Tarantool Column Store

Описание технической архитектуры

Версия 1.0.0 от 24.08.23

Оглавление

Термины и сокращения

В настоящем документе используются сокращения, перечисленные в таблице ниже.

Сокращение	Расшифровка
БД	База данных
ВМ	Виртуальная машина
ОС	Операционная система
ПО	Программное обеспечение
СУБД	Система управления базами данных
API	Application Programming Interface - Аппаратно-программный интерфейс. Описание способов (набор классов, процедур, функций, структур или констант), которыми одна компьютерная программа может взаимодействовать с другой программой
JavaScript	Мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили
JSON	JavaScript Object Notation – текстовый формат обмена данными, основанный на JavaScript
Доменная сущность	Объект, обладающий определённым смыслом (например, Лицо) и описываемый набором атрибутов
Атрибут	Именованная количественная или качественная характеристика (напр. “Адрес IP”)
Операция	Единичное действие, связанное с одной или несколькими Доменными Сущностями, описанное набором атрибутов
Матрица	Контейнер исторических данных об Операциях с колоночным видом хранения и представления данных
Индексная колонка	Колонка из матрицы, для которой строится индекс (например, ip-адрес или страна)
Счетчик	Рассчитываемое значение агрегата одной колонки в зависимости от условий
PreWhere	Условие фильтрации, которое использует индексную колонку
Batch	Список счетчиков, которые могут вычисляться параллельно при чтении с одним и тем же условием PreWhere. Счетчики в одном батче могут иметь разные условия срабатывания и глубину поиска
Глубина поиска	Количество записей, которые СУБД должна прочитать, чтобы сформировать расчет одного счетчика
Материализованный счетчик	Рассчитанное историческое значение, сохраняемое в основном массиве данных с возможностью последующего рекуррентного использования для дальнейших расчётов
Узел, Инстанс, Нода	Единичный экземпляр (процесс) запущенного ПО

1. Общие положения

1.1. Наименование системы

1.1.1. Полное наименование и условное наименование системы

Полное наименование системы: «Tarantool Column Store».
Условное наименование системы: «Tarantool Column Store», или Система, или TCS.

1.1.2. Полное наименование разработчика

Общество с ограниченной ответственностью «ВК Цифровые Технологии».

1.1.3. Сокращенное наименование разработчика

ООО «ВК Цифровые Технологии».

1.2. Цели, назначение и области использования

1.2.1. Назначение системы

Tarantool Column Store (далее по тексту TCS) представляет собой распределенное блочно-колоночное хранилище данных для решений, подразумевающих комплексные аналитические запросы с группировками и расчетом итоговых значений (агрегатов), в сфере финтеха, real-time маркетинга, логистики, телекома и др., реализованное на базе in-memory платформы Tarantool Enterprise.

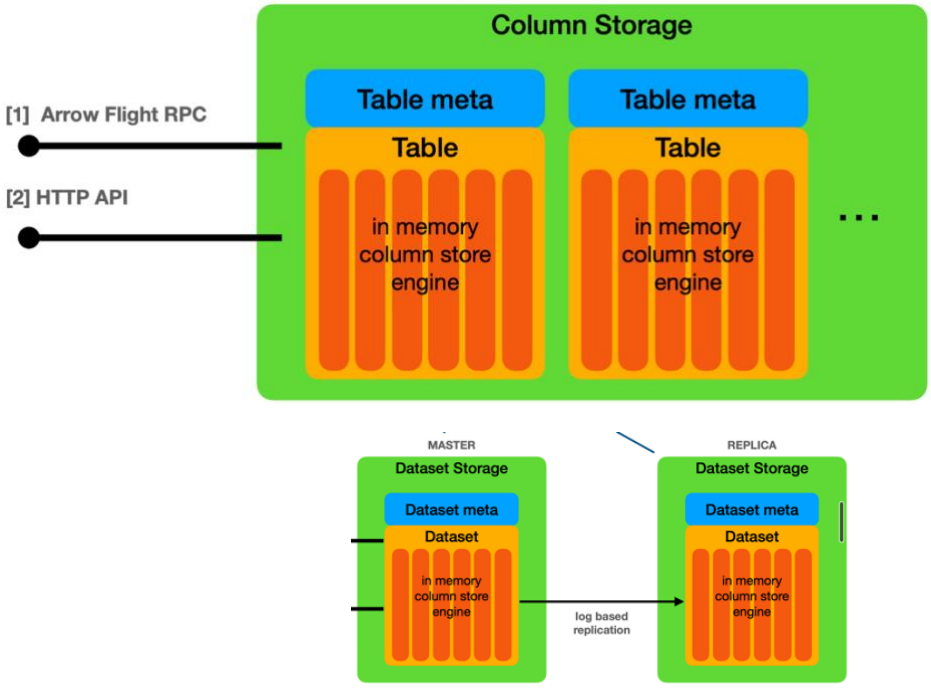
1.2.2. Цели создания системы

Целью создания системы является предоставление возможности хранить данные в оперативной памяти в блочно-колоночном виде и использовать их для обработки аналитических запросов в условиях высокой нагрузки.

Система ориентирована на выполнение аналитических запросов, требующих быстрого доступа к данным, а также поддерживает возможности оптимизаций производительности и обеспечение горизонтальной масштабируемости решения.

2. Основные технические решения

2.1. Концептуальная архитектура решения



Архитектура TCS состоит из следующих основных компонентов:

- Column Storage – распределенное блочно-колоночное хранилище данных:
  - Master - основной инстанс Tarantool (для записи);
  - Replica - резервный(-е) инстанс(-ы) Tarantool (для чтения).
- Интерфейс доступа к данным – обеспечивает возможность чтения/записи данных хранилища по следующим протоколам:
  - HTTP API для запросов на чтение с SQL;
  - Arrow Flight RPC.

2.2. Требования к режимам функционирования

TCS функционирует круглосуточно: 7 дней в неделю, 24 часа в сутки.

В зависимости от состояния компонентов TCS выделяются следующие основные режимы функционирования:

- Штатный режим;
- Аварийный режим функционирования;
- Режим обслуживания.

В штатном режиме функционирования TCS обеспечивает непрерывное функционирование сервисов и выполнение всех функций с заданными параметрами производительности.

Аварийным считается режим, при котором TCS или одна из его подсистем является полностью недоступной.

Для перевода TCS из аварийного в штатный режим функционирования необходимо выполнить комплекс мероприятий по восстановлению работоспособности, приведенный в документе «Эксплуатация экземпляра программного обеспечения».

В режиме обслуживания TCS обеспечивает:

- Проведение технического обслуживания, требующего полной или частичной остановки компонентов TCS, но не оказывающего существенного негативного влияния на основную работу пользователей.
- Модернизацию аппаратно-программного комплекса.
- Установка обновлений системного и прикладного ПО.

Общее время режима обслуживания TCS не должно превышать двух часов в сутки и должно приходиться на нерабочие часы основного числа пользователей.

2.3. Нефункциональные требования

Режим функционирования (штатный):

- 24/7, предусматривается технологическое окно;

- SLA = 99.7%;
- RPO = до 24 часов при наличии ежедневного бэкапирования;
- RTO = 2 часа.

Режим обслуживания:

- в технологическое окно.

2.3.1. Column Storage Master

Механизмы отказоустойчивости:

- избыточность по данным (репликация);
- механизм автоматического переключения трафика внутри и между компонентами Tarantool.

Объем потери данных: нет потерь

Время восстановления: 2 часа

Время простоя: недоступность на запись новых событий в компоненты TCS, находящихся в отказавшем ЦОДе на конфигурируемый таймаут (на практике, обычно устанавливается 10с для избежания преждевременных переключений)

2.3.2. Column Storage Replica

Механизмы отказоустойчивости:

- избыточность по вычислительной мощности;
- избыточность по данным (репликация);
- механизм автоматического переключения трафика внутри и между компонентами Tarantool.

Объем потери данных: нет потерь

Время восстановления: 2 часа

Время простоя: простой отсутствует

2.3.3. Интерфейс доступа к данным

Механизмы отказоустойчивости:

- избыточность по вычислительной мощности

Объем потери данных: нет потерь

Время восстановления: 2 часа

Время простоя: таймаут соединения

2.4. Требования к режиму работы

TCS должен поддерживать масштабирование до конфигурации, обеспечивающей соответствие следующим характеристикам:

Характеристика	Значение
Максимальное количество атрибутов (колонок) в хранилище	3000
Максимальное количество индексов (индексных колонок) в хранилище	100
Максимальная глубина каждого индекса по доменной сущности	1000
Среднее количество операций (нагрузка на запись)	10 000 RPS
Максимальное количество операций (пиковая нагрузка)	50 000 RPS
Среднее время вычисления счетчика, не более	100 мс
Среднее время вычисления счетчика в условиях пиковой нагрузки, не более	500 мс

2.5. Требования к диагностированию работы

2.5.1. Метрики

Метрики собираются и отдаются с помощью библиотеки metrics. Перечень и описание метрик, которые предоставляют компоненты ПО на базе Tarantool приведен в документации Tarantool, размещенной по адресу https://www.tarantool.io/ru/doc/latest/book/monitoring/metrics_reference/.

Сбор метрик осуществляется по следующим параметрам:

- **Rate** - количество запросов или задач, которые сервис обработал или принятых/отправленных сообщений. Пример: количество запросов к методу API. Метрика типа Counter. В момент запуска сервиса значение метрики равно 0. Значение метрики увеличивается на 1 после обработки каждого запроса к API или обработки каждой задачи.

- **Errors** - количество запросов или задач, обработка которых привела к ошибке. Пример: количество вызовов API подачи заявки, завершившихся с ошибкой авторизации. Метрика типа Counter. В момент запуска сервиса значение метрики равно 0. Значение метрики увеличивается на 1 после обработки каждого запроса, сообщения или задачи, который завершился с ошибкой.

- **Duration** - время, которое сервис затратил на обработку запроса или задачи. Метрика типа Histogram. В момент старта сервиса значение метрики по всем заданным интервалам гистограммы равно 0. Значение метрики, в соответствующем диапазоне гистограммы, увеличивается после обработки каждого запроса на 1. Нижняя граница нижнего интервала гистограммы – 0, нижняя граница верхнего интервала гистограммы – SLA, установленное на время обработки запроса или задачи. Используется не более 9 интервалов в гистограмме. Пример задания интервалов для мониторинга времени обработки сообщения: (0, 1], (1, 2], (2, 3], (3, +∞). Для расчета Duration не используется тип метрик Summary.

- Разделение расчета по методам и типам задач реализуется с использованием дополнительных меток (label).

- Рекомендуемые к использованию метки:

    - method - для методов API;

    - task_type или job_type для обработчиков задач.

2.5.2. Логирование

Логирование в TCS выполняет следующие функции:

- Запись логов в файл;

- Переконфигурирование во время исполнения: изменение уровня логирования, изменение пути к файлу с логом. Применение конфигурации логирования не требует перезагрузки компонентов ПО;
- Поддержка ротации логов. Для этого все компоненты ПО должны реализовывать обработку сигнала операционной системы HUP. По получении сигнала HUP компонент прекращает запись в файл лога, закрывает указатель на этот файл и открывает запись в файл лога по пути, указанному в конфигурации.
- Все компоненты записывают лог в формате JSON, пример:

```
{“timestamp”: “timestamp_value 1”, “message”: “log entry 1”}
{“timestamp”: “timestamp_value 2”, “message”: “log entry 2”}
```

Для всех операций должны регистрироваться следующие сведения:

- название операции;
- время операции в формате UTC с точностью до миллисекунд.

Что логируется:

- События жизненного цикла: запуск, остановка, изменение конфигурации.
- Ошибки, которые возникают во время работы.
- Взаимодействия между клиентом и TCS:
  - входящие/исходящие запросы;
  - результат выполнения запросов.

Уровень логирования VERBOSE обеспечивает расширенный уровень логирования, на котором в лог записывается:

- тело и заголовки запросов и ответов API;
- сообщения из очередей;
- сообщения, получаемые из потоков (stream) данных.

2.5.3. Алертинг

Для реализации оповещений о возникновении нештатных ситуаций в функционировании TCS предусматривается алертинг.

Алертинг реализуется средствами Zabbix с помощью механизмов Triggers и Notifications.

Для настройки правил оповещения для каждого правила описываются:

- метрика, на основе значения которой, будет рассчитываться алерт;
- пороги срабатывания алерта: значение метрики, при достижении/превышении которого отправляется оповещение;
- краткое описание рекомендуемых к выполнению действий для администратора ПО при срабатывании оповещения.

Перечень оповещений, значения порогов срабатывания оповещений формируются и дополняется в процессе и по результатам проведения тестирования, а также дополняется в процессе эксплуатации.

Оповещения настраиваются для:

- пороговых значений утилизации ресурсов сервера:
  - Например, занятое место на диске;
- пороговых значений утилизации ресурсов компонентами:
  - Например: объем данных в Tarantool достиг 80% выделенной процессу Tarantool памяти;
- пороговых значений следующих параметров работы TCS:
  - Количество запросов;
  - Количество ошибок в единицу времени;
  - Пороговых значений SLA на установленное время обработки запросов компонентами.
- пороговых значений времени прохождения заявок через компоненты ПО.

Общие характеристики оповещений:

- оповещения срабатывают при наступлении заданного условия срабатывания оповещения:
  - Достижения порогового значения метрик;

  - Аномальные отклонения от ожидаемых значений метрик;
  - Полное отсутствие метрик.
- при нормализации значения метрики, на которую сработало оповещения, срабатывают оповещения о нормализации метрики.

2.6. Требования к инфраструктуре

При планировании комплекса технических средств необходимо учитывать следующие требуемые характеристики TCS:

- Количество входящих запросов в секунду (RPS) на запись данных в хранилище Column Storage Master;
- Количество входящих запросов в секунду (RPS) на чтение данных из хранилища Column Storage Replica;
- Средний размер объекта – определяет требования к сетевым картам и количеству ядер CPU на серверах клиента;
- Совокупный объем хранилища в байтах – определяет требования к объемам физического хранилища серверов.