

Введение

Объектные базы данных. Лекция 1

Организация курса. Лекции

- Преподаватель:
Иванова Елена Владимировна
- 1 академических часа в неделю
- Балльно-рейтинговая система
- Зачет
- Веб-страница курса:
<http://foreva.susu.ru/courses/odb/>

Организация курса. Практика

- Преподаватель:
Савченко Дмитрий Игоревич
- 1 академический час в неделю

Определение базы данных

- **База данных (БД)** — совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных (ГОСТ)
- **База данных (БД)** — организованная в соответствии с определенными правилами и поддерживаемая в памяти компьютера совокупность данных, характеризующая актуальное состояние некоторой предметной области и используемая для удовлетворения информационных потребностей пользователей. (Когаловский М.Р.)
- **База данных (БД)** — некоторый набор перманентных (постоянно хранимых) данных, используемых прикладными программными системами какого-либо предприятия (Дейт К.Дж.)

Признаки базы данных

- хранится и обрабатывается в вычислительной системе
- имеет логическую структуру
- имеет схему или метаданные, описывающие логическую структуру БД в формальном виде

Из перечисленных признаков только первый является строгим, а другие допускают различные трактовки и различные степени оценки!

В соответствии с общепринятой практикой не называют базами данных файловые архивы, Интернет-порталы или электронные таблицы

Как данные хранить?

Понятие «модель данных»

Как эффективно
манипулировать данными?

- Понятие модели данных предложено в 1969 г. Эдгаром Коддом для описания реляционного подхода к организации БД. Понятие модели данных оказалось удобным и для реализационно-независимого представления и сопоставления других подходов.
- В классической теории баз данных, **модель данных** есть формальная теория представления и обработки данных в системе управления базами данных.
- **Система управления базами данных (СУБД)** – программный продукт и языковые средства, обеспечивающие управление созданием и использованием баз данных.

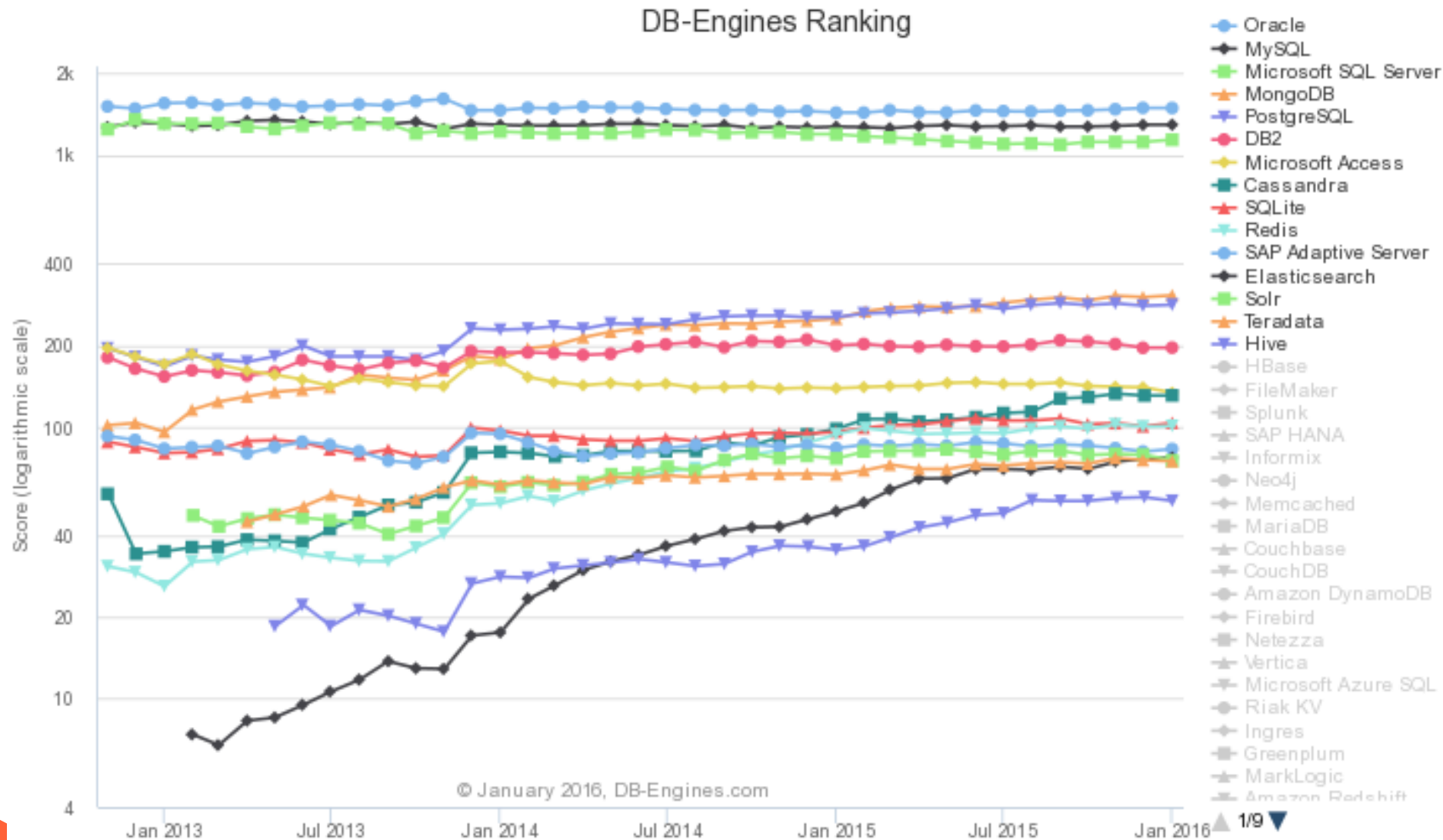
Модели данных

- Иерархическая
- Сетевая
- Реляционная
- Объектно-ориентированная
- Документ-ориентированная
- Хранилища «ключ-значение»
- Графовая
- Столбцовая
- др.

Рейтинг СУБД 2015 издания DB-Engines

Rank			DBMS	Database Model	Score		
Jan 2016	Dec 2015	Jan 2015			Jan 2016	Dec 2015	Jan 2015
1.	1.	1.	Oracle	Relational DBMS	1496.08	-1.47	+56.92
2.	2.	2.	MySQL	Relational DBMS	1299.26	+0.72	+21.75
3.	3.	3.	Microsoft SQL Server	Relational DBMS	1144.06	+20.90	-54.55
4.	4.	↑ 5.	MongoDB +	Document store	306.03	+4.64	+55.13
5.	5.	↓ 4.	PostgreSQL	Relational DBMS	282.40	+2.31	+27.91
6.	6.	6.	DB2	Relational DBMS	196.37	+0.24	-3.76
7.	7.	7.	Microsoft Access	Relational DBMS	134.04	-6.17	-5.10
8.	8.	8.	Cassandra +	Wide column store	130.95	+0.11	+32.20
9.	9.	9.	SQLite	Relational DBMS	103.74	+2.89	+7.54
10.	10.	10.	Redis +	Key-value store	101.16	+0.62	+6.92
11.	11.	11.	SAP Adaptive Server	Relational DBMS	83.18	+1.71	-0.60
12.	↑ 13.	↑ 16.	Elasticsearch	Search engine	77.21	+0.65	+28.17
13.	↓ 12.	↓ 12.	Solr	Search engine	75.39	-3.75	-1.35
14.	14.	↓ 13.	Teradata	Relational DBMS	74.95	-0.77	+7.90
15.	15.	↑ 17.	Hive	Relational DBMS	53.58	-1.69	+18.19
16.	16.	↓ 14.	HBase	Wide column store	53.37	-0.88	-0.22
17.	17.	↓ 15.	FileMaker	Relational DBMS	48.83	-1.29	-2.86
18.	18.	↑ 20.	Splunk	Search engine	43.12	-0.74	+10.05
19.	19.	↑ 21.	SAP HANA	Relational DBMS	38.61	-0.24	+8.71
20.	20.	↓ 18.	Informix	Relational DBMS	34.88	-1.52	+0.06
21.	21.	↑ 23.	Neo4j +	Graph DBMS	33.01	-0.18	+8.58
22.	22.	↓ 19.	Memcached	Key-value store	29.97	-0.96	-4.40
23.	23.	↑ 27.	MariaDB +	Relational DBMS	27.76	+0.02	+10.34
24.	24.	24.	Couchbase +	Document store	26.09	-0.17	+3.50
25.	25.	↓ 22.	CouchDB	Document store	24.12	-0.92	-2.20

Рейтинг СУБД 2015 издания DB-Engines



реляционные СУБД
используют 99,5% респондентов

Рейтинг СУБД 2016

- | | |
|--------------------|--------------------------|
| 1. MySQL | реляционная |
| 2. PostgreSQL | реляционная |
| 3. MS SQL Server | реляционная |
| 4. MongoDB | документ-ориентированная |
| 5. SQLite | реляционная |
| 6. Oracle Database | реляционная |
| 7. Firebird | реляционная |
| 8. CouchDB | документ-ориентированная |
| DB2 | реляционная |
| 9. MariaDB | реляционная |
| 10. RavenDB | документ-ориентированная |
| Redis | хранилище ключ-значение |
| SAP ASE | реляционная |



Реляционная модель данных

Использование реляционных баз данных было предложено Коддом из компании IBM в 1970 году.

- База данных состоит из **таблиц (отношений)**
- Колонки – **атрибуты** таблицы
- Строки – **кортежи** таблицы

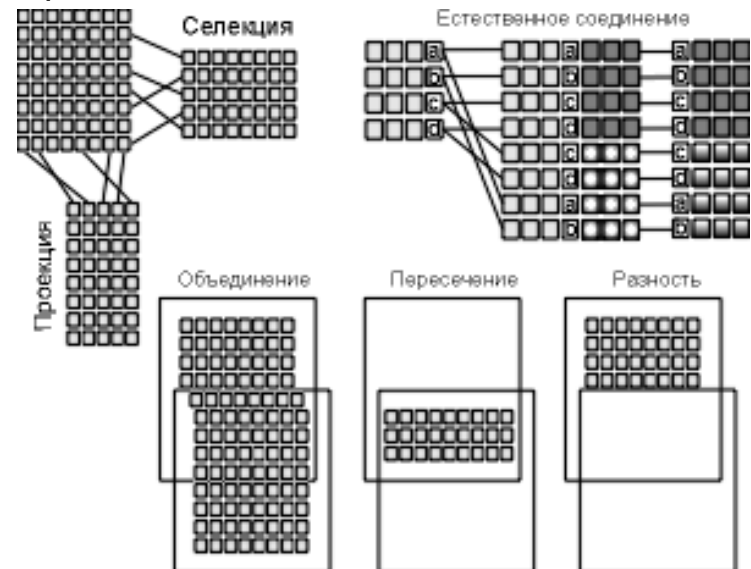
Сотрудники

Таб номер	ФИО	Пол	Дата рождения	Должность
003	Иванов И.И.	М	04.12.1989	прораб
123	Петров П.П.	М	14.05.1986	бухгалтер
563	Сидорова С.С.	Ж	23.02.1974	гл. бухгалтер
432	Антонова А.А.	Ж	17.06.1955	директор
111	Федоров Ф.Ф.	М	22.04.1964	зам. директора

Реляционная модель данных

- Предложив реляционную модель данных, Э.Ф.Кодд создал и инструмент для удобной работы с отношениями – **реляционную алгебру**. Каждая операция этой алгебры использует одну или несколько таблиц в качестве ее операндов и продуцирует в результате новую таблицу.
- Созданы языки манипулирования данными, позволяющие реализовать все операции реляционной алгебры.

- **SQL (Structured Query Language)** – *структуризованный язык запросов*



Объектно-ориентированные СУБД

- Проблема "несоответствия импеданса" (*impedance mismatch*) – кардинальные различия между прикладной программой и данными, хранимыми в реляционной базе данных.
 - Отсутствие неатомарных типов данных.
 - 1НФ запрещает «вложенные» таблицы.
 - Отсутствие пользовательских типов данных.
 - Нельзя определить новый тип данных и его операции на базе имеющихся типов и операций.
 - Отсутствие типов данных со сложной структурой
 - Структуры (записи), коллекции (множество, массив, список и др.), данные САПР и ГИС, документ-данные.
 - Отсутствие поддержки концепций ООП (объект, инкапсуляция, наследование, полиморфизм).

ODMG

- ODMG (Object Data Management Group) - консорциум поставщиков ООБД и других заинтересованных организаций, созданный в 1991 г. Его задачей является разработка стандарта на хранение объектов в базах данных. В настоящее время опубликован вторая версия стандарта, которую так и называют ODMG 2.0. Рассмотрим кратко основные положения этого документа.
- Стандарт на хранение объектов ODMG 2.0 разработан на основе трех стандартов:
 - управление базами данных (SQL),
 - объекты (стандарты OMG - Object Management Group),
 - стандарты на объектно-ориентированные языки программирования (C++, Smalltalk, Java).

ODMG

- ODMG добавляет возможности взаимодействия с базами данных в объектно-ориентированные языки программирования: определяются средства долговременного хранения объектов и расширяется семантика языка, вносятся операторы управления данными. Стандарт состоит из нескольких частей:
- **Объектная модель** - унифицированная основа всего стандарта. Она расширяет объектную модель консорциума OMG за счет введения таких свойств как связи и транзакции для обеспечения функциональности, требуемой при взаимодействии с базами данных.
- **Язык описания объектов** (ODL - Object Definition Language) - средство определения схемы базы данных (по аналогии с DDL в реляционных СУБД).
- **Язык объектных запросов** (OQL - Object Query Language) - SQL - подобный декларативный язык, который предоставляет эффективные средства для извлечения объектов из базы данных, включая высокоуровневые примитивы для наборов объектов и объектных структур.
- **Связывание с ОО-языками.** Стандарт связывания с C++, Smalltalk и Java определяет Object Manipulation Language (OML) - язык манипулирования объектами, который расширяет базовые ОО-языки средствами манипулирования и хранения объектов. Также включаются OQL, средства навигации и поддержка транзакций. Каждый ОО-язык имеет свой собственный OML, поэтому разработчик остается в одной языковой среде, ему нет необходимости разделять средства программирования и доступа к данным.

Объектные расширения реляционных СУБД.

Язык SQL-3

- Попытки совместить средства манипулирования данными реляционной модели и способы описания внешнего мира объектно-ориентированной модели получили развитие в языке SQL-3.
- Разработчики SQL-3 считают, что характеристики объекта определяется описанием строки таблицы. Поэтому, вводится специальная возможность описания нового типа данных:

Create type Address (

 number char (6),

 street char (30),

 aptno integer,

 city char (30),

 state char (2),

 zip integer

);

Объектные расширения реляционных СУБД. Язык SQL-3

- На основе нового типа могут быть определены таблицы, например:
Create table Addresses of Address;
- Новые типы допускается использовать и для определения столбцов (т.е. игнорируется требование атомарности атрибутов реляционной модели):

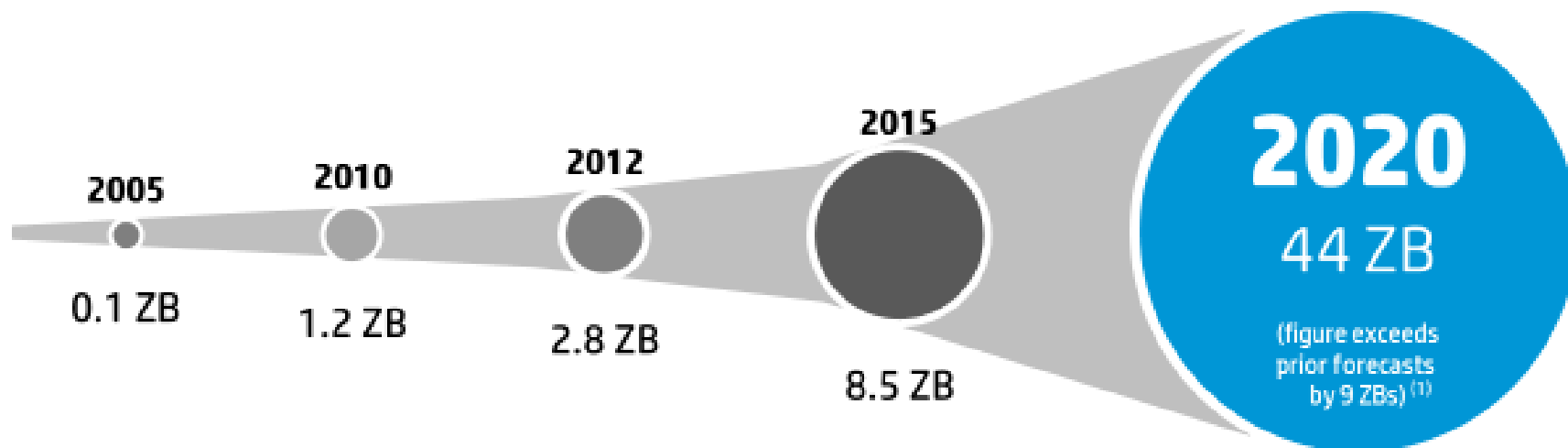
```
Create table People of new type Person (  
    name char (30),  
    address Address,  
    birthdate date,  
);
```

- Наследование определяется с помощью фразы under.

ORM

- ORM (англ. Object-Relational Mapping, рус. объектно-реляционное отображение) — технология программирования, которая связывает базы данных с концепциями объектно-ориентированных языков программирования, создавая «виртуальную объектную базу данных».
- С точки зрения программиста система должна выглядеть как постоянное хранилище объектов. Он может просто создавать объекты и работать с ними как обычно, а они автоматически будут сохраняться в реляционной базе данных.
- На практике всё не так просто и очевидно. Все системы ORM обычно проявляют себя в том или ином виде, уменьшая в некотором роде возможность игнорирования базы данных. Более того, слой транзакций может быть медленным и неэффективным (особенно в терминах сгенерированного SQL). Все это может привести к тому, что программы будут работать медленнее и использовать больше памяти, чем программы, написанные «вручную».

Проблема больших данных



- Объем хранимой информации удваивается каждые два года
- из всего объема существующих данных потенциально полезны 22%, из которых менее 5% были подвергнуты анализу

Проблема больших данных

- Нью-Йоркская фондовая биржа генерирует около ***терабайта данных в день***
- Объем хранилища социальной сети Facebook ***каждый день увеличивается на 500 терабайт***
- Проект Internet Archive уже хранит 2 петабайта данных и прирастает ***20 терабайтами в месяц***
- Эксперименты на Большом адронном коллайдере генерируют около ***50 терабайт данных в сутки***

Что такое «большие данные»?

- «Большие данные» характеризуются объемом, разнообразием и скоростью, с которой структурированные и неструктурированные данные поступают по сетям передачи в процессоры и хранилища, наряду с процессами преобразования этих данных в ценную для бизнеса информацию (Исследовательская компания Gartner)
- Характеристики больших данных:
 - **Объем**
 - **Разнообразие** (неструктурированность данных)
 - **Скорость** (поступление и извлечение данных)
 - [Ценность]

Недостатки реляционной модели

- ACID свойства (атомарность, согласованность, изолированность, долговечность) не позволяют наращивать производительность реляционных систем

Решение проблемы производительности реляционных СУБД

- Использовать более мощное оборудование (вертикальное масштабирование)
- Оптимизировать запросы, проанализировав планы их исполнения, и создать дополнительные индексы.
- Денормализация схемы БД
- **noSQL решения**
- **newSQL решения**

noSQL решения

- Термин «NoSQL» впервые был использован в 1998 году для описания реляционной базы данных, не использовавшей SQL
- Популярность NoSQL стал набирать в 2009 г, в связи с появлением большого количества веб-стартапов, для которых важнейшей задачей является поддержание постоянной высокой пропускной способности хранилища при неограниченном увеличении объема данных.

Классификация NoSQL решений

- **Хранилища ключ-значение.** Отличительной особенностью является простая модель данных — ассоциативный массив или словарь, позволяющий работать с данными по ключу. Основная задача подобных хранилищ — максимальная производительность, поэтому никакая информации о структуре значений не сохраняется.
- **Документ-ориентированные хранилища.** Модель данных подобных хранилищ позволяет объединять множество пар ключ-значение в абстракцию, называемую «документ». Документы могут иметь вложенную структуру и объединяться в коллекции. Однако это скорее удобный способ логического объединения, т.к. никакой жесткой схемы у документов нет и множества пар ключ-значение, даже в рамках одной коллекции, могут быть абсолютно произвольными. Работа с документами производится по ключу, однако существуют решения, позволяющие осуществлять запросы по значениям атрибутов.

Классификация NoSQL решений

- **Колоночные хранилища.** Этот тип кажется наиболее схожим с традиционными реляционными СУБД. Модель данных хранилищ подобного типа подразумевает хранение значений как неинтерпретируемых байтовых массивов, адресуемых кортежами <ключ строки, ключ столбца, метка времени>. Основой модели данных является колонка, число колонок для одной таблицы может быть неограниченным. Колонки по ключам объединяются в семейства, обладающие определенным набором свойств.
- **Хранилища на графах.** Подобные хранилища применяются для работы с данными, которые естественным образом представляются графами (например, социальная сеть). Модель данных состоит из вершин, ребер и свойств. Работа с данными осуществляется путем обхода графа по ребрам с заданными свойствами.

newSQL решения

- **newSQL** - класс современных реляционных СУБД, стремящихся совместить в себе преимущества NoSQL и транзакционные требования классических баз данных.
- Термин был предложен в 2011 году Мэтью Аслетом.
- Примеры СУБД:
 - VoltDB
 - MemSQL
 - SAP HANA
 - OrientDB