**Базы данных и SQL**

База данных (БД) – структурированное хранилище информации, организованные таким образом, чтобы обеспечить хранение, управление и доступ к информации. Чаще всего эти данные взаимосвязаны. Сама по себе БД – это просто хранилище данных, остальные свойства базы данных обеспечиваются за счет системы управления базой данных.

Система управления базой данных (СУБД) – программные средства для управления данными. Отвечает за поддержку языка БД (обеспечивает интерфейс для взаимодействия с данными), за механизм хранения и извлечения данных, за оптимизацию процессов извлечения данных.

**Виды БД:**

*Без использования СУБД*

|  |  |
| --- | --- |
| Файловые | Данные хранятся в виде текста в файле. |
| Иерархические | Имеет древовидную структуру (узел может иметь только одного родителя) Пример: файловая система. |
| Сетевые | Имеет структуру графа (узел может иметь более одного родителя) |

*С использованием СУБД*

|  |  |
| --- | --- |
| Реляционные | Структура в виде таблиц, связанных между собой |
| NoSQL | Общее название для нереляционных БД. |

**Типы СУБД:**

NoSQL (not only SQL) – Нереляционные СУБД, в которых данные могут быть организованы в виде иных структур, отличных от таблиц, ориентированных под свои задачи.

Используется язык отличный от SQL.

Отсутствие жесткой схемы данных (Приятное следствие отсутствия схемы — эффективность работы с разреженными (sparse) данными. Если в одном документе есть поле date\_published, а во втором — нет, значит никакого пустого поля date\_published для второго создано не будет)

Отказ об атомарности и согласованности данных в пользу горизонтального масштабирования.

1) Ключ-значение – база данных организована в виде словаря. Некая таблица, имеющая уникальный ключ и соответствующее ему значение. Подходит для кэширования. [Redis]

Высокая скорость, легкое масштабирование. Не подходит для сложных структур (сущности с разными типами данных) и сложных запросов.

2) Колоночные – Данные организованы в виде таблиц так же, как и в реляционной СУБД, но данные организованы не построчно (для чтения колонки придется вычитать всю строку до нужной колонки), а «поколоночно», т.е. колонка — это отдельная структура, в которой содержатся данные для всех строк по определенному ключу.

Подходит для аналитических запросов на больших объемах данных (эффективна при запросах для более 100 миллионов строк) – OLAP.

3) Ширококолоночные -

В отличие от реляционных не имеют жесткой схемы, то есть, хранит только имеющиеся данные (см. рис.2) [HBase, Cassandra]

За счет своей структуры

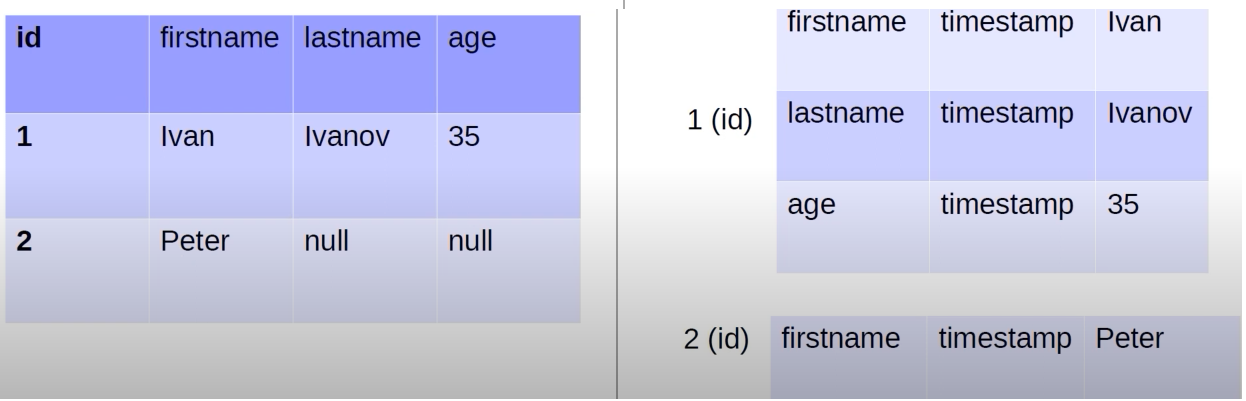
позволяет менять схему таблиц – не требует перестройки всей таблицы и не хранить null значения, лучшее сжатие. Cassandra – легко горизонтально масштабируема

Плавающая скорость записи – зависит от архитектурных особенностей и методов записи, но в среднем ниже, чем у реляционных

Общая структура:



Реляционная СУБД Колоночная СУБД



4) Документо-ориентированная – Данные организованы в виде коллекций (аналог таблицы), которые в свою очередь содержат документы (аналог строки в таблице). Документ представлен парами ключ-значение, имеет формат JSON или BSON

Гибкий – не требует схем данных, данные хранятся в одной сущности – не требует соединений, как в реляционной, горизонтально масштабируемый

5) Графовые -

реляционная алгебра

Файл-серверные, клиент-серверные, встраиваемые субд????

Теорема брюйера

ACID

OLAP