

# ClickHouse Update

core, integrations, cloud 2023 roadmap

**Tanya Bragin, Product @ ClickHouse**



# ClickHouse Core

# Core recent features and roadmap

**2022 Roadmap** <https://github.com/ClickHouse/ClickHouse/issues/32513>

**2023 Roadmap** <https://github.com/ClickHouse/ClickHouse/issues/44767>

## Completed

- ✓ Make clickhouse-keeper production ready
- ✓ Support for backup and restore
- ✓ AArch64 hardware support
- ✓ Replicated database engine
- ✓ Type inference for data import
- ✓ Hudi, Delta Lake format support
- ✓ Implement Analyzer in ClickHouse

## In progress

- IP Lightweight DELETE (experimental)
- IP Semistructured Data (experimental)
- IP Support for transactions (experimental)
- IP SQL Compatibility Improvements
- IP JOIN Improvements

**YouTube - Monthly Release Webinars** <https://www.youtube.com/c/ClickHouseDB>

**Sign up for next webinar on Jan 25!**



# Faster data onboarding

## Core recent features and roadmap

### Schema inference

**Add data to ClickHouse without having to define the schema upfront**

- ✓ Schema inference
- ✓ Table structure auto detection
- ✓ Format autodetection

### Semi-structured data

**Add support for natively storing semi-structured data**

- IP JSON Object Type (experimental)
- IP SQL/JSON support
- ... ?

### Datalakes integrations

**Integrate natively with popular datalakes formats**

- ✓ Hudi, Delta Lake format support
- Iceberg format support
- Optimized reading from Parquet



# More flexible analytics

## Core recent features and roadmap

### Analyzer by default

**Enabling Analyzer by default opens many possibilities in ClickHouse**

- IP Implement Analyzer in ClickHouse
  - Enable Analyzer by default
  - Simplify existing code
  - Start enabling new features

### Extended JOIN support

**Faster JOINS for more use cases; automatic JOIN algorithm selection**

- ✓ New JOIN algorithms “direct”, “parallel\_hash”, “full\_sorting\_merge”, “grace\_hash”...)
- Automatic JOIN algorithm selection (dep. Analyzer)
- TPC-H and TPC-DS benchmarks

### Standard SQL commands

**More functionality and seamless migrations / ecosystem integrations**

- ✓ Support for GROUPING SETS
- ✓ Window functions inside expressions
- ✓ GROUP BY ALL
  - Correlated subqueries (dep. Analyzer)



# [Umbrella] Analyzer, Planner migration #42648

[New issue](#)[Open](#)

13 of 42 tasks

kitaisreal opened this issue on Oct 25, 2022 · 16 comments



kitaisreal commented on Oct 25, 2022 · edited by novikd

Member



After [#31796](#) was merged now we have new infrastructure for query analysis and planning. There are a lot of things that new infrastructure brings to us:

1. Better performance of queries that spend considerable time during query analysis, planning stage. Better performance of queries with a lot of JOINS, a lot of subqueries. Example: [Performance degradation on query interpretation](#) [#39996](#).
2. A lot of new SQL features that are now possible to implement (better scalar subqueries, correlated subqueries).
3. Better support for JOINS (Support for indexes, FINAL, SAMPLE).
4. Ability to implement full featured query optimizer on top of new infrastructure.
5. Better usability, better exception messages.
6. Full SQL specification of our SQL extensions is now possible to write.

This ticket will contain tasks that are necessary to implement before we can enable new infrastructure by default.

## Analyzer

- ☒ SELECT (compound\_expression).\*, (compound\_expression).COLUMNS are not supported on parser level. Developer [@evillique](#). [Improve Asterisk and ColumnMatcher parsers](#) [#42884](#)
- ☒ SELECT a.b.c.\*, a.b.c.COLUMNS. Qualified matcher where identifier size is greater than 2 are not supported on parser level. Developer [@evillique](#). [Improve Asterisk and ColumnMatcher parsers](#) [#42884](#)
- ☐ Support function identifier resolve from parent query scope, if lambda in parent scope does not capture any columns.
- ☐ Support group\_by\_use\_nulls. Developer [@novikd](#).
- ☒ Support cache for scalar subqueries. Developer [@SmitaRKulkarni](#). [Support scalar subqueries cache](#) [#43640](#)

### Assignees



novikd



kitaisreal

### Labels

None yet

### Projects

None yet

### Milestone

No milestone

### Development

No branches or pull requests

### Notifications

[Customize](#)

Unsubscribe

You're receiving notifications because you're watching this repository.

8 participants



# Expanded data management capabilities

## Core recent features and roadmap

### Asynchronous inserts

**Ensure deduplication when using `async_insert`**

- ✓ `async_insert` setting
- ✓ dedup when using `async_insert`

### Transactions

**Bring ACID properties to use cases important to ClickHouse users**

**IP** `BEGIN TRANSACTION`, `COMMIT` and `ROLLBACK` statements

Ex. use case: Atomic INSERTs to multiple tables, materialized views, partitions

*In Preview (works with MergeTree only)*

### Lightweight deletes

**Update & delete data frequently without impacting performance**

**IP** Lightweight `DELETE` statement, without need to specify `ALTER`, implemented via delete-masks (experimental)



# Operations and resiliency

## Core recent features and roadmap

### Administration

**Operate ClickHouse more easily, reliably, and at lower cost**

- ✓ clickhouse-keeper (production ready)
- ✓ Flexible memory limits
- ✓ Backup & restore

### New platform support

**Operate ClickHouse on more platforms, e.g. ARM (AArch64)**

- ✓ ARM (AArch64) hardware support is now production ready. 100% functional tests are run in CI for AArch64.

### Hardening & testing

**Additional mechanisms to harden the codebase and builds**

- IP More complete fuzzing coverage
- Fully-static ClickHouse builds
- SQLogicTest integration





# **ClickHouse Ecosystem Integrations**

**Guess how many projects on Github  
integrate with ClickHouse?**

10

100

1000



- **2609** repositories with ClickHouse in their name on Github (excluding forks and ClickHouse itself)
- From which **169** repositories with >50 stars











The screenshot shows a GitHub search interface with the query 'clickhouse'. On the left, a sidebar lists repository statistics: Repositories (2K), Code (505K), Commits (239K), Issues (66K), Discussions (537), Packages (95), Marketplace (3), Topics (50), Wikis (483), and Users (104). Below this is a 'Languages' section with a red border, listing: Go (338), Python (327), Java (255), Shell (183), PHP (102), JavaScript (100), Dockerfile (82), C++ (72), TypeScript (68), and C# (46). The main area displays '2,610 repository results' sorted by 'Best match'. The top results are: 1. ClickHouse/ClickHouse (26.8k stars, C++, Apache-2.0 license, updated 33 minutes ago, 39 issues need help). 2. ClickHouse/clickhouse-go (2.2k stars, Go, Apache-2.0 license, updated 2 days ago, 6 issues need help). 3. ClickHouse/clickhouse-jdbc (1.1k stars, Java, Apache-2.0 license, updated 4 days ago, 1 issue needs help). 4. Altinity/clickhouse-operator (1.2k stars, Go, Apache-2.0 license, updated 3 days ago). 5. mymarilyn/clickhouse-driver (966 stars, Python, updated last month, marked as a sponsor).

# Ecosystem Integrations Team

## Support core, partner, and community integrations





### Core integrations

Core integrations are built or maintained by ClickHouse. These integrations are supported by ClickHouse and live in the ClickHouse GitHub organization.

Name	Logo	Category	Description	Resources
Amazon S3		Data ingestion	Import from, export to, and transform S3 data in flight with ClickHouse built in S3 functions and clickhouse-local.	<a href="#">Documentation</a>
ClickHouse Client		SQL client	ClickHouse Client is the native command-line client for ClickHouse.	<a href="#">Documentation</a>
DBever		SQL client	Free multi-platform database administration tool. Connects to Clickhouse through JDBC driver.	<a href="#">Documentation</a>
dbt		Data ingestion	Use dbt (data build tool) to transform data in ClickHouse by simply writing select statements. dbt puts the T in ELT.	<a href="#">Documentation</a>
Go		Language client	The Go client uses the native interface for a performant, low-overhead means of connecting to ClickHouse.	<a href="#">Documentation</a>
Java		Language client	The Java client is an async, lightweight, and low-overhead library for ClickHouse.	<a href="#">Documentation</a>
Kafka		Data ingestion	Bidirectional integration with Apache Kafka, the open-source distributed event streaming platform.	<a href="#">Documentation</a>
Node.JS		Language client	The official Node.js client for connecting to ClickHouse.	<a href="#">Documentation</a>
Python		Language client	A suite of Python packages for connecting Python to ClickHouse.	<a href="#">Documentation</a>
Superset		Visualization client	Explore and visualize your ClickHouse data with Preset or Apache Superset.	<a href="#">Documentation</a>



### Partner integrations

Partner integrations are built or maintained, and supported by, third-party software vendors.

Name	Logo	Category	Description
Arctype		SQL client	Arctype is a fast and easy-to-use database client for writing queries, building dashboards, and sharing data.
DataGrip		SQL client	DataGrip is a powerful database IDE with dedicated support for ClickHouse.
Deepnote		Data visualization	Deepnote is a collaborative Jupyter-compatible data notebook built for teams to discover and share insights.
Grafana		Data visualization	With Grafana you can create, explore and share all of your data through dashboards.

### Community integrations

Community integrations are built or maintained and supported by community members. No direct support is available besides the public github repositories and community Slack channels.

Name	Logo	Category	Description
Airbyte		Data ingestion	Use Airbyte, to create ELT data pipelines with more than 140 connectors to load and sync your data into ClickHouse.
Apache Spark		Data ingestion	Spark ClickHouse Connector is a high performance connector built on top of Spark DataSource V2.

# Ecosystem Integrations status and roadmap

	Available	Recent / Next
<b>Data Ingestion / Orchestration</b>	<ul style="list-style-type: none"> <li>✓ S3</li> <li>✓ Kafka</li> <li>🔥 DBT (<b>webinar Jan 24</b>)</li> <li>🔥 Airbyte</li> <li>✓ Vector (community)</li> <li>✓ PostgreSQL</li> <li>✓ MySQL</li> <li>✓ Decodable (partner)</li> <li>✓ Metaplane (partner)</li> </ul>	<ul style="list-style-type: none"> <li>🔥 Kafka Connect (beta)</li> <li>🟡 Confluent Cloud (partner)</li> <li>🔷 RedPanda Cloud (partner)</li> <li>✓ OpenTelemetry (community)</li> <li>✓ FluentBit (community)</li> <li>🔥 Apache Beam (community)</li> <li>🔷 Google Dataflow (partner)</li> <li>🔷 Azure Event Hub (partner)</li> <li>✓ EMQX (partner)</li> </ul>
<b>Data visualization</b>	<ul style="list-style-type: none"> <li>🔥 Grafana (partner) (<b>webinar Jan 26</b>)</li> <li>✓ Superset / Preset</li> <li>✓ Deepnote (partner)</li> <li>✓ HEX (partner)</li> </ul>	<ul style="list-style-type: none"> <li>🔥 Metabase (community)</li> <li>✓ Tableau (community)</li> <li>🟡 GCP Looker (partner)</li> <li>🔷 Azure Power BI (partner)</li> <li>🔷 AWS QuickSight</li> </ul>
<b>Language clients</b>	<ul style="list-style-type: none"> <li>🔥 Go</li> <li>✓ Java</li> <li>🔥 Python</li> <li>🔥 Node.js</li> </ul>	<ul style="list-style-type: none"> <li>✓ C# (community)</li> <li>🔷 Ruby (community)</li> <li>🔷 Rust (community)</li> </ul>
<b>SQL clients</b>	<ul style="list-style-type: none"> <li>✓ DBeaver</li> <li>✓ Datagrip</li> </ul>	

## Legend

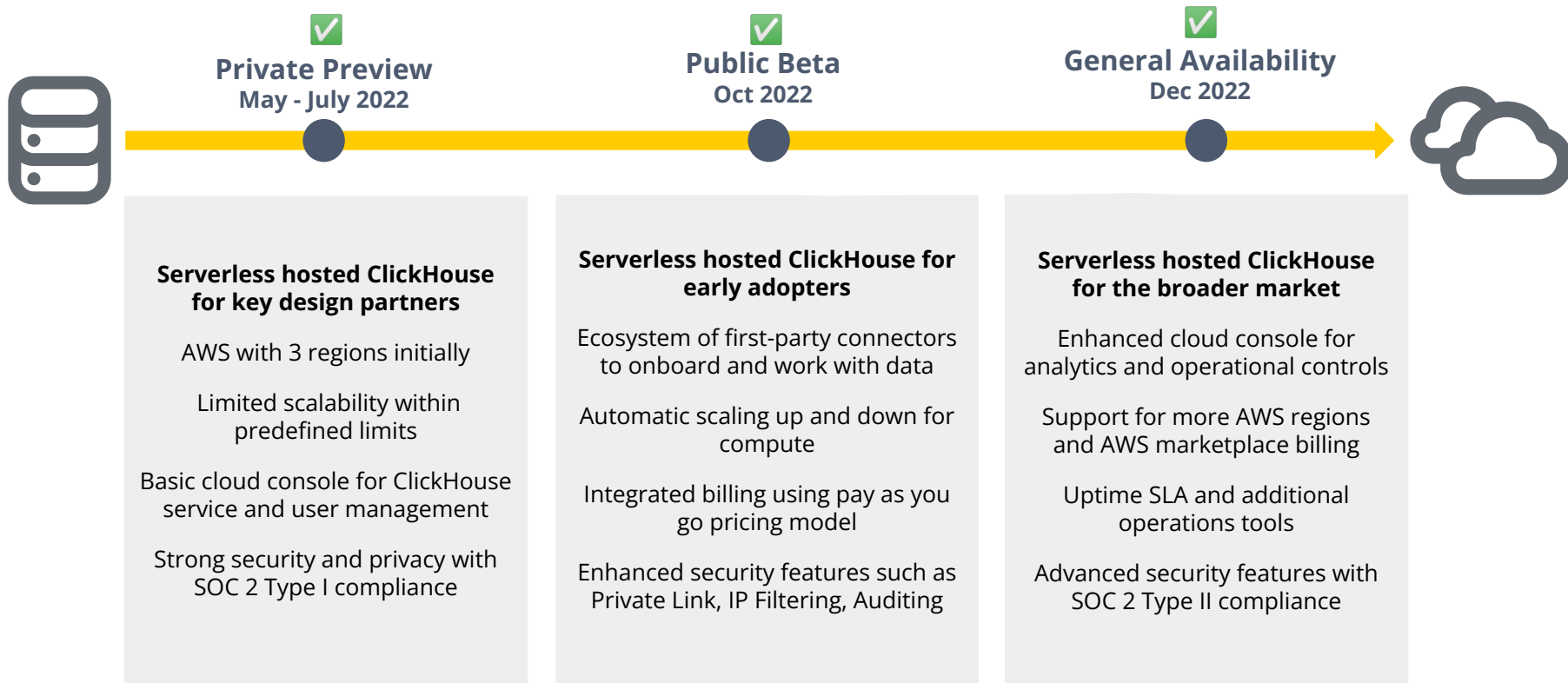
- ✓ Completed
- 🟡 In development
- 🔷 Coming Next / Evaluating
- 🔥 Recently improved




The background of the image features a dark blue-grey color with several overlapping circles of varying sizes and opacities, creating a layered, cloud-like effect. The text "ClickHouse Cloud" is positioned on the left side, within one of the lighter circles.

**ClickHouse Cloud**

# ClickHouse Cloud Milestones



 Services


 Integrations

 Members

 Activity

 Admin

 Help

 Share feedback

+ New Service

## Create a service

### Cloud provider



### Region

 N. Virginia (us-east-1) ▾

### Service name

### Purpose

- ✓ Designed to handle larger production workloads
- ✓ Unlimited storage with 24 GB+ total memory
- ✓ Usage based pricing with spend limits

AWS Private Link support, Uptime SLAs, and automatic scaling are available for production services only

Cancel

Create Service

## SUMMARY

✕

PLAN  
Serverless



## Analytics

Actions ▼

Connect ▼

Summary Backups Security

✓ Running

Location  
Oregon (us-west-2)

Version  
ClickHouse 22.12

Last successful backup  
21 hours ago

Created  
Sep 30, 2022

Aggregation period 1 minute Time period Last hour ▼

Data stored  
**1.62 GB**

No change in last hour

Data stored over time



Successful queries

• **100%**

Failed queries

• **0%**

SQL queries success over time



Total selects

**30**

Average selects per second: 0.01

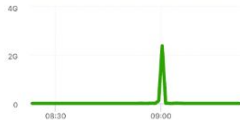
Selects (queries per second)



Total read size  
**2.34 GB**

Average read size per second: 682.53 KB

Read throughput (bytes per second)



Total inserts

**0**

Average inserts per second: 0

Inserts (queries per second)



Total write size

**0 B**

Average write size per second: 0 B

Write throughput (bytes per second)



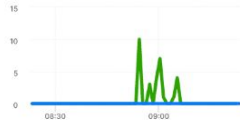
Total selects

• **100%**

Total inserts

• **0%**

SQL statements over time



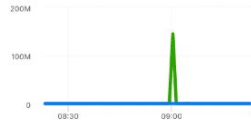
Selected rows

• **145 M**

Inserted rows

• **0**

Number of rows inserted and selected



Can't find what you're looking for?

Submit a request for us to show a new metric on this screen.

New metric

Share feedback



# Integrations

[Request Integration](#)

Improve your workflows by installing integrations or connecting apps to your ClickHouse service. If there's an integration that you would like to see added to this list, please [request it](#).

## Categories

[View all](#)
[Data Ingestion](#)
[Data Visualization](#)
[SQL Client](#)
[Language Client](#)

## Data Ingestion


**Amazon S3**

Import from, export to, and transform S3 data in flight with ClickHouse built-in S3 functions.

Core

[View Integration →](#)

**Kafka**

Integration with Apache Kafka, the open-source distributed event streaming platform.

Community

[View Integration →](#)

**Airbyte**

Use Airbyte, to create ELT data pipelines with more than 140 connectors to load and sync your data into ClickHouse.

Community

[View Integration →](#)

**dbt**

Use dbt (data build tool) to transform data in ClickHouse by simply writing SQL statements.

Core

[View Integration →](#)

**Vector**

A lightweight, ultra-fast tool for building observability pipelines with built-in compatibility with ClickHouse.

Partner

[View Integration →](#)

**PostgreSQL**

Perform SELECT and INSERT operations on data stored on a remote PostgreSQL server directly from ClickHouse

Core

[View Integration →](#)

**MySQL**

Perform SELECT and INSERT operations on data stored on a remote MySQL server directly from ClickHouse

Core

[View Integration →](#)

## Data Visualization


**Grafana**

With Grafana you can create, explore and share all of your data through dashboards.

Partner

[View Integration →](#)

**Superset**

Explore and visualize your ClickHouse data with Apache Superset.

Core

[View Integration →](#)

**Deepnote**

Deepnote is a collaborative Jupyter-compatible data notebook built for teams to discover and share insights.

Partner

[View Integration →](#)

**HEX**

Hex is a modern, collaborative platform with notebooks, data apps, SQL, Python, no-code, R, and so much more.

Partner

[View Integration →](#)




Saved queries

New Query

Search queries

Most referenced GH issu...

US Community Message...

Load Average

Memory Resident

Longest running queries

Running Queries

Finished Queries

OLD Community Messag...



Most referenced ...



US Community M...



Analytics

Most referenced GH issues (by issue num)

default

Run

Save

```
1 SELECT count, toInt32(gh_issue) as gh_issue_num, gh_issue_state, gh_issue_name, gh_issue_url, gh_issue_labels
2 FROM default.gh_issues_links
3 INNER JOIN gh_issues AS table2
4 ON gh_issue_num = table2.gh_issue_num AND gh_issue_state = 'open'
5 ORDER BY gh_issues.gh_issue_num
```

Search results...

Elapsed: 0.116s

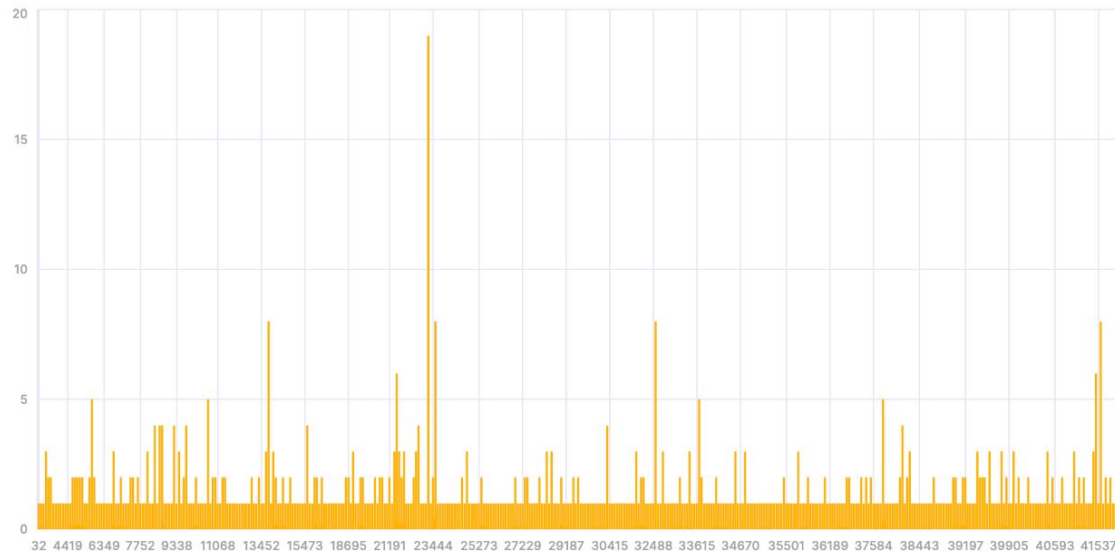
Read: 83,556 rows (7.99 MB)

Table

Chart

...

Most referenced GH issues (by issue num)



General

Advanced



Select chart type

Bar

Specify columns for the chart

Search available columns

gh\_issue\_state string

gh\_issue\_name string

gh\_issue\_url string

gh\_issue\_labels string

x-axis (any)

gh\_issue\_num number

y-axis (number)

count number

DRAW COLUMNS HERE

# ClickHouse Cloud

## Expansion to more cloud providers in 2023

### Azure & GCP

- Please register interest in private preview



**Thank you!**