How to import and replicate data from MySQL to ClickHouse ?

Why do we recommend ClickHouse for Analytics ?

- ClickHouse is a Columnar Store built for SORT / SEARCH queries performance on a very large volume of database.
- In Columnar Database Systems the values from different columns are stored separately, and data from the same column is stored together - Benefits Performance of Analytical Queries (ORDER / GROUP BY & Aggregation SQLs)
- Columnar Stores are best suited for analytics because of their ability to retrieve just those columns instead of reading all of the rows and filter out unneeded data makes the data accessed faster.

Installation - How to install ClickHouse on DEB and RPM Packages ?

From DEB Packages

To install official packages add the Yandex repository in /etc/apt/sources.list or in a separate /etc/apt/sources.list.d/clickhouse.list file:

```
$ deb http://repo.yandex.ru/clickhouse/deb/stable/ main/
```

If you want to use the most recent version, replace **stable with **testing** (this is recommended for your testing environments).

Then run these commands to actually install packages:

```
$ sudo apt-get install dirmngr      # optional
$ sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv E0C56BD4   # optional
$ sudo apt-get update
$ sudo apt-get install clickhouse-client clickhouse-server
```

From RPM Packages

To install official pre-compiled rpm packages for CentOS, RedHat and all other rpm-based Linux distributions, First you need to add the official repository:

```
$ sudo yum install yum-utils
$ sudo rpm --import https://repo.yandex.ru/clickhouse/CLICKHOUSE-KEY.GPG
$ sudo yum-config-manager --add-repo https://repo.yandex.ru/clickhouse/rpm/stable/x86_64
```

** If you want to use the most recent version, replace **stable** with **testing** (this is recommended for your testing environments).

Then run these commands to actually install packages:

```
$ sudo yum install clickhouse-server clickhouse-client
```

Start ClickHouse Services

To start the server as a daemon, run:

```
$ sudo service clickhouse-server start
```

If you don't have service command, run as

```
$ sudo /etc/init.d/clickhouse-server start
```

Transfer the data from MySQL into ClickHouse

Recommended solution - MinervaDB ClickHouse MySQL Data Reader (forked from Alitinity/clickhouse-mysql-data-reader)

GitHub - https://github.com/MinervaDB/MinervaDB-ClickHouse-MySQL-Data-Reader

Installation and configuration

** *Prerequisites* - we have to install pip3 (python3-pip in Ubuntu) as Python in version at least 3.4 is required. Then we can use pip3 to install some of the required Python modules

```
pip3 install mysqlclient
pip3 install mysql-replication
pip3 install clickhouse-driver
```

we have to clone the repository next. For Centos 7, RPM's are also available, it is also possible to install it using pip3 (clickhouse-mysql package) but we found that the version available through pip does not contain the latest updates and we want to use master branch from git repository:

```
git clone https://github.com/Altinity/clickhouse-mysql-data-reader
```

Installation using pip

```
pip3 install -e /path/to/clickhouse-mysql-data-reader/
```

Next step will be to create MySQL users required by clickhouse-mysql-data-reader to access MySQL data:

```
mysql> CREATE USER 'clickhousereader'@'%' IDENTIFIED BY 'MDB@2019';
Query OK, 0 rows affected (0.02 sec)
mysql> CREATE USER 'clickhousereader'@'127.0.0.1' IDENTIFIED BY 'MDB@2019';
Query OK, 0 rows affected (0.00 sec)
mysql> CREATE USER 'clickhousereader'@'localhost' IDENTIFIED BY 'MDB@2019';
Query OK, 0 rows affected (0.02 sec)
mysql> GRANT SELECT, REPLICATION CLIENT, REPLICATION SLAVE, SUPER ON *.* TO
'clickhousereader'@'%';
Query OK, 0 rows affected (0.01 sec)
mysql> GRANT SELECT, REPLICATION CLIENT, REPLICATION SLAVE, SUPER ON *.* TO
'clickhousereader'@'127.0.0.1';
Query OK, 0 rows affected (0.00 sec)
mysql> GRANT SELECT, REPLICATION CLIENT, REPLICATION SLAVE, SUPER ON *.* TO
'clickhousereader'@'localhost';
Query OK, 0 rows affected, 1 warning (0.01 sec)
```

My.CNF (We have enabled log_bin)

```
[mysqld]
# mandatory
server-id        = 200
log_bin          = /var/lib/mysql/bin.log
binlog-format    = row # very important if you want to receive write, update and delete row
events
# optional
expire_logs_days = 30
max_binlog_size  = 900M
# setup listen address
bind-address     = 0.0.0.0
```

When you have successfully completed the steps above, You can proceed with data import to ClickHouse. We recommend you run the import on a host with tables locked to avoid possible data corruption errors. You can use a MySQL slave (strongly recommended) as the source of the data. The command you will be running is:

```
clickhouse-mysql --src-server-id=200 --src-wait --nice-pause=1 --src-host=192.168.56.101
--src-user=clickhousereader --src-password=MDB@2019 --src-tables=wiki.pageviews
--dst-host=127.0.0.1 --dst-create-table --migrate-table
```

The above script connect to MySQL on host 192.168.56.101 using given credentials, it will copy the table 'pageviews' in the schema 'wiki' to a ClickHouse running on the local host (127.0.0.1). Table will be created automatically and data will be migrated.

** We have used WikiStat database in the example - http://dumps.wikimedia.org/other/pagecounts-raw/

Validate successful data import by counting records of table table "pageviews" in both MySQL and ClickHouse

MySQL:

```
mysql> SELECT COUNT(*) FROM wiki.pageviews\G
*************************** 1. row ***************************
COUNT(*): 852356145
1 row in set (3 min 57.26 sec)
```

ClickHouse:

```
ClickHouse-MDBLAB :) SELECT COUNT(*) FROM wiki.pageviews\G

SELECT COUNT(*)
FROM wiki.pageviews

Row 1:
──────────
COUNT(): 852356145

1 rows in set. Elapsed: 0.183 sec. Processed 852.36 million rows, 474.90 MB (4.78 billion
rows/s., 4.74 GB/s.)
```

*P.S. Compare the performance of same query in MySQL and ClickHouse above

How can you replicate data from MySQL to ClickHouse closely watching binary log position ?

We can use binary log file and position from where the clickhouse-mysql tool should start listening. You can find those details from slave after completing the first time import

```
clickhouse-mysql --src-server-id=200 --src-resume --src-binlog-file='binlog.000713'
--src-binlog-position=638 --src-wait --nice-pause=1 --src-host=192.168.56.101
--src-user=clickhousereader --src-password=MDB@2019 --src-tables=wiki.pageviews
--dst-host=127.0.0.1 --pump-data --csvpool
```

Limitations

- Only INSERT transactions are supported
- No support for DELETE or UPDATE
- No support for DDL - Any incompatible schema changes done on MySQL will eventually break MySQL to ClickHouse Replication