



ClickHouse для инженеров и архитекторов БД

Проекции и материализованные представления



Проверить, идет ли запись

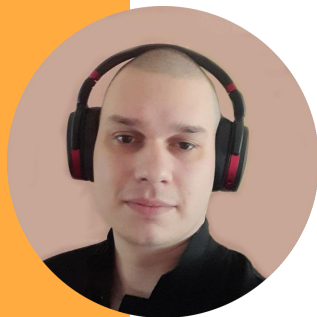
Меня хорошо видно && слышно?



Ставим "+", если все хорошо
"-", если есть проблемы

Тема вебинара

Проекции и материализованные представления



Константин Трофимов

Senior SRE / ClickHouse DBA в [VK](#)

Занимаюсь эксплуатацией ClickHouse с первых версий: 5 лет в VK, до этого в AdNow, до этого занимался Vertica. Сотни серверов, десятки кластеров, десятки петабайт данных.

Правила вебинара



Активно
участвуем



Off-topic обсуждаем
в учебной группе
#OTUS ClickHouse-2024-08



Задаем вопрос
в чат или голосом



Вопросы вижу в чате,
могу ответить не сразу

Условные обозначения



Индивидуально



Время, необходимое
на активность



Пишем в чат



Говорим голосом



Документ



Ответьте себе или
задайте вопрос

Карта курса



Темы модуля

«Масштабирование и манипуляции с данными»

1 (13). Проекция и материализованные представления

2 (14). Репликация и другие фоновые процессы

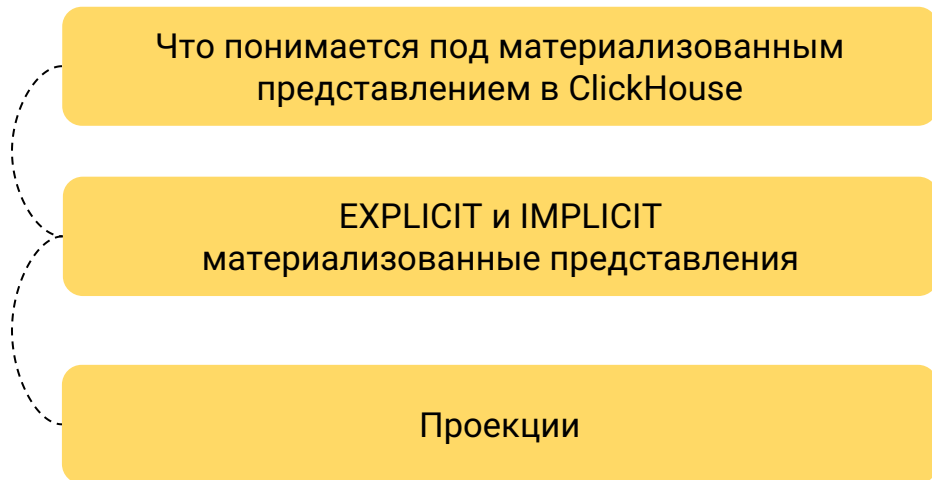
3 (15). Шардирование и распределенные запросы

4 (16). Мутация данных и манипуляции с партициями

5 (17). Сессия Q&A

Маршрут вебинара

Проекции и материализованные представления



Проекции и материализованные представления

Цели вебинара

После занятия вы сможете



1. Экономия ресурсов и ускорение запросов

2. Обогащение данных

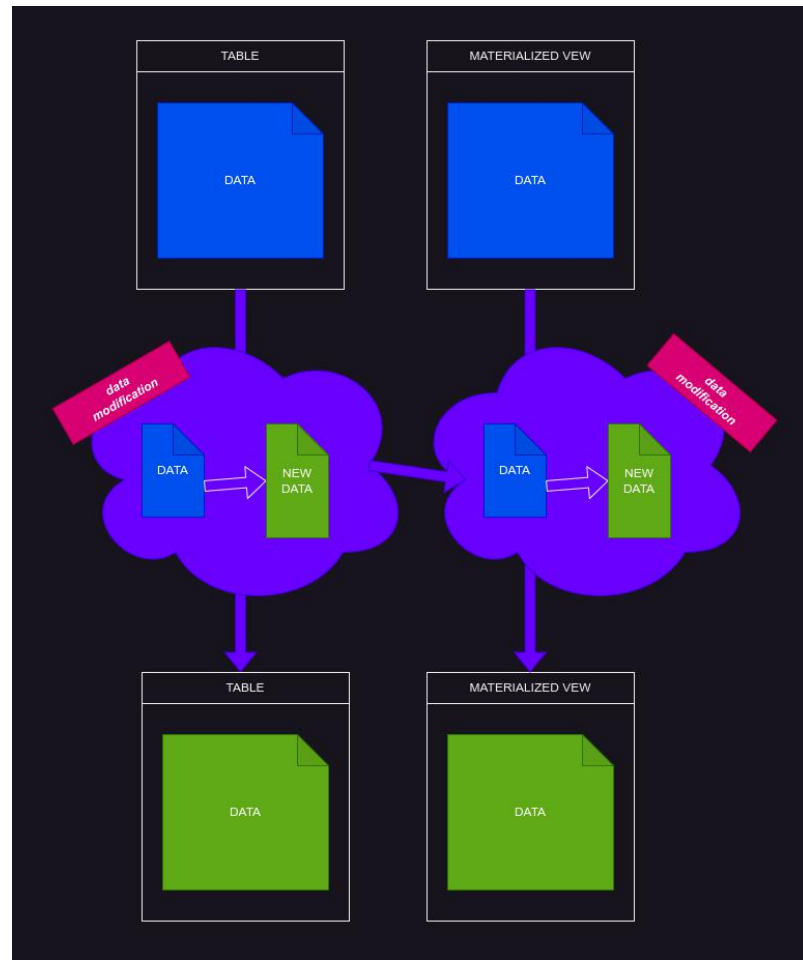


Что понимается под материализованным представлением в ClickHouse

В других системах

Материализованное представление (materialized view, сокращенно MV) является набором данных, преобразованных к новому виду на основании данных оригинальной таблицы.

При модификации данных, изменяются так же данные в материализованном представлении.

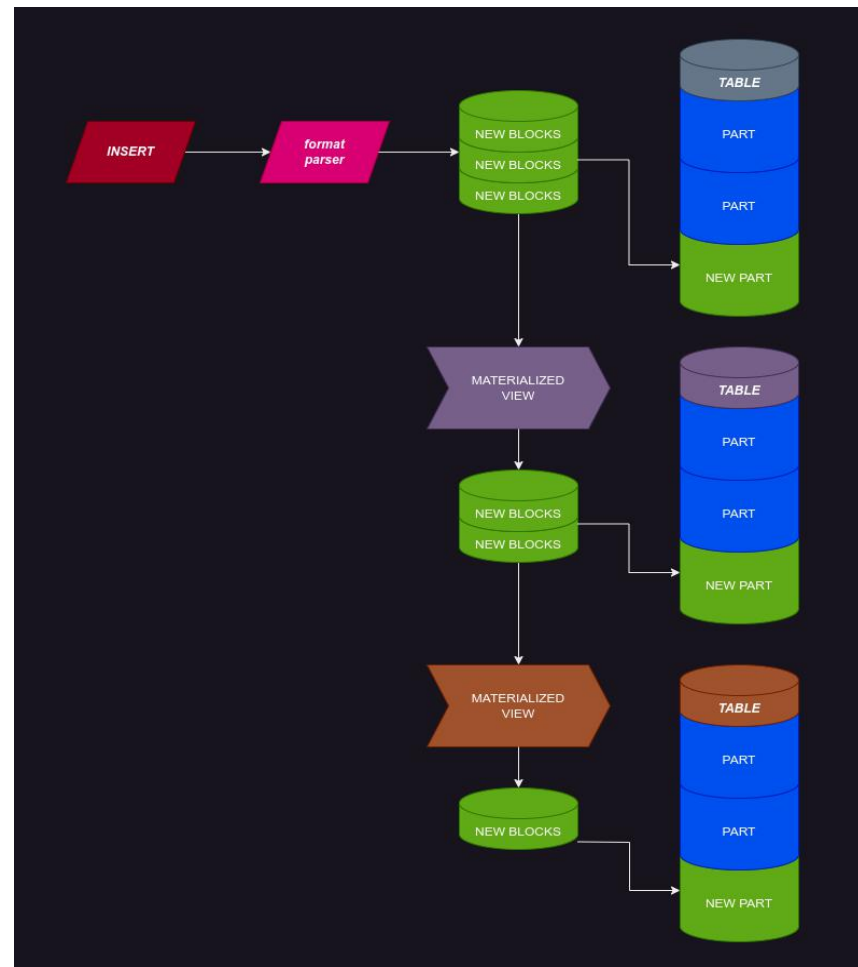


В ClickHouse

MV является **триггером на INSERT**, и **дообработчиком** данных, с записывающим **результат в другую** таблицу.

При добавлении новых данных посредством INSERT, они помимо вставки в целевую таблицу, так же дообрабатываются MV, и результат дообработки записывается ещё в одну таблицу.

Возможны каскады, сбор агрегатов с нескольких таблиц в одну, или с одной таблицы в несколько.



Цена за MV

1) Увеличение времени вставки в таблицу.

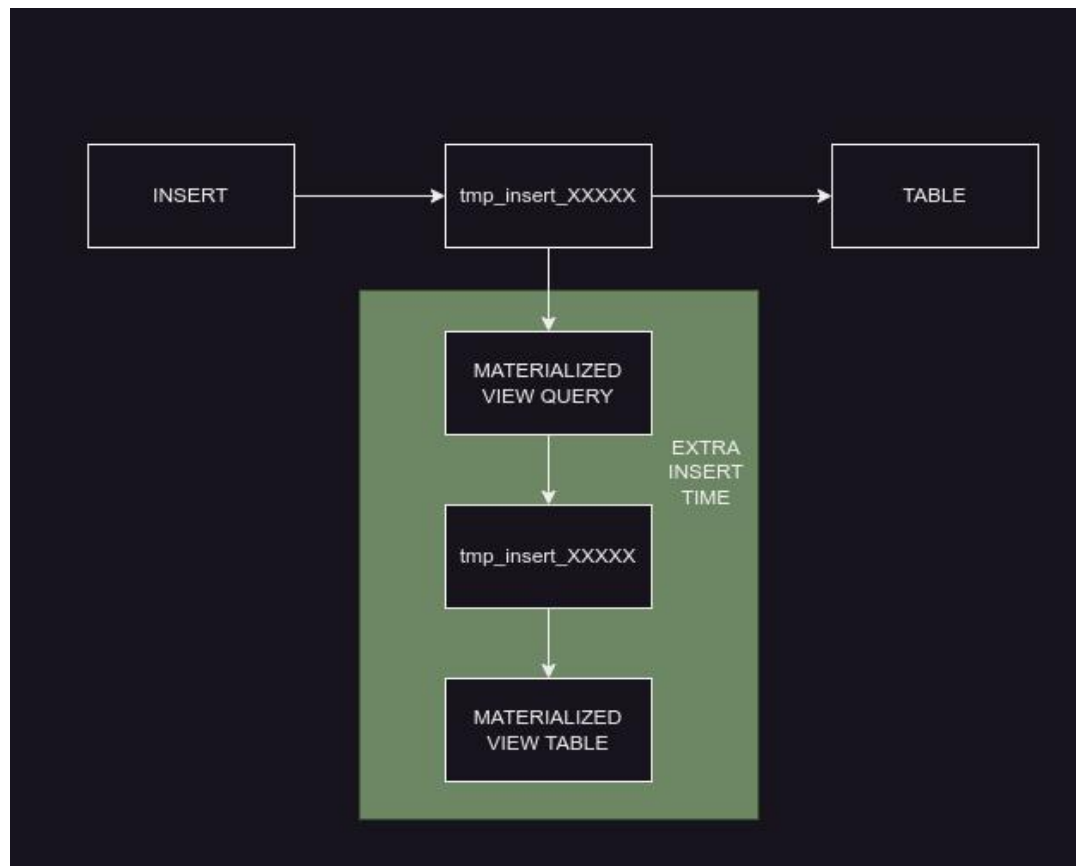
Вставка в MV будет произведена в рамках выполнения INSERT в оригинальную таблицу.

2) Повышение комплексной сложности миграций.

При выполнении миграций по оригинальной таблице, необходимо учитывать применимость материализованных представлений к новой схеме таблицы.

3) Необходимость учитывать MV при раздаче прав доступа.

Пользователь, выполняющий INSERT в таблицу, так же должен иметь право на INSERT во все таблицы, в которые этот INSERT принесет данные посредством MV.



Область применения MV

1) Сборка агрегатов и статистики для конечного пользователя.

Применяя Engine таблиц с дополнительной логикой агрегации данных, такие как AggregatingMergeTree, CollapsingMergeTree, SummingMergeTree

(**вебинар 3 (7). Движки MergeTree Family**)

можно достичь значительного ускорения запросов по сравнению с выборкой по оригинальным данным. MV будет заниматься наполнением таких таблиц, из данных оригинальной таблицы, по мере их вставки.

2) Обогащение данных. При объявлении MV можно использовать dictGet, доставая таким образом данных из других систем.

3) Интеграция с другими системами, специальными Engine таблиц,

(**вебинар 5 (9). Другие движки**)

где есть аналог INSERT, например Engine=Kafka.

Можно использовать MV как consumer, и сохранять данные из Kafka в ClickHouse.

Можно использовать MV как producer, и сохранять данные из ClickHouse в Kafka.

Вопросы?



Задаем
вопросы в чат



Ставим “-”,
если вопросов нет

EXPLICIT и IMPLICIT **материализованные** **представления**

IMPLICIT MATERIALIZED VIEW

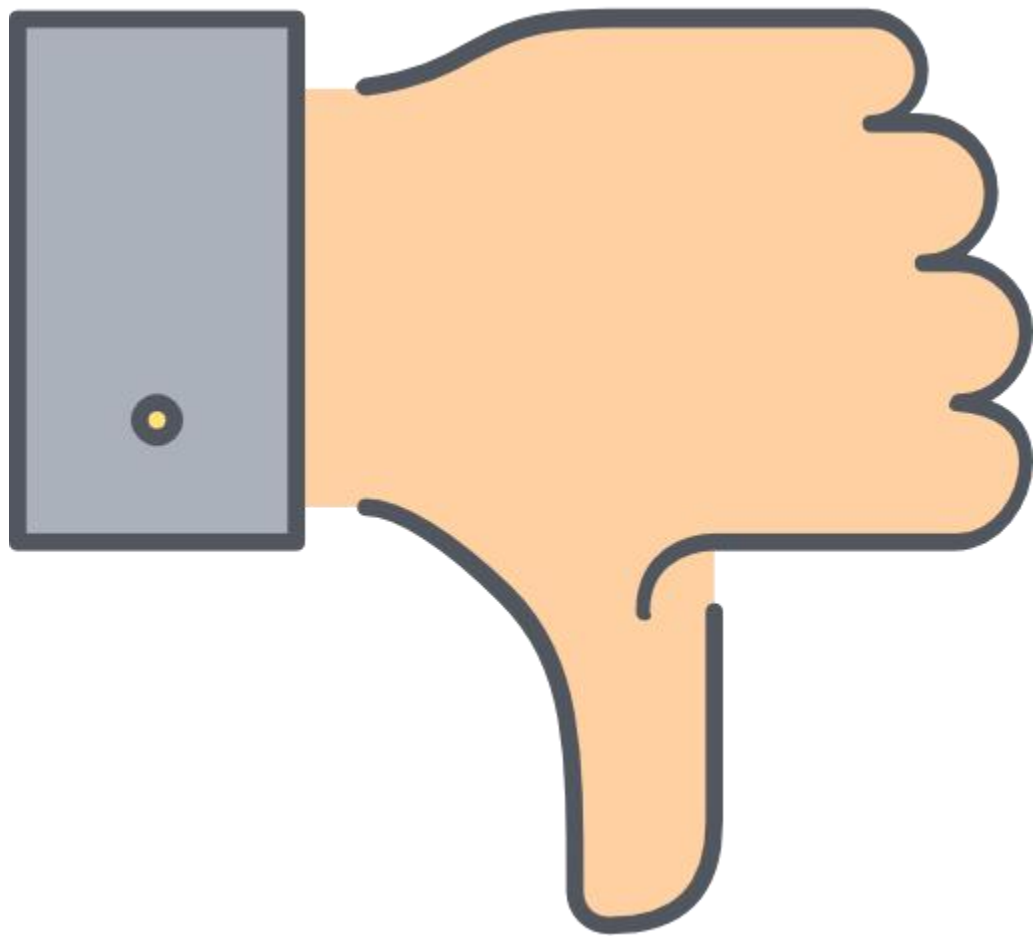
Таблица для хранения результирующих данных создается вместе с MV «mvName» как «.inner.mvName»

Таблица .inner дропается вместе с MV, но можно её переименовать и тогда она не дропнется вместе с MV.



Минусы IMPLICIT

- 1) Недокументированные, сложные методы изменения таблицы под MV
- 2) Синтаксис изменения запроса под MV требует включения экспериментальной функции
- 3) Примеры удалены из документации, это легаси



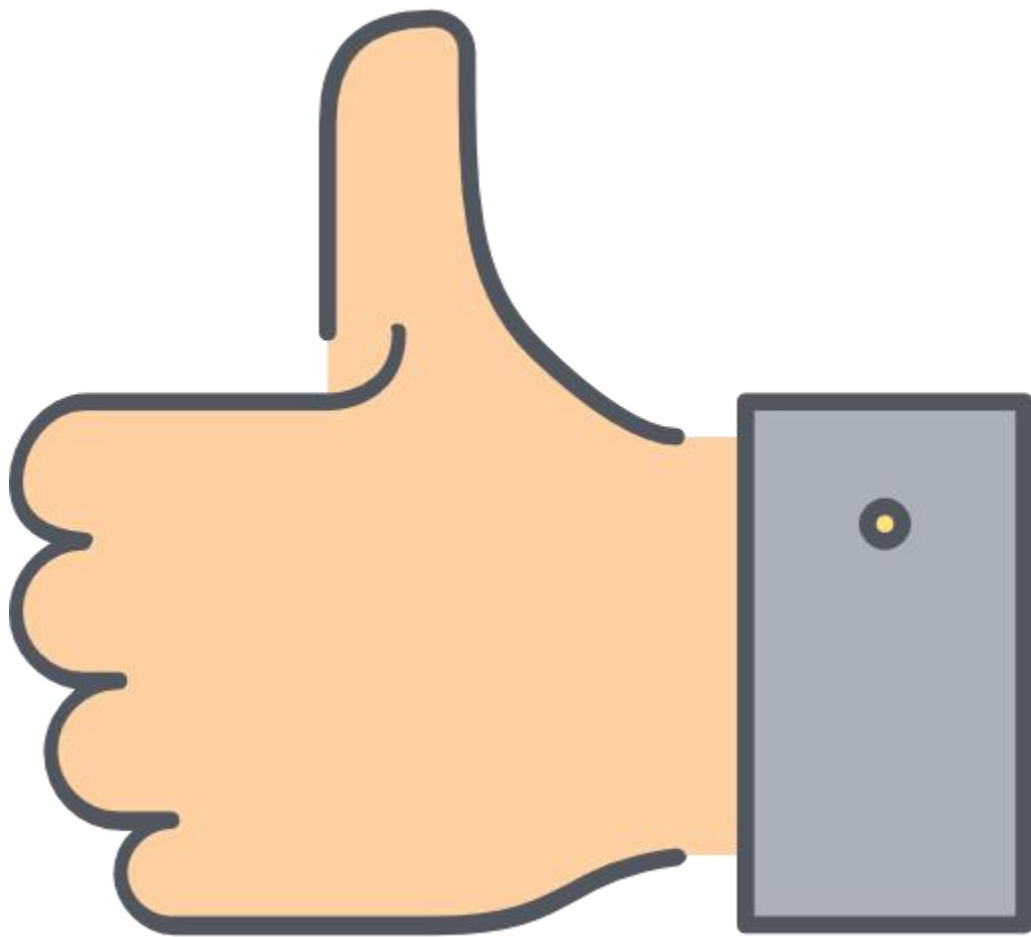
EXPLICIT MATERIALIZED VIEW

Таблицу для хранения результирующих данных необходимо создать заранее и отдельно.

<i>CREATE MATERIALIZED VIEW</i>	<i>name</i>	<i>TO tableName</i>
<i>AS</i>	<i>QUERY</i>	

Плюсы EXPLICIT

- 1) Можно менять запрос, пересоздавая MV (DROP+CREATE), не затрагивая данные
- 2) Таблица с результирующими данными независима, альтериться как обычно
- 3) Можно делать несколько MV в одну результирующую таблицу



**Практика - создание материализованных представлений,
конвертация IMPLICIT в EXPLICIT, смотреть в записи**

Вопросы?



Задаем
вопросы в чат



Ставим “-”,
если вопросов нет



Проекции

Что такое проекция

Проекция - ещё одна копия данных таблицы, с собственным ПК или даже промежуточными результатами агрегации.

Используется прозрачно для пользователя, не требует изменения SELECT-запросов к таблице.

Представляет собой подкаталог парта, с данными проекции, при Merge партов объединяются так же и проекции.

Управление проекциями

Добавление

ALTER TABLE [db.]name ADD PROJECTION [IF NOT EXISTS] name (SELECT <COLUMN LIST EXPR> [GROUP BY] [ORDER BY])

<COLUMN LIST EXPR> - произвольное выражение

ORDER BY - новый ПК

GROUP BY - для промежуточной агрегации, должно соответствовать <COLUMN LIST EXPR> так же, как и при обычном запросе SELECT

Материализация

При добавлении проекции, она начинает строиться только для вновь поступающих данных. Для существующих данных необходимо материализовать:

ALTER TABLE [db.]table MATERIALIZE PROJECTION [IF EXISTS] name [IN PARTITION partition_name]

IN PARTITION - материализовать в конкретной партии, если опущено материализуется во всех

Удаление

ALTER TABLE [db.]name DROP PROJECTION [IF EXISTS] name

Удаление только для конкретной партии

ALTER TABLE [db.]table CLEAR PROJECTION [IF EXISTS] name [IN PARTITION partition_name]

IN PARTITION - можно опустить, тогда удалится для всех, но продолжит собираться для вновь поступающих данных

Информация о проекциях

Для просмотра свойств проекций предусмотрены два системных представления.

- Представление ***system.projection_parts*** схоже с представлением `system.parts`, но содержит дополнительную информацию о родительских датапартах в полях, имеющих префикс `parent*`.
- Представление ***system.projection_parts_columns*** позволяет по имени таблицы или проекции посмотреть поля запроса проекции и их типы.

```
-- проверим куски нашей таблицы
SELECT name, active
FROM system.parts
WHERE table='tbl_with_projection';

-- и проверим куски проекции
SELECT name, parent_name, rows, active
FROM system.projection_parts
WHERE table='tbl_with_projection';

-- проверим типы полей проекции
SELECT column, type, column_position
FROM system.projection_parts_columns;
```

Использование проекций в запросах

optimize_use_projections

Включает или отключает поддержку проекций при обработке запросов SELECT.

Возможные значения:

- 0 — Проекции не поддерживаются.
- 1 — Проекции поддерживаются.

Значение по умолчанию: 0

Хотя явного изменения запросов не требуется, чтобы включить работу с проекциями необходимо установить `optimize_use_projections=1`. ClickHouse будет самостоятельно определять необходимость использования проекций при выполнении запросов.

force_optimize_projection

Включает или отключает обязательное использование проекций в запросах SELECT, если поддержка проекций включена (см. настройку `optimize_use_projections`).

Возможные значения:

- 0 — Проекции используются опционально.
- 1 — Проекции обязательно используются.

Значение по умолчанию: 0

Проекции используются принудительно, если проекции использовать нельзя, запрос выполнен не будет.

**Практика - создание и управление проекциями,
смотреть в записи**

Вопросы?



Задаем
вопросы в чат



Ставим “-”,
если вопросов нет

Рефлексия

Цели вебинара

Проверка достижения целей

1. Экономия ресурсов и ускорение запросов

2. Обогащение данных

Вопросы для проверки

Вопросы для проверки

1. Какой специальный Engine таблицы вы будете использовать чтобы не хранить исходные данные

2. Как приневсти данные из другой системы при помощи материализованного представления

Рефлексия



С какими впечатлениями уходите с вебинара?



Как будете применять на практике то, что узнали на вебинаре?

Следующий вебинар



18 ноября 2024 (понедельник)

Репликация и другие фоновые процессы



Ссылка на вебинар
будет в ЛК за 15 минут



Материалы
к занятию в ЛК —
можно изучать

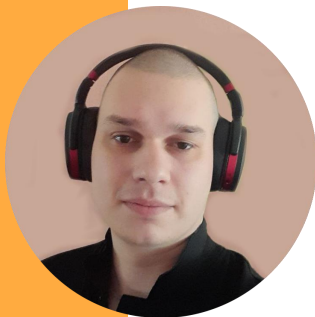


Обязательный
материал обозначен
красной лентой

**Заполните, пожалуйста,
опрос о занятии
по ссылке в чате**

Спасибо за внимание!

Приходите на следующие вебинары



Senior SRE / ClickHouse DBA в [VK](#)

Занимаюсь эксплуатацией ClickHouse с первых версий: 5 лет в VK, до этого в AdNow, до этого занимался Vertica. Сотни серверов, десятки кластеров, десятки петабайт данных.