**NOSql**

Термин NoSQL иногда понимают как Not SQL, но более правильно считать - Not Only SQL. Из-за большого разнообразия СУБД невозможно дать общую характеристику этого направления.

Отметим некоторые общие, но не обязательные черты:

1. Не реляционные модели данных.
2. Работа на кластерах серверов и возможности масштабирования по горизонтали.
3. Высокая степень доступности данных за счёт некоторого пренебрежения согласованностью данных.
4. как правило, не поддерживают ACID транзакции, хотя есть исключения.
5. использование двухшаговой технологии MapReduce.
   1. Первый шаг Map - master node распределяет задачу по остальным узлам, данные обрабатываются.
   2. Второй шаг Reduce - master node получает промежуточные результаты и формирует итог.

Основное преимущество технологии - возможность продолжать работу при сбоях; понятно, что использовать её следует в первую очередь когда не нужна идеальная точность результата.

Мы изучим две реализации NoSQL баз данных: документную СУБД MongoDB и графовую СУБД Neo4J.

**Разновидности моделей NoSQL**

1. **Ключ-значение (key/value)**

Модель данных: коллекция пар ключ-значение. Значение есть текстовый блок которые можно записать прочитать или удалить. Обеспечивают горизонтальное масштабирование, недостижимое для других моделей. Хороша для обработки постоянного потока небольших операций чтения и записи. Быстрая.

Примеры применения: игровые, рекламные приложения и интернет вещей (internet of things, IoT). Amazon DynamoDB обеспечивает задержку не более нескольких миллисекунд при любом масштабе.

Типичные представители -- Memcashed, Riak.

1. **Колоночные базы данных.**

Улучшенная версия хранилища «ключ-значение».

Модель данных: двумерный массив, в котором запись (ключ) содержит одну или несколько привязанных пар «ключ-значение. Хранят большое количество сущностей с многими атрибутами. Данные каждой колонки хранятся в отдельном файле. За счёт этого улучшается сжатие данных. Запросы ускоряются, если в запросе используются только немногие колонки. MapReduce.

Типичные представители — это HBase, Cassandra, Vertica. У Vertica имеется SQL интерфейс.

1. **Документ-ориентированная модель**.

Модель данных: коллекции документов, которые содержат коллекции ключ-значение и являются полуструктурированными. Иерархии сложнее чем у первых двух типов моделей. Предусматривается работа с компонентами документов.

Примеры применения: Каталоги и системы управления контентом, где каждый документ уникален и изменяется со временем.

Типичные представители — MongoDB, CouchDB.

Сравним. Слева документная модель, справа SQL.



1. **Графовые модели.**

Модель данных: узлы и отношения, которые могут содержать пары ключ-значение. Хорошо справляется со сложными проблемами на графах. Быстро работает с графами.

Примеры применения: социальные сети, сервисы рекомендаций, системы выявления мошенничества, геопространственная информация и графы знаний.

Типичные представители — Neo4J, InfoGrid.

1. **Нативные XML модели.**

Модель данных: в отличие от табличных моделей с XML опциями документы содержатся в естественных иерархических структурах.

Примеры применения: медицина, сайты с контентом, генерируемым на основе получаемых данных.

Типичные представители — BaseX, Sedna, eXistdb.

1. **Мультимодельные базы данных.**

Включают несколько моделей, в том числе и табличную.

**Преимущества и недостатки NoSQL баз данных**

**Преимущества:**

* Линейная масштабируемость – добавление узлов в кластер увеличивает производительность.
* Возможность работы с полуструктирированными данными, в том числе выполнение полнотекстового поиска.
* Возможность работы без задания схемы данных.
* Возможность использования репликации данных и шардинга – автоматического распределения данных по узлам сети, когда каждый сервер кластера отвечает за определенную часть информации. Это увеличивает скорость обработки данных, но достигается за счёт смягчения требований ACID.
* Повышение производительности за счет выбора моделей данных и шаблонов доступа;

**Недостатки:**

* ограниченные возможности встроенных языков запросов. Существуют средства трансляции SQL-выражений в код некоторых СУБД.
* сложности в поддержке ACID-требований к транзакциям из-за того, что модели NoSQL соответствуют не CAP-модели (согласованность, доступность, устойчивость к разделению), а модели BASE (базовая доступность, гибкое состояние и итоговая согласованность).
* сильная привязка приложения к конкретной СУБД из-за специфики внутреннего языка запросов и гибкой модели данных, ориентированной на конкретный случай.

***Задания для аудиторной работы***

В каждом варианте задания описаны требования, предъявляемые к проектируемой базе данных. Дайте выразительные имена вершинам, ребрам и свойствам, в которые могла бы заноситься указанная информация.

1. В базе данных должны записываться даты начала и завершения каждого восхождения, имена и адреса участвовавших в нем альпинистов, название и высота горы, страна и район, где эта гора расположена.
2. Базу данных использует для работы коллектив врачей. В таблицы должны быть занесены имя, пол, дата рождения и домашний адрес каждого их пациента. Всякий раз, когда врач осматривает больного, явившегося к нему на прием, или сам приходит к нему на дом, он записывает дату и место, где проводится осмотр, симптомы, диагноз и предписания больному, проставляет имя пациента, а также свое имя. Если врач прописывает больному какое-либо лекарство, в таблицу заносится название лекарства, способ его приема, словесное описание предполагаемого действия и возможных побочных эффектов.
3. В базе хранятся имена, адреса, домашние и служебные телефоны всех членов Думы. В Думе работает порядка сорока комиссий, все участники которых являются членами Думы. Каждая комиссия имеет свой профиль, например, вопросы образования, проблемы, связанные с жильем и так далее. Данные по каждой из комиссий включают: ее нынешний состав и председатель, прежние председатели и члены этой комиссии, участвовавшие в ее работе за прошедшие 10 лет, даты включения и выхода из состава комиссии, избрания ее председателей. Члены Думы могут заседать в нескольких комиссиях. В базу заносятся время и место проведения каждого заседания комиссии с указанием депутатов и служащих Думы, которые участвуют в его организации.
4. Фирме принадлежит небольшая флотилия рыболовных катеров. Каждый катер имеет «паспорт», куда занесены его название, тип, водоизмещение и дата постройки. Фирма регистрирует каждый выход на лов, записывая название катера, имена и адреса членов команды с указанием их должностей (капитан, боцман и т.д.), даты выхода и возвращения, а также вес пойманной рыбы отдельно по сортам (например, трески). За время одного рейса катер может посетить несколько банок. Фиксируется дата прихода на каждую банку и дата отплытия, качество выловленной рыбы (отличное, хорошее, плохое). На борту улов не взвешивается.
5. Разработать информационную систему обслуживания библиотеки, которая содержит следующую информацию: названия книг, ФИО авторов, наименования издательств, год издания, количество страниц, количество иллюстраций, стоимость, название филиала библиотеки или книгохранилища, в которых находится книга, количество имеющихся в библиотеке экземпляров конкретной книги, количество студентов, которым выдавалась конкретная книга, названия факультетов, в учебном процессе которых используется указанная книга.