ClickHouse
Roadmap & Favourite Features

Tyler Hannan

|||⊹ ClickHouse

ClickHouse Core

ClickHouse Core Roadmap 2023

https://github.com/ClickHouse/ClickHouse/issues/44767

- **Query analysis and optimization**
  - Enable Analyzer by default
  - Correlated subqueries (with decorrelation)
  - Automatically select optimal JOIN algorithms
- **Formats and integrations**
  - Support for Apache Iceberg
  - Optimization of reading for Parquet
- **Security and access control**
  - Bcrypt or other PBKDF
  - Data masking in row-level security
  - JWT support

- **Separation of storage and compute**
  - Parallel replicas (production readiness)
  - Parallel replicas with custom sharding key
  - Optimization of high-cardinality operations
- **Data Storage**
  - Asynchronous inserts by default
  - Lightweight deletes (production readiness)
  - Lightweight updates
  - Transactions for ReplicatedMergeTree

- **Testing and hardening**
  - Green CI: https://aretestsgreenyet.com/
  - More complete fuzzing coverage
  - Fully-static ClickHouse builds
  - Embedded documentation

Faster Data Onboarding

Core recent features and roadmap

Asynchronous inserts

Ensure deduplication when using async_insert

✅ async_insert setting

✅ dedup with async_insert

Datalakes integrations

Integrate natively with popular datalakes formats

✅ Hudi, Deltalake, Iceberg format support

✅ Optimized Parquet reading

Semi-structured data

Add support for natively storing semi-structured data

PR JSON Object type (experimental)

PR SQL/JSON support

More flexible analytics

Core recent features and roadmap

Analyzer by default

Enabling Analyzer by default opens many possibilities in ClickHouse

- ✅ Implement Analyzer in ClickHouse
- PR Enable Analyzer by default
- PR Simplify existing code
- ● Implement cost based optimizer

Extended JOIN support

Faster JOINs for more use cases; automatic JOIN algorithm selection

- ✅ New JOIN algorithms
- ● Dynamic JOIN algorithm selection and JOIN rewrite (dep. Analyzer)
- ● TPC-H and TPC-DS benchmarks

Indexing / Search

More functionality and seamless migrations / ecosystem integrations

- ✅ Inverted índices (experimental)
- PR More vector search functions
- PR More standard SQL support

Extended data management capabilities

Core recent features and roadmap

Lightweight operations

Delete data frequently without impacting performance

- ✅ Lightweight DELETES **(production-ready in 23.3)**
- PR Lightweight UPDATES

Transactions

Bring ACID properties to use cases important to ClickHouse users

- PR BEGIN TRANSACTION, COMMIT and ROLLBACK statements (experimental)

Query cache

Speed up repeated queries with built-in query cache

- ✅ Query cache **(production-ready in 23.5)**
- ✅ Enable in ClickHouse Cloud

ClickHouse Integrations

Ecosystem Integrations status and roadmap

	Available	Recent/Next
Data Ingestion / Orchestration	✅ S3 ✅ Kafka ✅ DBT ✅ Airbyte ✅ Vector (community) ✅ PostgreSQL **(Supabase webinar 5/2)** ✅ MySQL ✅ Decodable (partner) ✅ Metaplane (partner)	✅ Kafka Connect ✅ Confluent Cloud (partner) ✅ RedPanda (partner) ✅ OpenTelemetry (community) ✅ FluentBit (community) 🔥 Apache Beam (community) 🔶 Google Dataflow (partner) 🔶 Azure Event Hub (partner) ✅ EMQX (partner)
Data Visualization	✅ Superset / Preset ✅ Deepnote (partner) ✅ HEX (partner) **(HEX webinar 5/10)** ✅ Mode (partner) 🔥 Grafana (partner) 🔥 Metabase	✅ Tableau (community) 🟡 GCP Looker (partner) 🟡 AWS QuickSight 🔶 Azure Power BI (partner)
Language Clients	🔥 Go 🔥 Python 🔥 Node.js ✅ Java ✅ C# (community)	🔶 Ruby (community) 🔶 Rust (community)
SQL Clients	✅ DBeaver ✅ Datagrip	

Legend

✅ Completed
🟡 In development
🔶 Coming Next / Evaluating
🔥 Recently improved

See 75+ integrations on

https://clickhouse.com/docs/en/integrations

Favourite Features

ClickHouse

session_timezone

```
:) SELECT now()

┌───────────────────now()─┐
│ 2023-06-29 13:11:16 │
└─────────────────────────┘


:) SELECT now('America/Los_Angeles')

┌─now('America/Los_Angeles')─┐
│        2023-06-29 04:11:21 │
└────────────────────────────┘
```

session_timezone

```
:) SET session_timezone = 'America/Los_Angeles';

:) SELECT now();



┌───────────────────────now()─┐

│ 2023-06-29 04:11:29 │

└─────────────────────────────┘
```
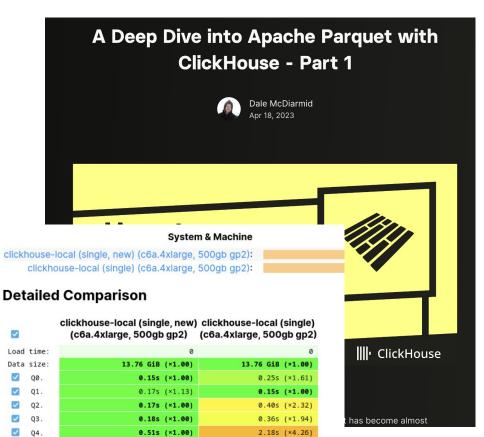
Faster Parquet Reading

23.4

From S3 and URLs.
100 times faster.

23.6

clickhouse-local over partitioned
Parquet files improved 2 times 🔥
clickhouse-local over a single Parquet
file improved 4 times ❤️‍🔥

A Deep Dive into Apache Parquet with
ClickHouse - Part 1

Dale McDiarmid
Apr 18, 2023

System & Machine

clickhouse-local (single, new) (c6a.4xlarge, 500gb gp2):
clickhouse-local (single) (c6a.4xlarge, 500gb gp2):

Detailed Comparison

		clickhouse-local (single, new) (c6a.4xlarge, 500gb gp2)	clickhouse-local (single) (c6a.4xlarge, 500gb gp2)
Load time:		0	0
Data size:		13.76 GiB (×1.00)	13.76 GiB (×1.00)
☑	Q0.	0.15s (×1.00)	0.25s (×1.61)
☑	Q1.	0.17s (×1.13)	0.15s (×1.00)
☑	Q2.	0.17s (×1.00)	0.40s (×2.32)
☑	Q3.	0.18s (×1.00)	0.36s (×1.94)
☑	Q4.	0.51s (×1.00)	2.18s (×4.26)
☑	Q5.	0.84s (×1.00)	6.05s (×7.13)

ClickHouse

t has become almost

SOUNDEX - Developed in 1918, obsolete

```
SELECT soundex('Hello, world!') AS x

 ┌─x────┐

 | H464 |

 └──────┘



SELECT soundex('Hallo, wereld!') AS x

 ┌─x────┐

 | H464 |

 └──────┘
```

Integration with MongoDB 5.1+

```
CREATE TABLE test_mongo ... ENGINE = MongoDB(

  'mongo1:27017', 'test', 'simple_table', 'testuser', 'clickhouse');


SELECT * FROM test_mongo;


SELECT * FROM mongodb(

  'mongo1:27017', 'test', 'simple_table', 'testuser', 'clickhouse',

  'x String, y UInt64, ...');
```

Describing A Release

```
$ sudo docker run --rm clickhouse/clickhouse-server:23.5 clickhouse-local --query
"SELECT * FROM system.contributors ORDER BY name" > contributors_23.5.txt


$ sudo docker run --rm clickhouse/clickhouse-server:23.6 clickhouse-local --query
"SELECT * FROM system.contributors ORDER BY name" > contributors_23.6.txt
```

Describing A Release

```
numbersix at thevillage in ~/clickhouse-mac

$ ./clickhouse local --query "
    SELECT arrayStringConcat(groupArray(line), ', ')
    FROM file('contributors_23.6.txt', LineAsString)
    WHERE line NOT IN (
        SELECT *
        FROM file('contributors_23.5.txt', LineAsString))
        FORMAT TSVRaw"



Chang Chen, Dmitry Kardymon, Hongbin Ma, Julian Maicher, Thomas Panetti, YalalovSM,
kevinyhzou, tpanetti, 郭小龙
```

Thank You!

ClickHouse