может быть любое (в т.ч. и бесконечное) число различных элементов. В пределах одной позиции порядок элементов не определен. В разных позициях могут быть совпадающие элементы. Позиция элемента в комплексе записывается с помощью верхнего индекса.

9. Покажите, что понятия множество и кортеж являются специализациями понятия комплекс.



Множества и n -местные кортежи есть специальные случаи комплексов.

Определение 2.2.8. Множество – это комплекс, все элементы которого находятся в одной, предположим первой позиции. n -местный кортеж или просто **кортеж** – это комплекс, который имеет по одному элементу в каждой позиции от 1 до n .

Сам комплекс, в отличие от его специализаций, не является структурным понятием ни одной из моделей данных. Однако это понятие способствует правильному восприятию других, встречающихся на практике форм данных.

10. В чем принципиальное отличие элементов, принадлежащих домену и атрибуту?

Определение 2.2.10. Существуют некоторые множества, элементы которых более или менее синтаксически однородны. Подобные однородные множества в моделировании данных носят название **доменов**.

Определение 2.2.11. Именованные домены, представляющие семантически значимые объекты, называются **атрибутами**. Атрибуты являются интерпретацией объектов реального мира и их характеристик.

Домен можно рассматривать как обобщение атрибутов. Атрибуты, определенные на общем домене, наследуют его свойства. И, наоборот, домен обладает всеми общими свойствами определенных на нем атрибутов. Обобщение в применении к атрибутам позволяет соотнести данные с доменами и установить общность различных атрибутов.

В отличие от элементов домена, элементы атрибута обладают не только синтаксической однородностью, но и семантической (в их условии принадлежности фигурируют особенности конкретной предметной области). Например, элементами домена могут быть даты, а элементами атрибута даты рождения взрослого населения России.

11. Дайте определения и уточните различия определений понятия отношение в математике, логике и моделировании данных.

Отношение

Математика

n -местным отношением R на множествах A_1, \dots, A_n называется подмножество прямого произведения $A_1 \times \dots \times A_n$. Другими словами, элементы x_1, \dots, x_n , где $x_i \in A_i (i = 1, \dots, n)$, связаны отношением R тогда и только тогда, когда $\langle x_1, \dots, x_n \rangle \in R$.

Пусть R – бинарное отношение. Определим обратное отношение R^{-1} следующим образом: $R^{-1} = \{\langle x, y \rangle | \langle y, x \rangle \in R\}$. Таким образом, R^{-1} связывает те же пары элементов, что и R , но «в другом порядке».

Логика

Пусть $E^n = E \times \dots \times E$ есть произведение n множеств E , т.е. множество всех кортежей $\langle x_1, \dots, x_n \rangle$, $x_i \in E (i = 1, \dots, n)$.
Отображение $P: E^n \rightarrow \{0, 1\}$ называется n -местным отношением (предикатом, логической функцией) над E . Множество $A \subseteq E^n$ всех кортежей, для которых $P(x_1, \dots, x_n) = 1$, определяет «свойство» кортежей: x_1, \dots, x_n состоят в отношении P тогда и только тогда, когда $\langle x_1, \dots, x_n \rangle \in A$.

Отношение

Моделирование данных

Пусть задано множество из n типов или доменов $T_i (i = 1, \dots, n)$, причем все они необязательно должны быть различными. Тогда r будет отношением, определенным на этих типах, если оно состоит из двух частей: заголовка и тела (заголовок еще иногда называют схемой или интенционалом отношения, а тело – экстенционалом отношения), где:
- заголовок – это множество из n атрибутов вида $A_i: T_i$; здесь A_i – имена атрибутов отношения r , а T_i – соответствующие имена типов;
- тело – это множество из m кортежей t ; здесь t является множеством компонентов вида $A_i: v_i$, в которых v_i – значение типа T_i , т.е. значение атрибута A_i в кортеже t .

Человека, прежде всего, интересует истинность или ложность того или иного факта, поэтому ему ближе и понятней определение логики. В информационных системах (и в базах данных, в частности) принято хранить только истинные факты и предполагать выполнение гипотезы о замкнутости мира: если кортеж не содержится в теле отношения, то соответствующее ему утверждение ложно.

Отношение в моделировании данных отличается от математического отношения отсутствием порядка в элементах кортежей, и, таким образом, одно отношение в смысле определения Отношение в моделировании данных соответствует сразу нескольким отношениям, в смысле определения математического отношения. Именно поэтому отношение в моделировании данных используется в базах данных.

12. В каком виде задаются в БД интерпретации данных?

Интерпретации данных строятся на основе форм данных, отношений и мыслительных процессов для структуризации данных. Интерпретации данных задаются в виде таблиц.

13. Каким формам данных приписываются интерпретации?

Через формы данных, такие как **кортежи**, **множества** и их вытекающие (домен (атрибут), отношения).

Основной принцип очень прост – интерпретации наследуются в соответствии с иерархией обобщения. Возможно, простейшим и широко распространенным способом представления данных и их интерпретаций являются таблицы. С ними люди привыкли работать повседневно. Табличные формы представления характерны и для большинства моделей данных.

14. Как они используются для интерпретации знаков (данных)?

Интерпретации наследуются в соответствии с иерархией обобщения. Поскольку каждый знак (данные) состоит в отношении есть некоторый со своим типом, для него истинно условие принадлежности (интенционал) множества, в форме которого представлены все знаки этого типа.

Интерпретация как типов объектов

- Множество используется для представления имен типов
- Кортж представляет множество значений, отношений, характеризующий данный объект

Интерпретация как тип связей между типами объектов

- Кортж, элементы которого являются знаки соответствующих объектов, задает связь конкретного типа

15. Установите соответствие между формами данных и их представлениями в виде таблиц и графов.

Объекты одного типа представляются отдельной таблицей, олицетворяющей соответствующее отношение. Заголовок таблицы (отношения) представляет собой общее имя и характеризует соответствующее понятие о предметах. Каждая строка таблицы (кортж) представляет конкретный объект и задает его значения однозначных характеристик, общих для всех объектов одного типа. Каждый столбец таблицы представляет один атрибут, имя которого помещается в шапку таблицы.

16. Что ассоциируется с понятиями интенционал БД и экстенционал БД?

Определение 2.3.20. **Интенционал БД** представляет собой схему БД, включающую определения структур данных и ограничения целостности. **Экстенционал БД** составляют реализации определенных в схеме БД форм данных – множеств и отношений.

Глава 2.3

1. Что такое ограничение целостности (ОЦ) и для чего они предназначены?

Определение 2.3.1. **Ограничение целостности (ОЦ)** можно представлять себе как логическое условие, которое для реализации данного множества, атрибута, отношения или нескольких отношений либо истинно, либо ложно.

ОЦ вводятся в модели данных в целях повышения ее семантичности и расширения возможностей поддержания целостности данных. Первый аспект связан с адекватностью отражения реального мира в схеме, а второй – с возможностями СУБД обеспечивать соответствие порождаемых состояний БД требованиям, выражаемым ОЦ.

2. Дайте определения понятий, связанных с процессом верификации ОЦ.
С процессом верификации декларативных ОЦ в СУБД связан ряд определений.

Определение 2.3.2. Рассмотрим явное ограничение C_i , указанное в схеме S и состояние БД DBS_k . Это ограничение будет:

- 1) **правильно построено**, если оно соответствует синтаксическим правилам задания ОЦ;
- 2) **удовлетворено состоянием** БД DBS_k , если оно истинно для него;
- 3) **удовлетворяемо**, если существует некоторое состояние БД DBS_k , удовлетворяющее C_i ;
- 4) **недостоверно**, если никакое состояние БД не удовлетворяет C_i ;
- 5) **логическим следствием** C_1, \dots, C_n (т.е. избыточным ограничением), если C_i удовлетворяют все состояния БД, которые удовлетворяют C_1, \dots, C_n ;
- 6) **эквивалентно** C_j , если C_i и C_j – логические следствия друг друга.

3. Охарактеризуйте способы задания ОЦ с точки зрения их предпочтительности.

Выделяют **задание внутренних и явных ОЦ**. Первые тесно связаны с правилами структуризации, предполагаются самими формами данных и не требуют никаких дополнительных деклараций. Диапазон возможностей внутренних ограничений весьма узок. Поэтому в моделях данных предусматривают правила задания явных ОЦ, существенно расширяющие возможности передачи в схеме БД закономерностей ПрО. Чем больше ограничен диапазон структур данных, на представление которых ориентирована модель, тем больше число видов внутренних ОЦ она предусматривает и тем меньше видов явных ОЦ необходимо и возможно задавать.

4. Какие типы ОЦ выделяются в моделировании данных? Каковы их области действия?

Типизация ограничений целостности

СЛУЖАЩИЙ — СЛУЖБА — КОМПАНИЯ

Атрибуты СЛУЖАЩИЙ: № СЛУЖАЩЕГО, ФАМИЛИЯ, АДРЕС, ПОЛ. Атрибуты КОМПАНИЯ: НАЗВАНИЕ, ГОРОД.

1. Ограничения на значения атрибутов

2. Ограничения на отображения:

а) между атрибутами одного отношения

б) между отношениями

Большинство ОЦ можно отнести к одному из следующих типов:

- 1) Ограничения на значения атрибутов.
- 2) Ограничения на отображения:
 - а) между атрибутами одного отношения,
 - б) между отношениями.

На схеме показаны области действия каждого из типов ОЦ.

В моделировании данных особую значимость имеют обобщенные ОЦ, т.е. ограничения, относящиеся к интенциональным свойствам данных. Это значит, что ОЦ фиксируют в схеме общие закономерности данных о ПрО, которые выполнялись в прошлом, наблюдаются сейчас и будут справедливы и в будущем.

5. Перечислите виды ОЦ на значения атрибутов.

1) Ограничения целостности на значения атрибутов

Множество допустимых значений атрибутов можно задавать:

- 1) принадлежностью к определенному типу или домену
Рост INTEGER
- 2) сравнением с константой или значением атрибута того же отношения (возможны более сложные выражения)
Рост > 50 или Рост > Вес + 110
- 3) диапазоном
Рост BETWEEN 50 AND 300
- 4) перечислением значений
Пол IN {'м', 'ж'}
- 5) более сложным логическим выражением, включающим в виде атомов конструкции 2 – 4
(Пол = 'ж' AND Рост > Вес + 110) OR
(Пол = 'м' AND Рост > 180)

По поводу ОЦ, определяющих принадлежность атрибута к тому или иному типу значений или домену, следует сделать одно замечание. Их можно трактовать как:

- структурное определение,
- внутреннее ограничение целостности,
- явное ограничение целостности

6. В чем особенность традиционного определения понятия отображение в математике?

Отображение в математике

Отображение (функция, оператор) есть закон соответствия, сопоставляющий каждому элементу множества A некоторый (единственный) элемент множества B . $\varphi: A \rightarrow B$ означает, что задано отображение A в B , называемое φ .

Классификация Кофмана

Соответствие Γ между множествами E_1 и E_2 определено, если задан обычный граф $G \subseteq E_1 \times E_2$. Тогда говорят, что G – граф соответствия Γ , E_1 – область определения, а E_2 – область значений Γ . Соответствие, обратное Γ , обозначается Γ^{-1} , и E_2 – область определения, а E_1 – область значений Γ^{-1} .

Отображением множества E_1 во множество E_2 называется такое соответствие, которое любому $x \in E_1$ сопоставляет, по крайней мере, один $y \in E_2$. Тогда говорят, что элемент y – образ элемента x , а x – переменная или аргумент.

Функцией E_1 в E_2 называется такое отображение, которое каждому $x \in E_1$ сопоставляет один и только один $y \in E_2$.

Традиции, сложившиеся в математике, требуют обязательного наличия единственного образа для каждого аргумента.

7. Укажите основные отличия математического отображения от семантически значимого отображения.

Традиции, сложившиеся в математике, требуют обязательного наличия единственного образа для каждого аргумента.

Определение 2.3.3. Семантически значимое отображение – это понятие, определяющее некоторый закон предметной области, по которому каждому объекту моделируемого мира может быть поставлен в соответствие (а может быть, и нет) один или более объектов.

8. Дайте определения, характеризующие роли элементов множеств, участвующих в отображении.

Один из экземпляров отображения РЕБЕНОК показан в текстовой форме (Саша = ребенок (Сидоров)). Для этого использована традиционная для функции нотация $=$ ($()$). Ниже представлены экземпляры обратного отображению РЕБЕНОК отображения РОДИТЕЛЬ. Они передают ту же самую информацию, только сменилась ориентация отображения, и объекты поменялись ролями (образы стали прообразами и наоборот). *Это связано с тем, что отображение РЕБЕНОК эквивалентно инверсии отображения РОДИТЕЛЬ, и оба этих отображения полностью определяются бинарным отношением РОДИТЕЛЬ-РЕБЕНОК, представленным в табличной форме в верхней части слайда

9. Почему в моделировании данных приходится рассматривать реальные и потенциальные ООО и ОЗО?

Совокупность всех объектов, имеющих образы при отображении φ назовем **реальной областью определения отображения (РООО)** φ , а совокупность всех φ -образов этих объектов – **реальной областью значений отображения (РОЗО)** φ . Семантика же отображений стабильна и в любой момент времени отражает тот закон, который справедлив для всех возможных состояний их экземпляров. По этой причине наряду с динамичными РООО и РОЗО для отображений необходимо рассматривать потенциальные области определения и значений отображений. **Область определения отображения φ (ООО)** – это совокупность объектов, которые в силу смысла закона φ могли иметь в прошлом или имеют в настоящем или смогут иметь в будущем хотя бы один φ -образ. **Область значений отображения φ (ОЗО)** – это совокупность объектов, которые в силу смысла закона φ могли быть в прошлом или являются в настоящем или смогут стать в будущем φ -образами некоторых объектов

10. Поясните фразу бинарное отношение множеств определяет два отображения между ними.

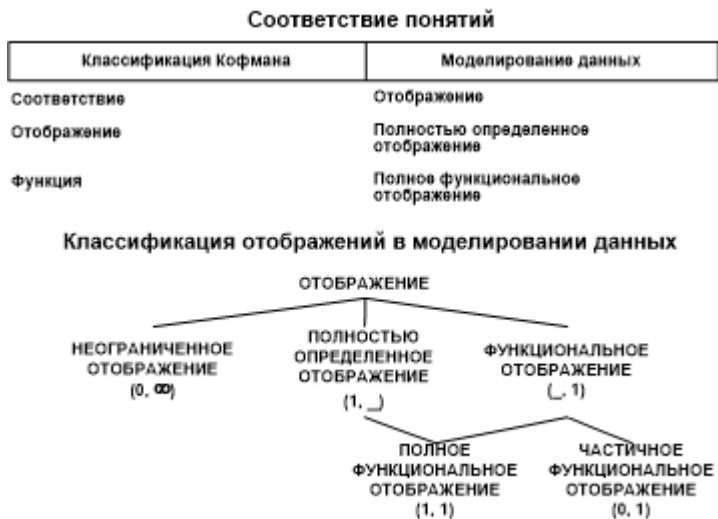
Есть определение отображения в моделировании данных: Бинарное отношение R множеств S_1 и S_2 определяет два отображения $R: S_1 \rightarrow S_2$ и $R^{-1}: S_2 \rightarrow S_1$, каждое из которых является обратным по отношению к другому. Другими словами, отображение R эквивалентно инверсии отображения самого себя. При таком раскладе объекты меняются ролями (образы становятся прообразами, и наоборот). Если один из атрибутов отношения использовать как РООО, а второй – как РОЗО, каждый кортеж будет определять один экземпляр одного отображения. Если поменять атрибуты местами, те же кортежи определят экземпляры обратного отображения.

11. Дайте определения КЧ, МинКЧ и МаксКЧ. К каким элементам относятся эти характеристики?

Определение 2.3.4. φ -отображение объекта x характеризует **кардинальное число** объекта x при отображении φ (обозначение – $\hat{E}_\varphi(x)$) – мощность его области образов при этом отображении. (число образов)

Количественные характеристики отображений (продолжение)		
Минимальное	кардинальное	число (МинКЧ)
отображения φ - это наименьшее из кардинальных чисел $K\varphi(x)$ объектов x , являющихся экземплярами области определения отображения (ООО) φ :		
$\text{МинКЧ}_{\varphi} = \min_{x=\text{экземпляр}(\text{ООО}(\varphi))} K\varphi(x)$		
$\text{МинКЧ}_{\text{родитель}} = 2 \quad \text{МинКЧ}_{\text{ребенок}} = 0$		
Максимальное	кардинальное	число (МаксКЧ)
отображения φ - это наибольшее из кардинальных чисел $K\varphi(x)$ объектов x , являющихся экземплярами области определения отображения (ООО) φ :		
$\text{МаксКЧ}_{\varphi} = \max_{x=\text{экземпляр}(\text{ООО}(\varphi))} K\varphi(x)$		
$\text{МаксКЧ}_{\text{родитель}} = 2 \quad \text{МаксКЧ}_{\text{ребенок}} = \infty$		

12. Перечислите и укажите характеристики введенных типов отображений.



Отображение называется **неограниченным**, если любой прообраз может или вообще не иметь образа, или иметь произвольное количество образов.

Пусть для отображения $S_1 \rightarrow S_2$ имеем пару кардинальных чисел $(1, \infty)$, т.е. предполагается, что каждый объект S_1 отображен, по крайней мере, в один объект S_2 . Такое отображение называется **полностью определенным**.

Если максимальное кардинальное число равно единице, то отображение есть функция. Так, $R(S_1(0, \infty): S_2(0, 1))$ определяет **функциональное отображение** $S_1 \rightarrow S_2$, т.е. любой объект S_1 отображается не более чем одним объектом S_2 (хотя в данном случае не все объекты S_1 обязаны иметь образы в S_2). Представлению функционального отображения в форме графа свойственно не более одной дуги, выходящей из вершины-прообраза.

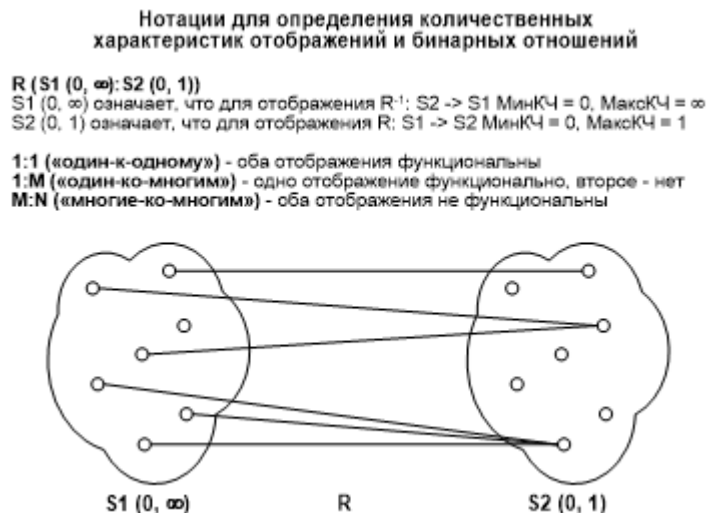
Если МинКЧ функционального отображения равно 0, такое отображение называется **частичным функциональным отображением**. В таком случае любой прообраз может либо не иметь образа, либо иметь не более одного образа. Соответственно, в графе такого отображения могут быть либо висящие вершины-прообразы, либо вершины-прообразы, которым инцидентна одна дуга экземпляра отображения.

Если МинКЧ функционального отображения равно 1, такое отображение называется **полным функциональным отображением**. В таком случае каждый прообраз должен иметь один и только

один образ. Если посмотреть на граф экстенционала такого отображения, то нетрудно заметить, что из каждого элемента ООО выходит ровно одна дуга экземпляра отображения.

Определение 2.3.8. Говорят, что бинарное отношение, определяющее пару функциональных отображений, есть отношение **один к одному** (**1:1**), бинарное отношение с одним функциональным, а другим нефункциональным отображениями есть отношение **один ко многим** (**1:M**) или **многие к одному** (**M:1**) и бинарное отношение, определяющее пару нефункциональных отображений, есть отношение **многие ко многим** (**M:N**).

13. Какими символическими нотациями можно описать бинарные отношения? Сравните их информативность на примере различных типов бинарных отношений.



Первая нотация обеспечивает исчерпывающую информацию, как о бинарном отношении, так и об обоих определяемых им отображениях – $R(S_1(0, \infty) : S_2(0, 1))$. Она говорит, прежде всего, о том, что бинарное отношение R определено на множествах S_1 и S_2 . $S_1(0, \infty)$ означает, что МинКЧ и МаксКЧ отображения $S_2 \rightarrow S_1$ равны 0 и ∞ соответственно. Иными словами, **любой элемент S_2 может быть связан минимум с 0 и максимум с ∞ элементов S_1** . $S_2(0, 1)$ означает, что МинКЧ и МаксКЧ отображения $S_1 \rightarrow S_2$ равны 0 и 1 соответственно. Иными словами, **любой элемент S_1 может быть связан минимум с 0 и максимум с 1 элементом S_2** .

Поскольку в БД имеют дело с ПрО, в которых и число объектов, и число связей, и число значений характеристики конечно, знак ∞ в данном случае означает отсутствие ярко выраженной верхней границы МаксКЧ. И, если оно для текущего экстенционала БД равно i , то не исключено такое изменение состояния БД, что оно станет равным $i + 1$.

Вторая нотация, хоть и используется чаще, менее информативна, поскольку характеризует лишь МаксКЧ отображений.

Определение 2.3.8. Говорят, что бинарное отношение, определяющее пару функциональных отображений, есть отношение **один к одному** (**1:1**), бинарное отношение с одним функциональным, а другим нефункциональным отображениями есть отношение **один ко многим** (**1:M**) или **многие к одному** (**M:1**) и бинарное отношение, определяющее пару нефункциональных отображений, есть отношение **многие ко многим** (**M:N**).

14. Укажите отличие сложного отображения от простого отображения. Приведите примеры сложных отображений, укажите их характеристики и тип.

Определение 2.3.14. Простые отображения, у каждого экземпляра которых есть один объект-прообраз и один объект-образ.

Сложные отображения, каждый экземпляр которых имеет более одного объекта-прообраза и/или объекта-образа. (такие отображение, у которых прообраз или образ или и то, и другое одновременно являются кортежами других отображений).

Примером является отношение: Ребенок: Мать*Отец = Ребенок (к кортежу мама, папа, ставится соответствие Ребенка)

Классическим примером такой ситуации в моделировании данных является поставка некоторым поставщиком детали для конкретного проекта. Каждый факт такой поставки представляет собой тернарную связь между конкретными поставщиком, деталью и проектом.

15. Дайте три определения возможного ключа.

Определение 2.3.15. Если атрибут (группа атрибутов) отношения функционально определяет все другие атрибуты этого отношения, этот атрибут (группа атрибутов) называется **возможным ключом (потенциальным ключом, ключом-кандидатом)** отношения.

Определение 2.3.16. Если атрибут (группа атрибутов) отношения функционально определяет отношение, этот атрибут (группа атрибутов) называется **возможным ключом (потенциальным ключом, ключом-кандидатом)** отношения. Т.е. отображение между возможным ключом и самим отношением функционально. Другими словами, возможный ключ уникально идентифицирует кортежи отношения.

Определение 2.3.17. Если атрибут (группа атрибутов) отношения не имеет значений-дубликатов в кортежах отношения, этот атрибут (группа атрибутов) называется **возможным ключом (потенциальным ключом, ключом-кандидатом)** отношения. (недостаток: смотрим на конкретный экстенционал)

16. Что означает определенность значения атрибута?

Определенность значения атрибута есть отсутствие неопределенности значения в этом атрибуте.

Неопределенные значения (часто в моделях данных они задаются псевдопеременной или константой *NULL*) используются в качестве значений атрибутов любых типов в тех случаях, когда:

- либо значение атрибута для кортежа пока не известно,
- либо данная характеристика не применима к конкретному объекту, и атрибут, соответствующий этой характеристике, никогда не примет какого-то определенного значения для кортежа, представляющего этот объект.

Так вот иногда законы ПрО требуют обязательного наличия определенного значения атрибута в каждом кортеже. В частности, одной из особенностей первичного ключа является то, что в каждом кортеже отношения он должен иметь определенное уникальное значение, иначе все его значения не смогут быть знаками объектов.

Определение 2.3.19. Если между отношением и некоторым его атрибутом отображение полностью определено (а точнее, оно – полное функциональное), этот атрибут **не может иметь неопределенных значений** в кортежах отношения.

17. Какие особенности каких отображений скрываются в определении возможного ключа и обязательности значений атрибута?

Если атрибут отношения функционально определяет все другие атрибуты этого отношения, этот атрибут называется **возможным ключом (потенциальным ключом)**.

Слова функционально определяет означают, что отображения между возможным ключом и любым атрибутом отношения функциональны.

Если атрибут отношения функционально определяет отношение, этот **атрибут называется возможным**.

Глава 2.4

1. Дайте определение расширенного состояния БД.

Расширенное состояние БД – динамика БД с введением некоторых дополнительных объектов – индикаторов текущих и других управляющих элементов. Эти объекты в строгом смысле не относятся к объектам БД, но они связаны с ней и могут изменяться в результате выполнения операций. Совместно с конкретной реализацией данных они определяют состояние БД.

Текущее состояние БД определяется совокупностью значений хранящихся данных, а также значением индикатора текущей, указывающим на строку, к которой осуществлялось последнее обращение. Выполнение операции дать следующую строку не приведет к изменению реализации БД, но состояние БД изменится, так как изменится значение индикатора текущей.

2. Что такое операция над данными?

Определение 2.4.1. Операции над данными, выражаемые средствами языка манипулирования данными, переводят БД из состояния DBS_i в состояние DBS_{i+1} (или в неопределенное состояние), причем изменение состояния происходит или при изменении реализации данных, или при изменении управляющих элементов.

3. Какие два компонента можно выделить в любой операции над данными?

Определение 2.4.2. Операции обычно задаются в терминах селекции и действия. **Действие** определяет характер операции, а **селекция** – критерий отбора данных, над которыми должно быть произведено действие.

4. Какие действия над данными предусмотрены в технологии БД?

Определение 2.4.3. В технологии БД выделяют пять основных видов действий над данными:

- а) **установка текущих** – типичные ключевые слова – *SET CURRENCY, FIND*;
- б) **включение** (добавление новых элементов данных в БД) – *INSERT, ADD*;
- в) **обновление** (модификация существующих элементов данных в БД) – *UPDATE, MODIFY*;
- г) **удаление** (исключение элементов данных из БД) – *DELETE, REMOVE*;
- д) **выборка** (получение данных из БД) – *SELECT, GET*.

5. Какими способами можно селектировать данные для выполнения тех или иных действий с ними?

Определение 2.4.4. Селекция может осуществляться посредством:

- а) логической позиции в БД – **селекция по текущей**;
- б) значений данных – **селекция по данным**;
- в) связей между данными – **селекция по связям**.

6. Какие два класса операций и языков манипулирования данными выделяются в технологии БД? Каковы их особенности?

Определение 2.4.5. Существенный признак, по которому различаются операции и языки данных, определяется характером результата единичной директивы селекции, осуществляемой в ходе выполнения одной операции.

Если результат всегда представлен единственным элементом полученным при прохождении по логическому пути (т.е. при навигации) в структуре БД, то соответствующие операции называются **навигационными**. Напротив, операции могут специфицировать как бы новую подсхему, определяемую на схеме БД, которой в общем случае соответствует множество элементов, существующих в БД. Такие операции называются **спецификационными**.

Язык, все операции которого являются навигационными, называется **навигационным языком**. Язык, все операции которого являются спецификационными, называется **спецификационным языком**. Однако встречаются языки, которые могут иметь как те, так и другие операции.

7. Что такое процедура БД?

Определение 2.4.6. Процедура базы данных представляет собой последовательность операций, выполняемых при определенных условиях. Общность этих условий обеспечивает возможность автоматического, без вмешательства пользователя, инициирования процедур.

8. Какие виды процедур БД вам известны?

Один из видов процедур БД – вычисление значений, которые непосредственно не хранятся в БД, например, вычисление сумм, подсчет числа экземпляров, определение минимума и максимума. Мы будем называть соответствующие процедуры БД **функциями агрегирования**.

Второй вид процедур БД – вычисление значения атрибута, например, вычисление возраста человека по заданной текущей дате и дате рождения, что выполняется по запросу о возрасте. Любая процедура БД, вычисляющая значение атрибута, называется **виртуальным атрибутом**.

Третий вид процедур БД – контроль целостности БД. Верификация ограничений может осуществляться автоматически или с помощью процедур БД, вызываемых явной директивой пользователя или запускаемых при возникновении специфических ситуаций или выполнении специфических операций. Процедуры БД для проверки и/или поддержки ОЦ носят название **триггеров целостности**. Такие процедуры не вырабатывают значений данных, а только уведомляют систему об успешном или неуспешном завершении или вызывают принудительное возвращение БД в целостное состояние, согласующееся с определенными ограничениями (что можно трактовать как некоторый побочный эффект).

Четвертый вид процедур БД – обеспечение контроля над доступом. Контроль над доступом не сводится только к парольной защите. Он может быть связан с анализом данных и проверкой полномочий и аутентичности пользователя. Соответствующая процедура БД может выполнять сложные вычисления, по результатам которых могут приниматься важные решения. Например, процедура БД может получать в качестве исходных данных идентификационные параметры пользователя, время, дату, вид действия и вырабатывать сообщения для пользователя, журнала или лица, ответственного за соблюдение правил доступа. Процедуры такого рода называются **триггерами безопасности**.

Другой пример – процедура удаления кортежей-дубликатов, позволяющая модифицировать спецификационную операцию, в результате которой образуется отношение. Такие процедуры называются **операторами доступа**.

Если тип связей между отношениями является полным (хотя бы одно из определяемых им отображений полностью определено), то удаление кортежа может привести к запуску процедуры, удаляющей все кортежи, связанные с удаляемым. Такие процедуры называются **триггерами, запускаемыми включением, удалением или обновлением**.

Процедуры БД могут применяться для сбора статистики или для реализации каких-либо других функций администрирования БД. Такие процедуры носят название **процедур администратора БД**. Они не изменяют данные, но могут модифицировать состояние БД.

Функции агрегирования – вычисление значений, которые непосредственно не хранятся в БД, например, вычисление сумм, подсчет числа экземпляров, определение минимума и максимума.

Виртуальный атрибут - вычисление значения атрибута, например, вычисление возраста человека по заданной текущей дате и дате рождения, что выполняется по запросу о возрасте.

Триггеры целостности - контроль целостности БД. Процедуры БД для проверки и/или поддержки ОЦ

Триггеры безопасности – обеспечение контроля над доступом, который может быть связан не только с парольной защитой, но и с анализом данных и проверкой полномочий и аутентичности пользователя. (Процедура БД может получать в качестве исходных данных идентификационные параметры пользователя, время, дату, вид действия и вырабатывать сообщения для пользователя, журнала или лица, ответственного за соблюдение правил доступа.)

Операторы доступа – процедуры БД, которые могут использоваться также для расширения языка данных операциями, первоначально в нем не предусмотренными. (например, процедура удаления кортежей-дубликатов, позволяющая модифицировать спецификационную операцию, в результате которой образуется отношение)

Триггеры, запускаемые включением, удалением или обновлением - процедуры БД, которые инициируются операциями модификации одних данных и выполняют косвенные модификации

других данных.

Процедуры администратора БД - Процедуры БД, которые могут применяться для сбора статистики или для реализации каких-либо других функций администрирования БД. Они не изменяют данные, но могут модифицировать состояние БД.

