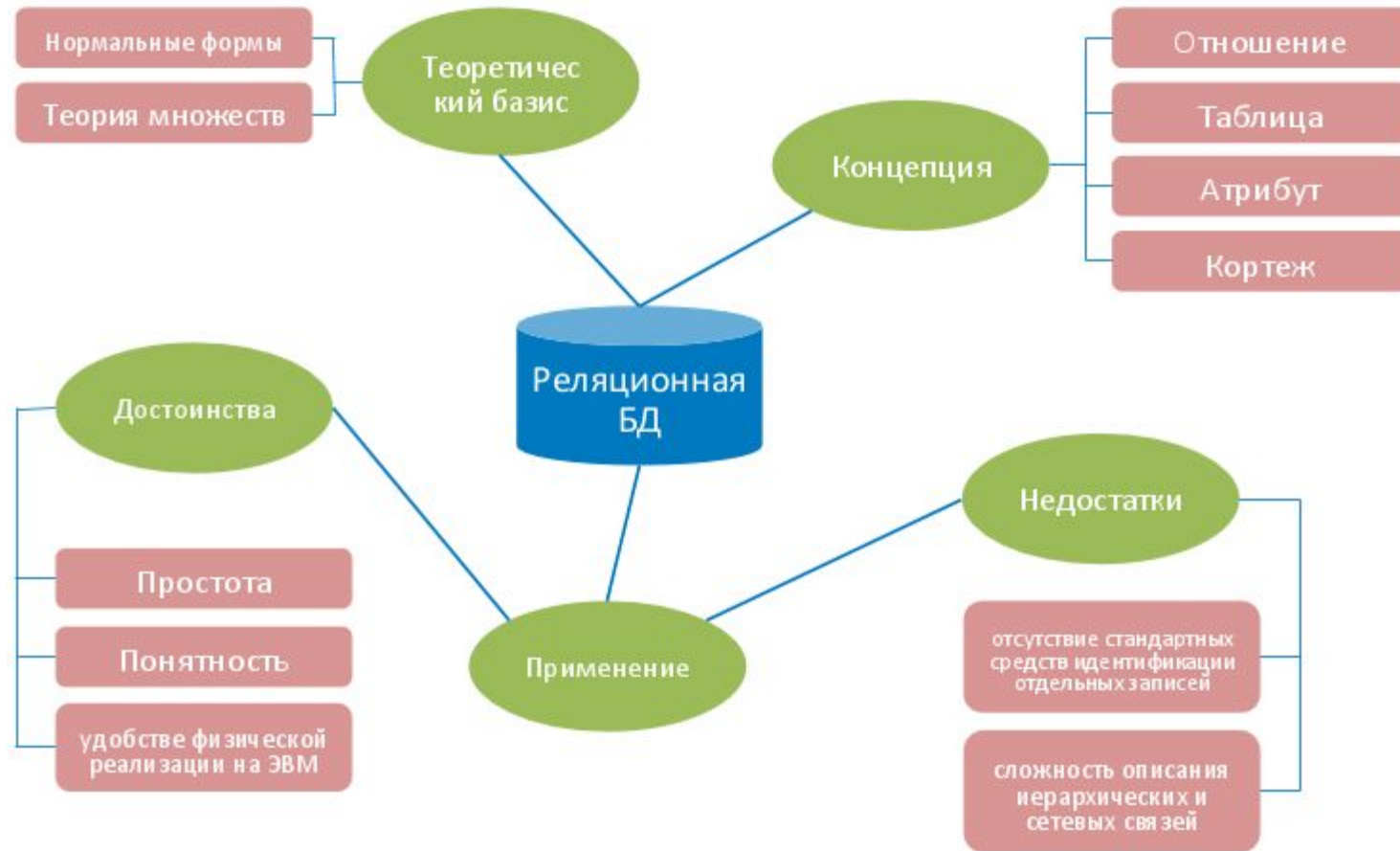




Введение в реляционные базы данных

Понятие реляционных БД



Основные понятия

1. Тип данных
2. Домен
3. Атрибут
4. Кортеж
5. Отношение
6. Первичный ключ

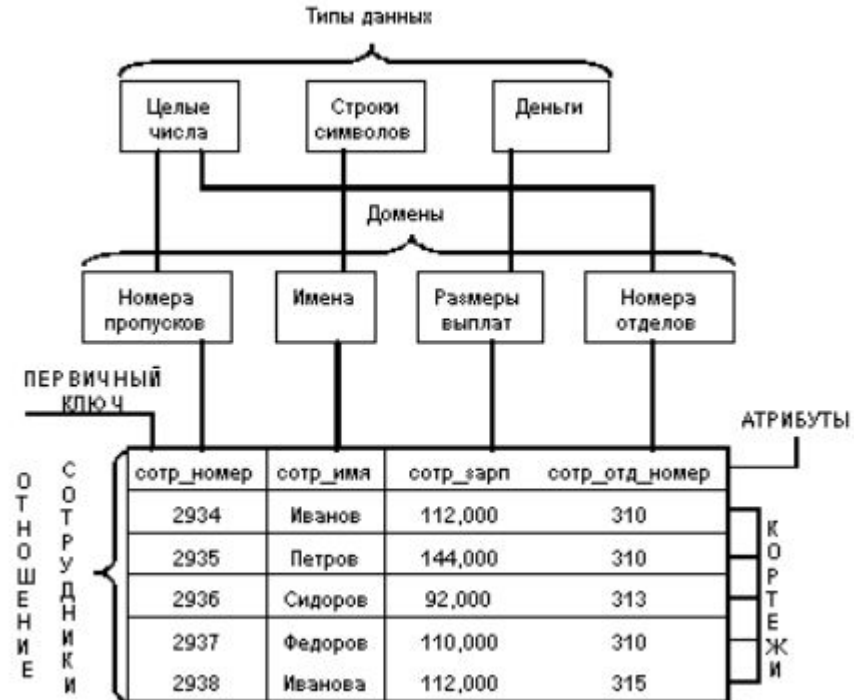


Схема БД (в структурном смысле) – это набор именованных схем отношения.
Схема реляционной базы данных – набор пар <имя_ВАRr? Nr>, включающий имена и заголовки всех переменных отношения, которые определены в базе данных

Первичные и внешние ключи

•••• Первичный ключ

- Первичным ключом (ключом отношения, ключевым атрибутом) называется атрибут отношения, однозначно идентифицирующий каждый из его кортежей.
- Подмножество **S** множества атрибутов заголовка переменной отношения
- В любое время значение первичного ключа в любом кортеже тела отношения отличается от значения первичного ключа в любом другом кортеже тела этого отношения
- Никакое собственное подмножество **S** этими свойствами не обладает
- Ключ называется составным, если в него входит более одного атрибута

•••• Внешний ключ

- Атрибут называется внешним ключом, если его значения однозначно характеризуют сущности, представленные кортежами некоторого другого отношения
- Значения внешнего ключа задают значения первичного ключа другого отношения
- Может быть составным, т.е. состоять из нескольких атрибутов

Резюме

Название из теории	Практическое название
Домен	Тип данных
Схема (заголовок) отношения	Схема (заголовок) таблицы
Атрибут	(Заголовок) столбца
Схема БД	Схема БД
Кортеж	Строка, запись
Тело отношения	Таблица
Первичный ключ	Идентификатор

Неключевой атрибут

Любой атрибут отношения, не входящий в состав ключа

Взаимно независимые атрибуты

Два или более атрибута взаимно независимы, если ни один из этих атрибутов не является функционально зависимым от других

Функциональные зависимости

Функциональная зависимость

В отношении r атрибут Y функционально зависит от атрибута X (X и Y могут быть составными) в том и только в том случае, если каждому значению X соответствует в точности одно значение Y .

Частичная функциональная зависимость

Зависимость неключевого атрибута от части составного ключа

Полная функциональная зависимость

Зависимость неключевого атрибута от всего составного ключа

Транзитивная функциональная зависимость

Функциональная зависимость $X \rightarrow Y$ называется транзитивной, если существует такой атрибут Z , что имеются функциональные зависимости $X \rightarrow Z$ и $Z \rightarrow Y$ и отсутствуют функциональные зависимости $Y \rightarrow X$, $Y \rightarrow Z$, $Z \rightarrow X$

Декомпозиция

- **Декомпозиция**

Разбиение путем проецирования отношения на два или более отношений (рис. Декомпозиция 1)

- **Декомпозиция без потерь**

Обратимая декомпозиция, т.е. имеется возможность собрать исходное отношение из декомпозированных отношений без потери информации (рис. Декомпозиция 2)

СЛУЖАЩИЕ ПРОЕКТЫ				
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	ПРО_НОМ	ПРОЕКТ_РУК
2934	Иванов	22000.00	1	Иванов
2941	Иваненко	22000.00	2	Иваненко

Декомпозиция (1). Отношения СЛУЖ и СЛУ_ПРО

СЛУ_НОМ	СЛУ_ИМЯ	СЛУ_ЗАРП
2934	Иванов	22000.00
2941	Иваненко	22000.00

СЛУ_НОМ	ПРО_НОМ	ПРОЕКТ_РУК
2934	1	Иванов
2941	2	Иваненко

Декомпозиция (2). Отношения СЛУЖ и ЗАРП_ПРО

СЛУ_НОМ	СЛУ_ИМЯ	СЛУ_ЗАРП
2934	Иванов	22000.00
2941	Иваненко	22000.00

СЛУ_ЗАРП	ПРО_НОМ	ПРОЕКТ_РУК
22000.00	1	Иванов
22000.00	2	Иваненко

Нормальные формы

- Первая нормальная форма (1NF)
- Вторая нормальная форма (2NF)
- Третья нормальная форма (3NF)
- Нормальная форма Бойса-Кодда (BCNF)
- Четвертая нормальная форма (4NF)
- Пятая нормальная форма, или нормальная форма проекции-соединения (5NF или PJ/NF)

Свойства нормальных форм

1. Каждая следующая нормальная форма в некотором смысле лучше предыдущей нормальной формы
2. При переходе к следующей нормальной форме свойства предыдущих нормальных форм сохраняются
3. Метод нормализации
Декомпозиция отношения, находящегося в предыдущей нормальной форме, на два и более отношений, которые удовлетворяют требованиям следующей нормальной формы

Первая нормальная форма

НОМЕР_ОТДЕЛА	ОТДЕЛ		
	СОТР_НОМЕР	СОТР_ИМЯ	СОТР_ЗАРП
310	2934	Иванов	112,000
	2935	Петров	112,500
313	2937	Федоров	110,000
315	2938	Иванова	112,000

СОТР_НОМЕР	СОТР_ИМЯ	СОТР_ЗАРП	СОТР_ОТД_НОМЕР
2934	Иванов	112,000	310
2935	Петров	144,000	310
2936	Сидоров	92,000	313
2937	Федоров	110,000	310
2938	Иванова	112,000	315

Значения всех атрибутов должны быть атомарными:

- ✓ **атомарным**, если его смысл — «пин-код кредитной карты»
(при разбиении на части или переупорядочивании смысл теряется)
- ✓ **неатомарным**, если его смысл — «набор цифр»
(при разбиении на части или переупорядочивании смысл **не** теряется)

Аномальные обновления

Под *аномалиями обновления* понимаются трудности, с которыми приходится сталкиваться при выполнении операций добавления кортежей в отношении (INSERT), удаления кортежей (DELETE) и модификации кортежей (UPDATE).

СЛУЖАЩИЕ_ПРОЕКТЫ				
СЛУ_НОМЕР	СЛУ_ИМЯ	СЛУ_ЗАРП	ПРО_НОМ	ПРОЕКТ_РУК
2934	Иванов	22000.00	1	Иванов
2941	Иваненко	22000.00	2	Иваненко

Декомпозиция (1). Отношения СЛУЖ и СЛУ_ПРО

СЛУ_НОМ	СЛУ_ИМЯ	СЛУ_ЗАРП
2934	Иванов	22000.00
2941	Иваненко	22000.00

СЛУ_НОМ	ПРО_НОМ	ПРОЕКТ_РУК
2934	1	Иванов
2941	2	Иваненко

Декомпозиция (2). Отношения СЛУЖ и ЗАРП_ПРО

СЛУ_НОМ	СЛУ_ИМЯ	СЛУ_ЗАРП
2934	Иванов	22000.00
2941	Иваненко	22000.00

СЛУ_ЗАРП	ПРО_НОМ	ПРОЕКТ_РУК
22000.00	1	Иванов
22000.00	2	Иваненко

Аномалии обновления, возникают из-за наличия неминимальных функциональных зависимостей



Возможная декомпозиция

Для преодоления этих трудностей можно произвести декомпозицию *переменной отношения*

СЛУЖАЩИЕ_ПРОЕКТЫ_ЗАДАНИЯ на две *переменных отношений* – СЛУЖ {СЛУ_НОМ, СЛУ_УРОВ, СЛУ_ЗАРП} и СЛУЖ_ПРО_ЗАДАН {СЛУ_НОМ, ПРО_НОМ, СЛУ_ЗАДАН}.

На основании *теоремы Хита* эта декомпозиция является декомпозицией без потерь, поскольку

в исходном отношении имелась FD {СЛУ_НОМ, ПРО_НОМ} → СЛУ_ЗАДАН.

Для преодоления этих трудностей можно произвести декомпозицию *переменной отношения* СЛУЖАЩИЕ_ПРОЕКТЫ_ЗАДАНИЯ на две *переменных отношений* – СЛУЖ {СЛУ_НОМ, СЛУ_УРОВ, СЛУ_ЗАРП} и СЛУЖ_ПРО_ЗАДАН {СЛУ_НОМ, ПРО_НОМ, СЛУ_ЗАДАН}. На основании *теоремы Хита* эта декомпозиция является декомпозицией без потерь, поскольку в исходном отношении имелась FD {СЛУ_НОМ, ПРО_НОМ}->СЛУ_ЗАДАН.

Возможная декомпозиция



Вторая нормальная форма



Переменная отношения находится во второй нормальной форме тогда и только тогда, когда она находится в первой нормальной форме и каждый неключевой атрибут функционально полно зависит от первичного ключа

Третья нормальная форма



Переменная отношения находится в третьей нормальной форме в том только в том случае, когда она находится во второй нормальной форме и каждый неключевой атрибут нетранзитивно функционально зависит от первичного ключа

Процедура нормализации

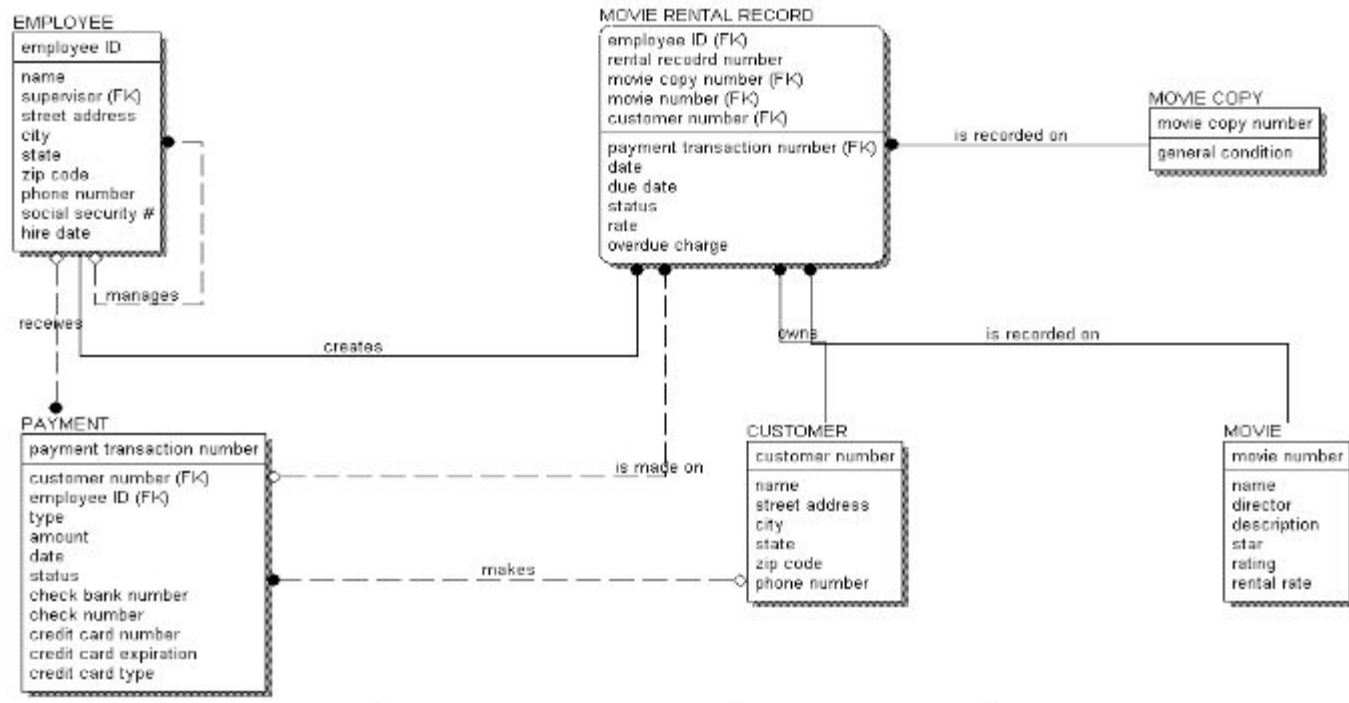
- **Если** таблица **T** имеет составной ключ вида **(K1, K2)** и набор атрибутов **Y**, зависящий только от его части (например, **K2**), **то** создать таблицу со схемой **(K2, Y)**, где **K2** – ключ, а из **T** удалить **Y** и сделать там **K2** внешним ключом.
- **Если** таблица **T** имеет ключ **K** и два неключевых набора атрибутов **X** и **Y**, такие, что **X** зависит от **K**, а **Y** зависит от **X**, **то** следует создать таблицу со схемой **(X, Y)**, где **X** – ключ, а из таблицы **T** удалить **Y** и сделать там **X** внешним ключом.
- **Если** таблица **T** имеет несколько ключей **(K1, K2)** и **(K1, K3)**, и между их частями существуют зависимости (например, **K2 → K3**), **то** атрибуты этих зависимостей следует выделить в отдельную таблицу (со схемой **(K2, K3)**, где **K2** – ключ), а из таблицы **T** удалить **K3** и сделать там **K2** внешним ключом.

Ограниченность реляционной модели

- Модель не обеспечивает достаточных средств для представления смысла данных
- Во многих прикладных областях затруднительно моделировать предметную область на основе плоских таблиц
- Не предоставляются средства для формального представления зависимостей
- Не предоставляется механизма для разделения сущностей и связей

Семантические модели данных (Entity-Relationship)

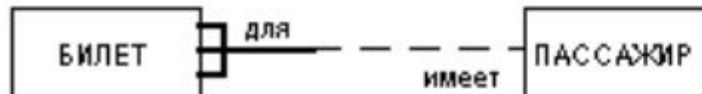
- Главным назначением семантических моделей является обеспечение возможности выражения семантики данных
- Наиболее часто на практике семантическое моделирование используется на первой стадии проектирования базы данных



Сущность. Связь. Атрибут

АЭРОПОРТ
например,
Шереметьево,
Хитроу

Сущность. Реальный или представляемый объект, информация о котором должна сохраняться и быть доступной



Связь. Графически изображаемая ассоциация, устанавливаемая между двумя типами сущностей



Атрибут. Любая деталь, которая служит для уточнения, идентификации, классификации, числовой характеристики или выражения состояния сущности

ЧЕЛОВЕК
пол, например,
М или Ж
год рождения,
1978
фио, например,
Иванов Иван
Иванович