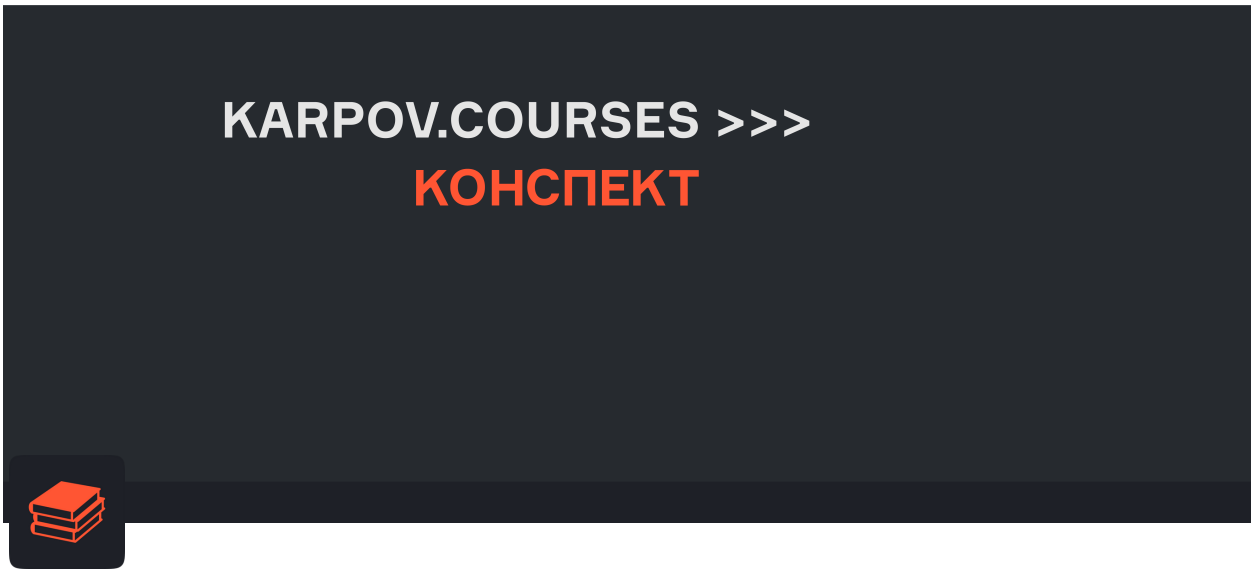


> Конспект > 1 урок > Реляционные и MPP Базы данных. Что и как в них хранить

> Где и как хранить много однотипной информации?

Задачи:

- Сохранить много данных
- Быстро добавлять новые данные
- Быстро выполнять расчеты над данными и получать результаты
- Гарантировать целостность и непротиворечивость данных
- Иметь возможность восстановиться после сбоя
- Построить КХД

Способы решения:

- Записать все на бумагу и создать несколько копий
- Сохранить все в файлы на диске
- Записать данные в библиотеку магнитных лент
- Использовать Excel
- Использовать Базы Данных и Системы Управления Базами Данных

> Базы Данных и Системы Управления Базами Данных

> Базы Данных (БД)

База данных - это организованная коллекция данных хранящаяся и доступная в электронном виде при помощи вычислительных машин.

Элементы базы данных:

- Модель
- Логическая структура
- Объекты БД

- Язык определения данных DDL (Data Definition Language)
- Язык изменения данных DML (Data Manipulation Language)

> Система Управления Базами Данных (СУБД)

Система Управления Базами Данных - комплекс программ, позволяющих создать БД и манипулировать данными.

СУБД обеспечивает:

- Безопасность
- Надежность хранения
- Целостность данных

Основные функции:

- Управление данными на внешних дисках
- Управлние данными в оперативной памяти
- Журнализация изменений
- Резервное копирование
- Восстановление БД после сбоев
- Поддержание языков БД DDL и DML

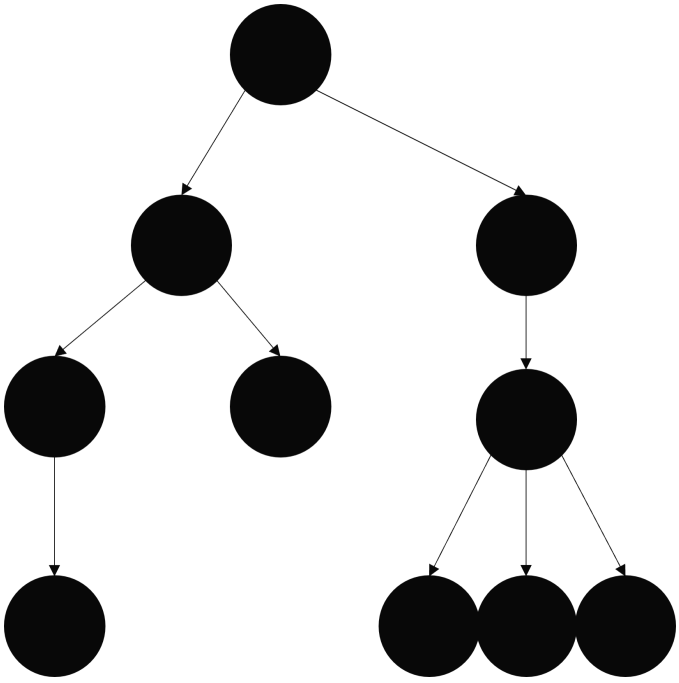
Компоненты СУБД:

- Ядро. Управляет всеми процессами находящимися в БД и управляет всеми приложениями входящими в ее состав.
- Процессор языка БД (оптимизатор). Интерпретирует язык запроса пользователя к БД. Также строит план выполнения запроса и возращает его результат.
- Подсистема поддержки времени выполнения. Позволяет выполнять все пользовательские запросы и отдавать результаты. Кроме этого позволяет пользователям работать параллельно.
- Сервисные программы. Набор дополнительных компонентов, которые расширают функционал БД. Например, функция резервного копирования.

> Модели Баз Данных

> Иерархическая

Элементы организованы в структуры, связанные между собой иерархическими или древовидными связями. Родительский элемент может иметь несколько дочерних элементов. Но у дочернего элемента может быть только один предок.



Особенности:

- Простая для понимания структура
- "Дешевая" навигация по узлам и связям
- Жесткая структура

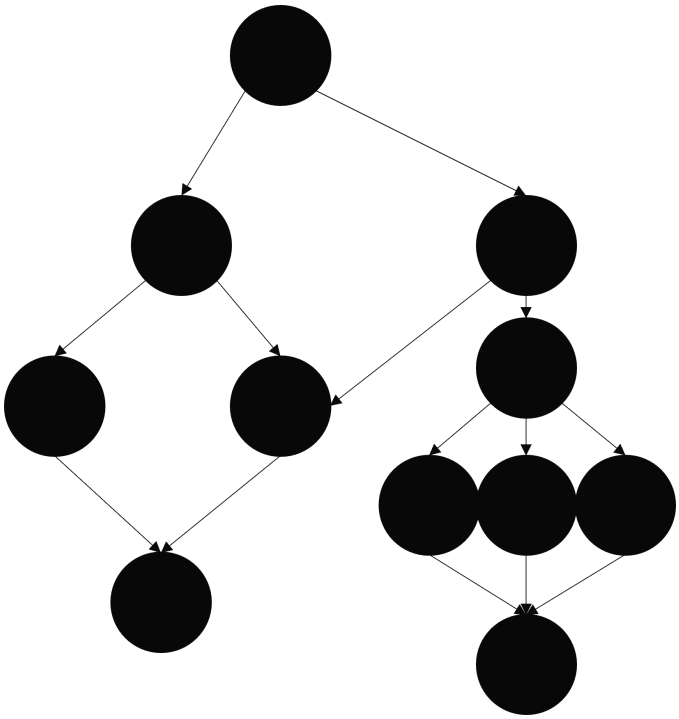
Примеры:

- IMS
- Windows Registry
- TDMS
- Cache

- Файловые системы

> Сетевая

Модель сетевой базы данных позволяет каждой записи иметь несколько родителей и несколько дочерних записей, которые, когда они визуализируются, принимают форму сетевой структуры сетевых записей.



- Простая для понимания структура
- Эффективные вычисления
- Сложна для изменений

Примеры:

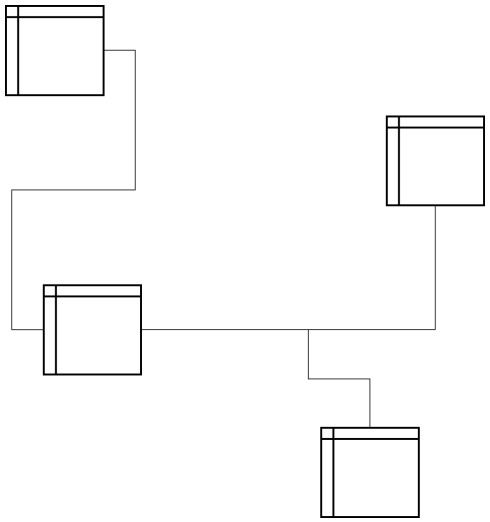
- IDS, IDS/2
- IDMS (Cullinet)
- DMS-1100, DMS-90 (UNIVAC)
- DBMS-10 (DEC)
- IMAGE/3000 (HP)

- UDS (Siemens)

> Реляционная

В реляционной модели, как объекты, так и их отношения представлены только таблицами. Основана на реляционной алгебре и включает в себя:

- Отношения/Relations
- Поля/Колонки/Атрибуты/Columns
- Строки/Rows/Tuples



Свойства ACID:

Atomicity - Атомарность. Если мы что-то запустили (транзакции), то они либо все выполняются, либо все не выполняются.

Consistency - Согласованность. Все изменения в БД приводят ее к согласованности, т.е. все измененные данные в одной таблице и связанные с другими таблицами будут меняться вместе и связно. Зафиксируются только допустимые результаты.

Isolation - Изолированность. Все тразакции изолированы друг от друга. В момент когда выполняется какая-то транзакция последовательно меняющая данные в нескольких объектах таблиц, никакая другая транзакция не может увидеть изменения пока они не завершились. Все транзакции видят согласованное состояние БД.

Durability - Устойчивость. Позволяет БД оставаться согласованной в момент сбоя.

Достоинства:

- Простота и доступность для понимания конечным пользователям
- Применение математического аппарата реляционной алгебры
- Полная независимость данных
- Изоляция физической структуры от логической
- SQL
- ACID
- Идеально для КХД

Недостатки:

- "Дорогой" доступ к данным
- Большое кол-во отношений
- Не все данные "красиво" зайдут в реляционную модель
- Высокая стоимость владения для Корпоративных решений

> NoSQL

- Графовые - вершины, ребра и их свойства.
- Объектно-ориентированные
- Ключ-значение - являются по сути словарем, позволяющим извлечь однозначное значение по ключу.
- Семейство столбцов - данные хранятся по столбцам, а не по строкам.
- Документые - похожи на ключ-значение, только значения с разметкой (XML, JSON), которая и образует "документ".

> PostgreSQL

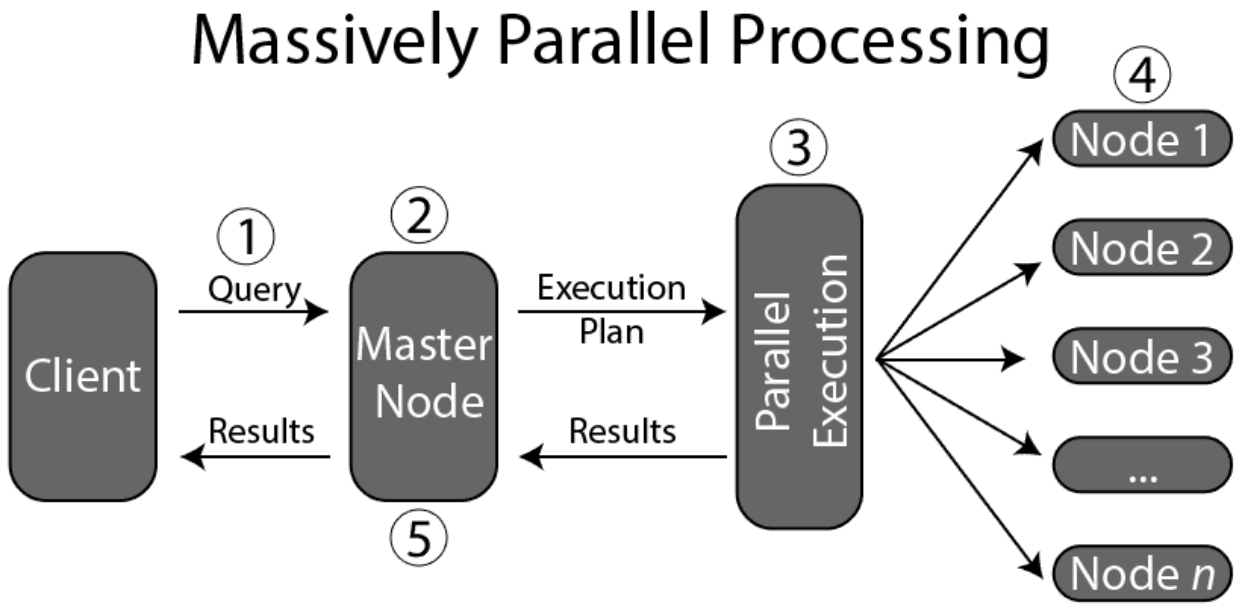
- OpenSource проект
- Динамически развивается и исправляется

- Поддержка любого размера БД
- Высокая совместимость с SQL2016
- Доступны несколько вариантов кластеров и реплик
- Поддержка дополнительных языков
- Встроенная поддержка SSL

> MPP

MPP (Massive Parallel Processing) или Массово-параллельная архитектура - это параллельные вычисления на нескольких серверах объединенных в один кластер.

- Данные распределены по нескольким row / column store сегментам кластера
- Обработка данных как можно ближе к месту их хранения
- Мощный интерконнект для обмена порциями данных
- Shared Nothing архетиктура
- Идеально для OLAP / DWH



Достоинства:

- Быстрота обработки больших объемов данных
- SQL Conformance
- Легкое горизонательное масштабирование
- Отказоустойчивость за счет создания релик/зеркал
- Стандартное Hardware для сегментов
- Row / Column store
- Сжатие хранимых данных
- Параллельная загрузка частей данных напрямую в шарды
- Cloud Compatibility. Совместимость с облачными решениями.

Недостатки:

- Высокие требования к элементам инфраструктуры
- Низкая производительность на OLTP профиле нагрузки
- Ограничения в поддержке всех функций SQL
- Возможности возникновения перекосов данных
- Специфическая поддержка и обслуживание

> GreenPlum

- OpenSource проект
- Динамически развивается и исправляется
- Высокая совместимость с PostgreSQL
- Поддержка разных языков (Python, R, C)
- Поддержка SSL
- Легкая масштабируемость

> Глоссарий

Шардинг — метод разделения и хранения единого логического набора данных в виде множества баз данных. Другое определение шардинга — горизонтальное разделение данных.

Нода - объект в базе данных, узел графа (например, сервер).

Кластер - это группа серверов (именуемых "нодами"), которые работают вместе, выполняют общие задачи и клиенты видят их как одну систему.

Репликация - поддержания двух (или более) наборов данных в согласованном состоянии.

DDL (Data Definition Language) - это группа операторов определения данных. Другими словами, с помощью операторов, входящих в эту группы, мы определяем структуру базы данных и работаем с объектами этой базы, т.е. создаем, изменяем и удаляем их.

DML (Data Manipulation Language) - это группа операторов для манипуляции данными. С помощью этих операторов мы можем добавлять, изменять, удалять и выгружать данные из базы, т.е. манипулировать ими.