

Настройки ClickHouse в Логикор

В данной статье описаны возможные настройки ClickHouse в Логикор. Перед ознакомлением с данной документацией рекомендуется изучить [статью Хранение событий в базе данных Логикор](#).

Настройки ClickHouse состоят из нескольких элементов:

- Общие настройки сервера: `/etc/clickhouse-server/config.xml`**
Здесь находятся общие настройки сервера, портов, таблиц логирования, метрик, а также различные параметры по используемым ресурсам и потокам сервера базы данных.
- Настройки кластера Логикор: `/etc/clickhouse-server/config.d/config-cluster.xml`**
Перечисляются хосты шардов и реплик ClickHouse, используемых в кластере Логикор, а также список хостов ClickHouse-Keeper для координации базы данных.
- Настройки макросов: `/etc/clickhouse-server/config.d/config-macros.xml`**
Как правило в макросах хранятся идентификаторы шарда и реплики для конкретного хоста ClickHouse, которые должны быть уникальны в кластере.
- Настройки хранения: `/etc/clickhouse-server/config.d/config-ilm.xml`**
Данные настройки отвечают за используемые диски и варианты записи данных по разделам и дискам.
- Настройки профилей: `/etc/clickhouse-server/users.d/default-profile.xml`**
В данных настройках находятся профили пользователей с предустановленными настройками, например, настройками по выполнению запросов.
- Пользователи: `/etc/clickhouse-server/users.d/default-user.xml`**
В данный момент присутствует 2 предустановленных пользователя — **logger** для записи данных в базу данных и **logger_reader** — для чтения данных из базы данных.
- Некоторые настройки могут относиться непосредственно к таблицам и находится в рамках созданных таблиц. Такие настройки можно посмотреть с помощью запроса **SHOW TABLE {tablename}**.

Общую информацию о формате и возможностях заполнения файлов конфигурации можно прочитать [в документации по ссылке](#).

Общие настройки сервера

Общие настройки сервера определяются в файле `/etc/clickhouse-server/config.xml`. Здесь находятся общие настройки сервера, портов, таблиц логирования, метрик, а также различные параметры по используемым ресурсам и потокам сервера базы данных.

Более подробно ознакомиться с возможными настройками и их значениями можно [в документации по ссылке](#).

Настройки кластера Логикор

Настройки кластера **logger_cluster**, а также настройки координаторов **ClickHouse-Keeper** расположены в файле `/etc/clickhouse-server/config.d/config-cluster.xml`. В них перечисляются хосты шардов и реплик ClickHouse, используемых в кластере Логикор, а также список хостов ClickHouse-Keeper для координации базы данных.

Подробную информацию об устройстве кластеров в ClickHouse можно прочитать [в документации](#). А также можно ознакомиться с описанием заполнения конфигурации кластера [в документации](#).

Настройки кластера базы данных

Пример настроек кластера **logger_cluster**:

```
<clickhouse>
  <remote_servers>
    <logger_cluster>
      <shard>
        <replica>

          <host>clickhouse-1-1</host>
          <port>9000</port>
          <user from_env="CLICKHOUSE_USER" replace="1"/>
          <password from_env="CLICKHOUSE_PASSWORD" replace="1"/>

        </replica>

        <replica>

          <host>clickhouse-1-2</host>
          <port>9000</port>
          <user from_env="CLICKHOUSE_USER" replace="1"/>
          <password from_env="CLICKHOUSE_PASSWORD" replace="1"/>

        </replica>
      </shard>
    </logger_cluster>
  </remote_servers>
</clickhouse>
```

```
...
</remote_servers>
</clickhouse>
```

В данном примере для кластера **logger_cluster** используются исключительно 2 реплики в рамках одного шарда. Количество шардов и реплик может меняться в зависимости от потребностей конкретной инсталляции.

В качестве параметра **host** можно использовать **IP** либо **Hostname** серверов ClickHouse, участвующих в кластере.

В данном примере учетные данные для доступа к хостам кластера передаются в качестве переменных среды, но могут передаваться другим доступным способом.

Настройки кипера для кластера

Также в данном файле присутствует информация о подключенных узлах ClickHouse-Keeper:

```
<zookeeper>
<node>

  <host>clickhouse-keeper-1</host>
  <port>9181</port>

</node>

<node>

  <host>clickhouse-keeper-2</host>
  <port>9181</port>

</node>

<node>

  <host>clickhouse-keeper-3</host>
  <port>9181</port>

</node>
</zookeeper>
```

В данном примере используется 3 узла **ClickHouse-Keeper** для координации реплицируемых таблиц.

Настройки удаленных серверов

Для использования тенантов в системе Логикор необходимо указать список доступных внешних серверов ClickHouse. Данная настройка расположена в этом же файле и по умолчанию разрешен доступ ко всем внешним серверам ClickHouse:

```
<remote_url_allow_hosts>

  <host>127.0.0.1</host>
  <host_regexp>.*</host_regexp>

</remote_url_allow_hosts>
```

Настройки макросов

Настройки макросов уникальны для каждого узла кластера ClickHouse. Они располагаются в файле **/etc/clickhouse-server/config.d/config-macros.xml**.

Пример настроек:

```
<clickhouse>

  <macros>

    <shard>1</shard>
    <replica>2</replica>

  </macros>

</clickhouse>
```

В данном примере настройки переменных макросов относятся ко второй реплике первого шарда.

Данный файл может быть заполнен иначе, в соответствии [с документацией по ссылке](#).

Настройки хранения

Данные настройки отвечают за используемые диски, разделы, необходимое зарезервированное место на дисках, а также за настройку параметров автоматического перемещения данных между уровнями хранения (`move_factor`). Подробнее о настройках можно почитать [в документации ClickHouse](#). На описание возможных настроек хранения можно посмотреть [в документации](#).

Настройки дисков

В файле `/etc/clickhouse-server/config.d/config-ilm.xml` приведены настройки используемых дисков с указанием путей монтирования (`path`), а также размер зарезервированного свободного места на каждом диске в байтах (`keep_free_space_bytes`), который должен оставаться свободным, и которое ClickHouse не должен использовать для размещения данных. Пример настройки:

```
<disks>

  <hot_disk>

    <path>/data/hot/</path>
    <keep_free_space_bytes>20000000000</keep_free_space_bytes>

  </hot_disk>

  <warm_disk>

    <path>/data/warm/</path>
    <keep_free_space_bytes>20000000000</keep_free_space_bytes>

  </warm_disk>

  <cold_disk>

    <path>/data/cold/</path>
    <keep_free_space_bytes>20000000000</keep_free_space_bytes>

  </cold_disk>

</disks>
```

Обратите внимание, что даже при указании зарезервированного свободного места на диске, фактическое свободное место может оказаться меньше, так как при работе (объединение/слияние партов, рекомпрессии партов, репликации и прочих действия) ClickHouse может временно превышать лимит, что приведет к уменьшению свободного места на диске. Превышение этих лимитов связано с тем, что при объединении несколько партов сначала создается новый парт, в который записываются данные, а после записи всех данных исходные парты удаляются.

Рекомендуется резервировать размер, не менее размера одной средней партиции данных.

Настройки разделов

На основе существующих дисков могут быть настроены несколько политик хранения, где каждая политика хранения будет определяться используемыми разделами на дисках.

В данном примере обозначены две политики хранения **logiq_policy** и политика **default**:

```
<logiq_policy>

  <volumes>

    <hot_volume>
      <disk>hot_disk</disk>
    </hot_volume>

    <warm_volume>
      <disk>warm_disk</disk>
    </warm_volume>

    <cold_volume>
      <disk>cold_disk</disk>
    </cold_volume>

  </volumes>

  <move_factor>0.2</move_factor>

</logiq_policy>
```

```
<default>
  <volumes>
    <default>
      <disk>default</disk>
    </default>
  </volumes>
</default>
```

В рамках политики **logiq_policy** указан параметр **move_factor = 0.2**, который отвечает за то, что при заполнении раздела более чем на 80% данные будут автоматически перемещаться на следующий свободный раздел. Разделы будут использоваться по мере их перечисления в списке **volumes**.

В случае, если все разделы заняты более чем на 80% (из примера) — при попытке записи ClickHouse будет выдавать ошибку о недостаточном объеме свободного диска.

Создаваемые в базе данных таблицы будут размещаться на дисках и разделах в соответствии с настроенной для таблицы политики хранения. Если политика хранения для таблицы не задана, то будет использоваться политика по умолчанию (default).

Диск default

Также стоит обратить внимание на диск **default**. В отличие от остальных дисков, которые были описаны ранее в примере и сконфигурированы в списке **disks**, путь до основного диска **default** определятся в настройке **path** файла **/etc/clickhouse-server/config.xml**.

Пример определения диска **default**:

```
<path>/var/lib/clickhouse/</path>
```

Настройки профилей

В файле **/etc/clickhouse-server/users.d/default-profile.xml** определены два профиля в рамках Логикор: **default** и **logger_reader** — для чтения данных.

Пример настройки:

```
<profiles>
  <default>
    <!-- Async Inserts options -->
    <async_insert>1</async_insert>
    <async_insert_max_data_size>20000000</async_insert_max_data_size>
    <async_insert_busy_timeout_ms>600</async_insert_busy_timeout_ms>
    <async_insert_deduplicate>1</async_insert_deduplicate>
    <wait_for_async_insert>0</wait_for_async_insert>

    <!-- Date/Time options -->
    <date_time_output_format>iso</date_time_output_format>
    <date_time_input_format>best_effort</date_time_input_format>
    <format_csv_null_representation/>

    <!-- Queries options -->
    <materialize_ttl_after_modify>0</materialize_ttl_after_modify>

    <!-- Data Types Options -->
    <enable_json_type>1</enable_json_type>
  </default>

  <logger_reader>
  </logger_reader>
</profiles>
```

Настройки профилей подробно описаны [в документации в разделе Настройки Сессии](#).

Данные настройки применяются для сессий подключения и для выполнения запросов к базе данных в рамках этих сессий.

В рамках каждого профиля указаны преднастроенные параметры, которые нет необходимости передавать вместе с

каждым запросом. При передаче соответствующих параметров с запросом — данные предустановленные параметры будут переопределены (параметры из запроса имеют приоритет над параметрами профиля).

Настройки пользователей

В файле `/etc/clickhouse-server/users.d/default-user.xml` задаются предустановленные пользователи для каждого узла ClickHouse.

[В документации ClickHouse](#) можно посмотреть подробное описание настроек пользователей через xml.

Пример настройки пользователя `logger_reader` для Логикор:

```
<users>
  <logger_reader>
    <profile>logfile</profile>
    <networks>
      <ip>::/0</ip>
    </networks>
    <password from_env="CLICKHOUSE_READER_PASSWORD" replace="true"/>
    <quota>default</quota>
    <access_management>0</access_management>
    <named_collection_control>1</named_collection_control>
  </logger_reader>
</users>
```

Для каждого пользователя указан его профиль (из Настроек Профилей), а также учетные данные и дополнительные настройки прав.

Настройки таблиц

Для каждой таблицы могут быть установлены собственные настройки при их создании.

Для просмотра настроек таблицы, например, `messagesTable`, необходимо выполнить запрос:

```
SHOW CREATE TABLE logger.messagesTable
```

При этом будет отображена схема таблицы, а также ее настройки:

```
CREATE TABLE logger.messagesTable
(
  `timestamp` DateTime('UTC'),
  `uuid` UUID,
  ...
)
ENGINE = ReplicatedMergeTree('/clickhouse/tables/messagesTable/{shard}', '{replica}')
PARTITION BY toStartOfInterval(timestamp, toIntervalDay(1))
ORDER BY (timestamp, uuid)
TTL
timestamp + toIntervalDay(7) RECOMPRESS CODEC(ZSTD(8)),
timestamp + toIntervalDay(9) TO DISK 'cold_disk',
timestamp + toIntervalMonth(6)
SETTINGS index_granularity = 8192,
  ttl_only_drop_parts = 1,
  merge_with_ttl_timeout = 86400,
  storage_policy = 'logiq_policy'
```

В данном примере указано, что таблица `messagesTable` является реплицируемой таблицей (ReplicatedMergeTree) с использованием макросов `{shard}` и `{replica}` из **Настроек Макросов**.

При этом, информация о синхронизации в ClickHouse-Keeper будет располагаться по указанному пути: `/clickhouse/tables/messagesTable/{shard}/{replica}`.

Партиционирование

Параметр **PARTITION BY toStartOfInterval(timestamp, toIntervalDay(1))** указывает настройку партиционирования — то есть объединения данных в большие разделы таблицы. В данном примере партиционирование ежедневное, то есть данные (события) из каждого дня будут объединены в одну партицию с целью оптимизации выполнения запросов. При этом, фактически, партиция будет состоять из множества партов данных, которые со временем будут объединяться между собой.

Подробную информацию о партиционировании можно найти [в документации ClickHouse](#).

Первичный ключ

Параметр **ORDER BY (timestamp, uuid)** задает ключ сортировки таблицы. В данном случае, первичным ключом будет являться пара — **timestamp + uuid**.

TTL

Далее приведен пример настройки **TTL** (time to live) таблицы. В данной таблице срок исполнения **TTL** будет отсчитываться от основной даты события — **timestamp**. В рамках примера заложено, что:

- через 7 дней (относительно даты события — **timestamp**) будет произведена рекомпрессия данных в кодировке ZSTD(8): **timestamp + toIntervalDay(7) RECOMPRESS CODEC(ZSTD(8))**;
- через 9 дней парты таблицы будут перемещены на диск **cold_disk**: **timestamp + toIntervalDay(9) TO DISK 'cold_disk'**;
- данные автоматически удалятся из таблицы через 6 месяцев: **timestamp + toIntervalMonth(6)**.

Подробную информацию по настройке TTL таблиц можно найти [в документации ClickHouse](#).

Настройки

В разделе **SETTINGS** приведены индивидуальные настройки, относящиеся к конкретной таблице:

- **index_granularity = 8192** — системная настройка, отвечающая за гранулярность индекса таблицы;
- **ttl_only_drop_parts = 1** — настройка, отвечающая за удаление партов данных по TTL целиком, вместо удаления конкретных строк из самого парта;
- **merge_with_ttl_timeout = 86400** — минимальный интервал срабатывания TTL (раз в сутки);
- **storage_policy = 'logiq_policy'** — настройка используемой политики хранения таблицы. В данном случае используется настроенная ранее политика **logiq_policy**.

Подробную информацию по настройкам таблиц можно найти [в документации ClickHouse](#).

Изменение TTL

Для изменения **TTL** можно использовать конструкцию:

```
ALTER TABLE [db.]table_name [ON CLUSTER cluster] MODIFY TTL ttl_expression;
```

В качестве примера можно привести изменение **TTL** для таблицы **messagesTable**:

```
ALTER TABLE logger.messagesTable ON CLUSTER logger_cluster
```

```
MODIFY TTL
```

```
timestamp + toIntervalDay(2) RECOMPRESS CODEC(ZSTD(8)),
timestamp + toIntervalDay(4) TO DISK 'cold_disk',
timestamp + toIntervalMonth(1)
```

```
SETTINGS materialize_ttl_after_modify=0
```

В данном примере изменены параметры **TTL** для основной таблицы.

Обратите внимание на то, что использована настройка **materialize_ttl_after_modify=0**. Данная настройка означает, что при изменении TTL не нужно менять TTL для уже сформированных и сохраненных (для старых) партов. В противном случае, при больших объемах хранимых данных, это может привести к большой загрузке ClickHouse и потере работоспособности. Таким образом новые значения TTL будут применяться исключительно к новым (вновь создаваемым) партам.

Данная настройка также находится в профиле **default**. Но при выполнении запроса важно под каким профилем исполняются запросы, поэтому стоит указывать эту настройку дополнительно.

Рекомендации

1. Для исполнения запросов на всех нодах кластера можно использовать следующие конструкции:
 - Для создания/редактирования таблиц и их параметров, включая TTL:

ON CLUSTER logger_cluster

- Для запросов к данным, например, ``system.errors``:

```
SELECT * FROM clusterAllReplicas(logger_cluster, system.errors)
```

2. Резервируйте на диске достаточное количество места для работы ClickHouse:

```
<keep_free_space_bytes>20000000000</keep_free_space_bytes>
```

3. В рамках **TTL** используйте рекомпрессию на быстрых дисках и до перемещения на медленные диски:

```
timestamp + toIntervalDay(2) RECOMPRESS CODEC(ZSTD(8)),
```

```
timestamp + toIntervalDay(4) TO DISK 'cold_disk',
```

```
timestamp + toIntervalMonth(1)
```

4. Не меняйте **TTL** у уже существующих данных (партов), это может привести к зависанию базы данных, так как является очень дорогостоящей по CPU операцией. Используйте для этого следующую настройку:

```
materialize_ttl_after_modify=0
```

ID статьи: 1636

Последнее обновление: 7 мая, 2026

Обновлено от: Михалева А.

Ревизия: 1

База знаний Логикор -> Документация -> Система хранения и обработки данных «Логикор». Версия 2.7.0 -> Логикор.

Руководство разработчика -> Управление конфигурационными файлами -> Настройки ClickHouse в Логикор

<https://docs.axel.pro/entry/1636/>