



LSM-tree

В отличие от индексов реляционных БД, в которых используется B-tree и hash алгоритмы, особенностью NoSQL БД является возможность использования и LSM-tree алгоритмов. LSM (Log-Structured Merge) Tree состоит из двух основных компонентов: входной буфер (memtable) и набор файлов сегментов (sstable).

Входной буфер - это память, в которой данные хранятся во время их записи в БД. Когда новые элементы данных добавляются в LSM Tree, они сначала записываются в журнал (log), который хранится в оперативной памяти или на диске. Это позволяет ускорить операции записи, так как запись в журнал происходит гораздо быстрее, чем запись в файл на диске. Когда он заполняется, его содержимое сливается (merge) в один или несколько сегментов. Эти сегменты содержат отсортированные данные в порядке ключей. После создания сегменты сохраняются на диске и используются для последующего чтения данных. Когда количество сегментов становится слишком большим, они снова объединяются в один сегмент для оптимизации производительности. При поиске данных в LSM Tree все данные объединяются в единую структуру, что позволяет быстро и эффективно искать данные.

LSM Tree - это широко используемая структура данных в различных системах управления базами данных, таких как Apache Cassandra, LevelDB и RocksDB. В Clickhouse реализован аналогичный алгоритм MergeTree, но оно не содержит «memtable» и «log», вставленные данные записываются непосредственно в файловую систему.

NewSQL

NewSQL - это класс баз данных, который объединяет преимущества NoSQL и RDBMS (реляционных СУБД) в единой системе. NewSQL СУБД тоже предназначены для обработки больших объемов данных, с высокой доступностью, масштабируемостью и производительностью.

NewSQL обычно используется для обработки структурированных данных, когда необходима полная ACID-совместимость и более жесткие требования к надежности и безопасности, в то время как NoSQL используется для обработки неструктурированных данных, где гибкость и высокая масштабируемость являются более важными факторами.

При этом, существует несколько причин, почему NoSQL базы данных не теряют своей актуальности, несмотря на появление NewSQL:

1. Гибкость и масштабируемость: NoSQL базы данных имеют гибкую схему данных и могут хранить данные различных типов. Эти преимущества делают их идеальными для хранения неструктурированных данных, таких как документы, видео и изображения.
2. Производительность: NoSQL базы данных используют индексы и кэширование, чтобы обеспечить быстрый доступ к данным. Они также могут использовать параллельную обработку данных и распределение нагрузки. Это преимущество делает их лучшим решением, например, для обработки больших объемов данных в режиме реального времени.
3. Низкая стоимость: NoSQL базы данных обычно бесплатны или имеют низкую стоимость по сравнению с реляционными базами данных.

Таким образом, выбор зависит от конкретных потребностей бизнеса

DuckDB

Помимо упомянутых, существуют также виды БД, которые сложнее отнести к определенному типу.



DuckDB — это SQL СУБД с открытым исходным кодом, предназначенная для обеспечения быстрой, эффективной и надежной обработки данных для аналитических рабочих нагрузок. Она написана на C++ и доступна под лицензией MIT.

Одной из ключевых особенностей DuckDB является ее способность обрабатывать большие объемы данных в памяти, не требуя специализированного оборудования или распределенной архитектуры. Это делает ее идеальным для использования во встроенных приложениях, таких как устройства IoT, или приложениях в памяти, таких как анализ данных и машинное обучение.

DuckDB спроектирована так, чтобы быть высокопроизводительным, с упором на эффективную обработку запросов и поиск данных с малой задержкой. Она использует колоночный формат хранения и включает ряд оптимизаций для повышения производительности, таких как SIMD-векторизация, алгоритмы с поддержкой кэширования (cache-aware) и параллельная обработка.

Еще одной важной особенностью DuckDB является поддержка стандартного SQL, включая поддержку составных типов данных, таких как массивы и JSON. Она также включает в себя ряд дополнительных возможностей, таких как создание кастомных функций, оконные функции, а также поддержка транзакций и ACID-совместимое управление данными.

DuckDB в первую очередь предназначена для рабочих нагрузок OLAP. Хотя DuckDB поддерживает обработку транзакций и обеспечивает управление данными, совместимое с ACID.

DuckDB не классифицируется как NoSQL или NewSQL. DuckDB придерживается реляционной модели данных и поддерживает запросы SQL, что отличает от NoSQL. DuckDB также не относится к категории NewSQL, потому что не фокусируется конкретно на распределенных архитектурах или горизонтальной масштабируемости.

Как вам урок?



Далее >

Слёрм ©

[+7 \(495\) 248-05-80](tel:+74952480580)

[Лицензия №ДЛ-1368 от 22.08.2019](#)

[Политика конфиденциальности](#)

[Публичная оферта](#)

