ClickHouse as a High-Performance Monitoring System, from zero to hero

DevOpsAI SG 2025

ClickHouse

24.03.26

Welcome!

ClickHouse

Agenda

STAND A CHANCE TO WIN AN R2-D2 LEGO SET!

Scan the QR code and put your name in for the lucky draw!

Meet us at the ClickHouse-sponsored coffee cart during afternoon tea break!

01 Introduction

ClickHouse

Who am I ???

ClickHouse

The definition of observability

ClickHouse

Definition (software)

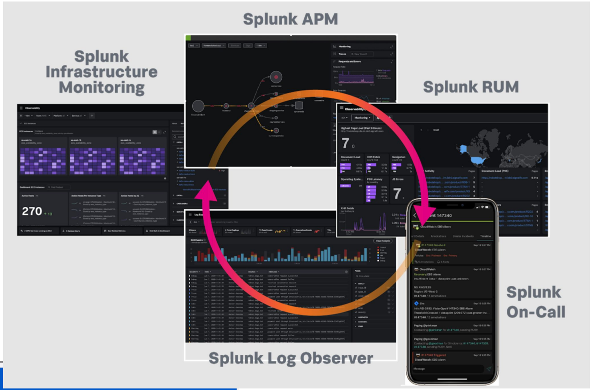
“In software engineering, more specifically in distributed computing, observability is the **ability to collect data** about programs' execution, modules' internal states, and the communication among components. To improve observability, software engineers use a wide range of **logging and tracing techniques** to gather telemetry information, and **tools to analyze and use it**. Observability is **foundational** to **site reliability engineering**, as it is the first step in **triaging a service outage**. One of the goals of observability is to minimize the amount of prior knowledge needed to debug an issue”

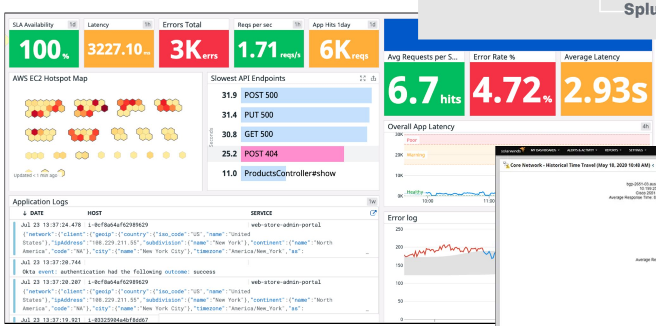
Quoted from *https://en.wikipedia.org/wiki/Observability_(software)*

What is happening in the field?

- There are many observability platforms (data dog, splunk, dyntrace, sumologic), aren't they enough for covering everything?

- **Detour**:
    - costs,
    - transparency,
    - speed (performance),
    - complexity vs flexibility

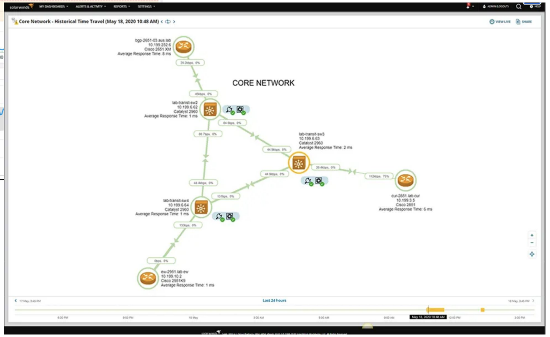




31.9	POST 500
31.4	PUT 500
30.8	GET 500
25.2	POST 404
11.0	ProductsController#show

Application Logs 1w

```
↓ DATE                  HOST                                 SERVICE
Jul 23 13:37:24.478 | i-0cf8a64af62989629                 | web-store-admin-portal
{"network":{"client":{"geoip":{"country":{"iso_code":"US","name":"United
States"},"ipAddress":"108.229.211.55","subdivision":{"name":"New York"},"continent":{"name":"North
America","code":"NA"},"city":{"name":"New York City"},"timezone":"America/New_York","as":
Jul 23 13:37:20.744 |
Okta event: authentication had the following outcome: success
Jul 23 13:37:20.207 | i-0cf8a64af62989629                 | web-store-admin-portal
{"network":{"client":{"geoip":{"country":{"iso_code":"US","name":"United
States"},"ipAddress":"108.229.211.55","subdivision":{"name":"New York"},"continent":{"name":"North
America","code":"NA"},"city":{"name":"New York City"},"timezone":"America/New_York","as":
Jul 23 13:37:19.921 | i-03325904a4bf8dd67
```

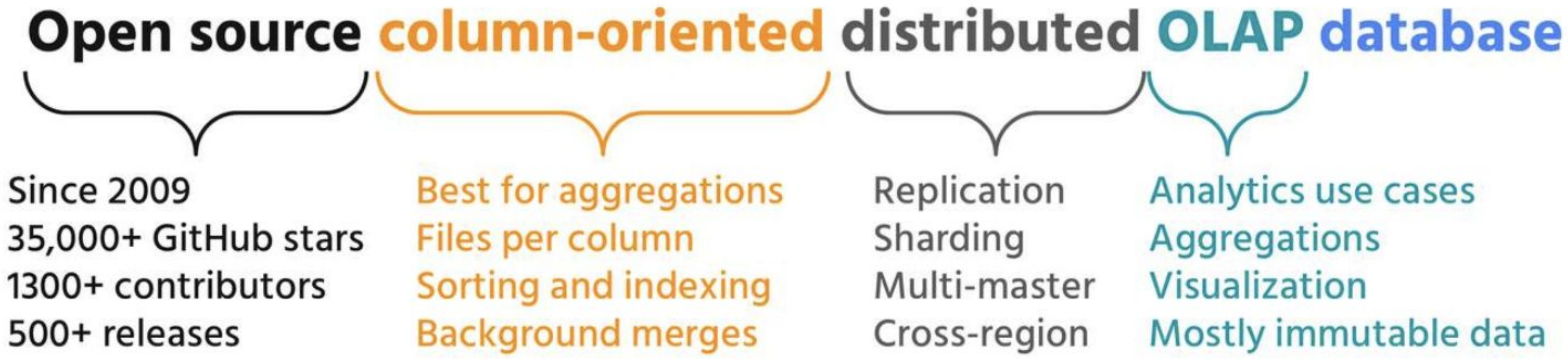

Core Network - Historical Time Travel (May 18, 2020 10:48 AM)
CORE NETWORK
Last 24 hours

Yet another product / platform in the field…

ClickHouse - new kid on the block

- Costs - we might not be the cheapest but definitely not expensive (price calculator)
- Transparency - our pricing is transparent, the data stored is totally visible in table format (SQL), no middleware in between, can fit into workflows through connectors
- Performance - we are fast (ClickBench) in both ingestion and retrieval; storage is efficient (lz4 by default, also available for zstd)
- Complexity - check how Splunk and others' UI were, ClickHouse is simply SQL and tables which are developer friendly (previous slide)

What is ClickHouse actually?



Open source column-oriented distributed OLAP database

Open source	column-oriented	distributed	OLAP
Since 2009	Best for aggregations	Replication	Analytics use cases
35,000+ GitHub stars	Files per column	Sharding	Aggregations
1300+ contributors	Sorting and indexing	Multi-master	Visualization
500+ releases	Background merges	Cross-region	Mostly immutable data

Objective of the workshop

Illustrate what a typical (simplified) observability workflow looks like and how ClickHouse fits into the flow.

Throughout the journey:

- Understand the basics of ClickHouse (e.g. Schema, Materialized Views)
- Extensibility through integrations (e.g. with Python client and Grafana)

02 Pre Requisites

ClickHouse

Tools before you start (15 min)

Instructions (for Mac only) available at https://shorturl.at/BO0HO

- Docker Desktop
- Docker Compose (for older version of Docker Desktop that does not come with `compose`)
- (optional) Git
- (optional) Python3
- A modern web browser (chrome, firefox)

And...

We are all GOOD !

03

ClickHouse fundamentals

ClickHouse

Understanding Columnar Storage

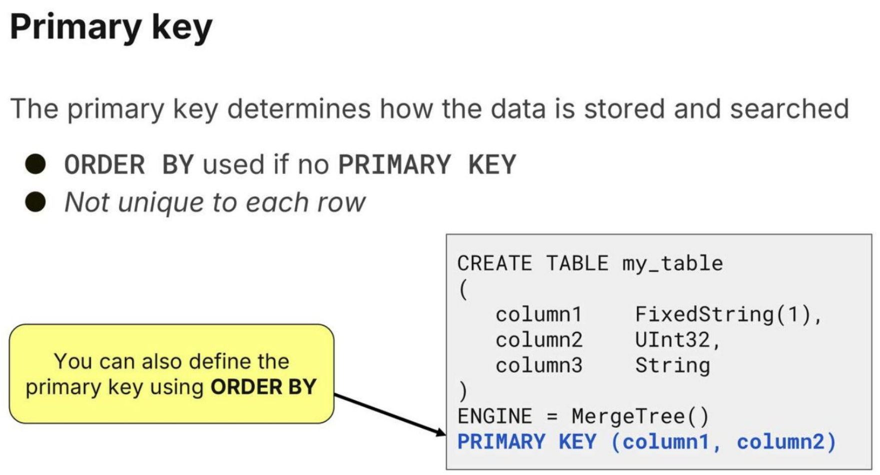
- One reason ClickHouse is so fast is because it is *column-oriented*
- But you need to understand columnar storage to get the benefit

So what does it mean?

Concepts

- In ClickHouse, per table required a table engine; typically we will be using MergeTree(s) which is optimized for high performance inserts and queries (should be good enough for most cases)

- Primary key is different...



Primary key

The primary key determines how the data is stored and searched

- `ORDER BY` used if no `PRIMARY KEY`
- *Not unique to each row*

You can also define the primary key using **ORDER BY**

```
CREATE TABLE my_table
(
   column1    FixedString(1),
   column2    UInt32,
   column3    String
)
ENGINE = MergeTree()
PRIMARY KEY (column1, column2)
```

Concepts (cont’)

- Data are retrieved in Granule(s)

  - **granule** – a logical breakdown of rows inside an uncompressed block; default is 8,192 rows
  - **primary key** – the sort order of a table
  - **primary index** – an in-memory index containing the values of the primary keys of the first row of each granule
  - **part** – a folder of files consisting of the column files and index file of a subset of a table’s data

A *granule* is the smallest indivisible amount of data that ClickHouse reads when searching rows

04

Data Modeling - Basics

Schema

- In ClickHouse, table Schema (aka DDL) is important.
- It is a big-data oriented SQL db (not **noSQL** db) check it out on Elasticsearch too!
- It is possible to alter the existing columns but have to pay a price; it is done through \`mutation\`
- Instead of updating the schema, we could consider Materialized views

Materialized View

Use cases:

- If your source / raw data required to be processed (e.g. extraction of certain fields from a json data structure)
- If the existing table's Index (sorting-key / primary-key) does not meet a use case's requirements
- To filter out data from the source table into dedicated tables (e.g. forward a data row based on the `region` field's value, each region would have its own table)
- To consolidate various source tables' data into a single target table

Materialized View

- General link : https://shorturl.at/o94Oy
- Refreshable MV : https://shorturl.at/SbbPw
- Incremental MV : https://shorturl.at/IxJRy

08

FAQ

ClickHouse

Who am I ???

ClickHouse

STAND A CHANCE TO WIN AN R2-D2 LEGO SET!

Scan the QR code and put your name in for the lucky draw!

Meet us at the ClickHouse-sponsored coffee cart during afternoon tea break!

Thank you!

ClickHouse