

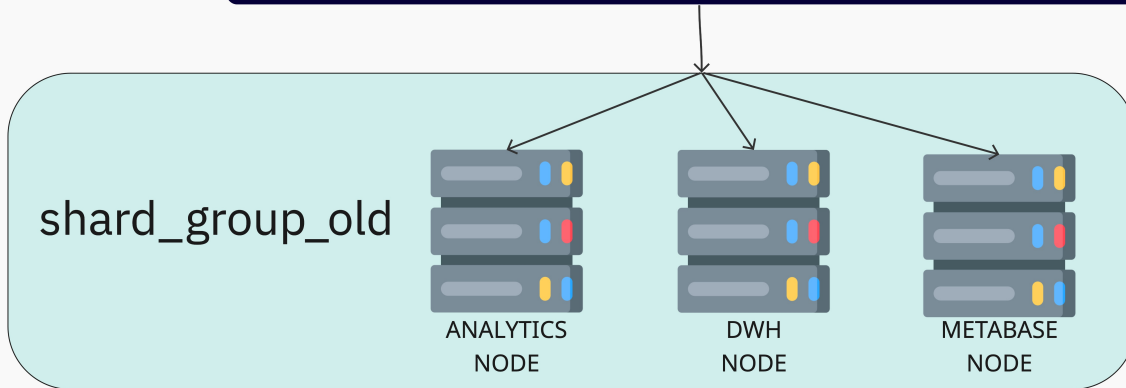
3. ON CLUSTER: Как правильно использовать

Для понимания необходимо ознакомиться с [2. Физическая архитектура аналитического СХД](#).



Модификатор `ON CLUSTER` отправляет запрос на все ноды внутри указанной шард-группы.

```
1 CREATE TABLE ods.assembly__orders ON CLUSTER shard_group_old
2 ...
```



Работа с *Replicated** таблицами

При создании и удалении таблиц необходимо использовать `ON CLUSTER`. Иначе таблица не создается/удалится на всех нодах.

- `CREATE TABLE/VIEW ... ON CLUSTER shard_group_old`
- `DROP TABLE/VIEW ... ON CLUSTER shard_group_old`

Следующие команды обязательно использовать без `ON CLUSTER`!

- `TRUNCATE ...`
- `ALTER TABLE ... DROP/REPLACE PARTITION ...`
- `OPTIMIZE TABLE ...`
- `ALTER TABLE ... ADD/DROP/RENAME/MODIFY/... COLUMN/TTL`
- `ALTER TABLE ... DELETE (, подробнее)`
- `ALTER TABLE ... UPDATE (, подробнее)`

ClickHouse сам автоматически синхронизирует изменения для *Replicated** таблиц, подробнее ниже.



Запускать `ALTER TABLE ... DELETE/UPDATE` строго не рекомендуется, а использование с модификатором `ON CLUSTER` может поломать таблицу. [Пример такого кейса](#).

Как работает ON CLUSTER

Если запрос выполнен с модификатором `ON CLUSTER`, то он попадает в очередь кластера `system.distributed_ddl_queue`. Из этой очереди еще невыполненные запросы выполняются синхронно по одному (на случайной ноды, после она отдает задания на репликацию данных остальным нодам).

При запуске тяжелого запроса с `ON CLUSTER`, например, `ALTER TABLE ... DELETE`, все последующие запросы в этой очереди будут ожидать завершения данного тяжелого запроса, прежде чем начнутся. Помимо этого появляется дополнительная нагрузка на ZooKeeper и сам ClickHouse.

Если таких запросов будет много, то очередь забьется, что приведет к поломке кластера и последующему инциденту.

Если же выполнить, к примеру, `ALTER TABLE ... (Replicated* таблица) REPLACE PARTITION ...` без `ON CLUSTER`, то нода, из которой запущен запрос, сама его выполнит, минуя `system.distributed_ddl_queue`, и пошлет задания на репликацию остальным нодам.

Нельзя использовать конструкцию `CREATE TABLE ... ON CLUSTER ... AS SELECT ...`

В этом случае ClickHouse выполнит запрос на всех трех нодах!

Что утяжелит нагрузку от запроса на кластер в три раза. Более того, пока этот запрос не выполнится, стоящие в очереди задания из `system.distributed_ddl_queue` будут его ждать.

Использование `CREATE TABLE ... ON CLUSTER ... AS SELECT ...` может привести к поломке кластера и последующему инциденту.

*`CREATE TABLE ... AS SELECT ...` (Без `ON CLUSTER`) использовать можно.