



Online Meetup

July 9th 2023

CONTENTS

01 / PART ONE
About Greptime

02 / PART TWO
Why Greptime

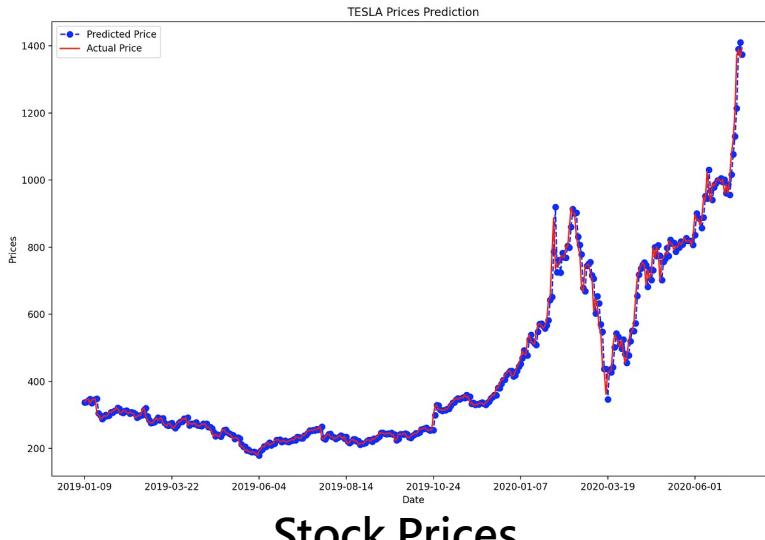
03 / PART THREE
GreptimeDB v0.3

04 / PART FOUR
GreptimeCloud Demo

About Greptime

What is Time Series Data?

Collected over time and is ordered chronologically. Typically, these data points track changes of a particular variable and consist of successive measurements over a fixed time interval made from the same source.



Stock Prices



Weather



Observability



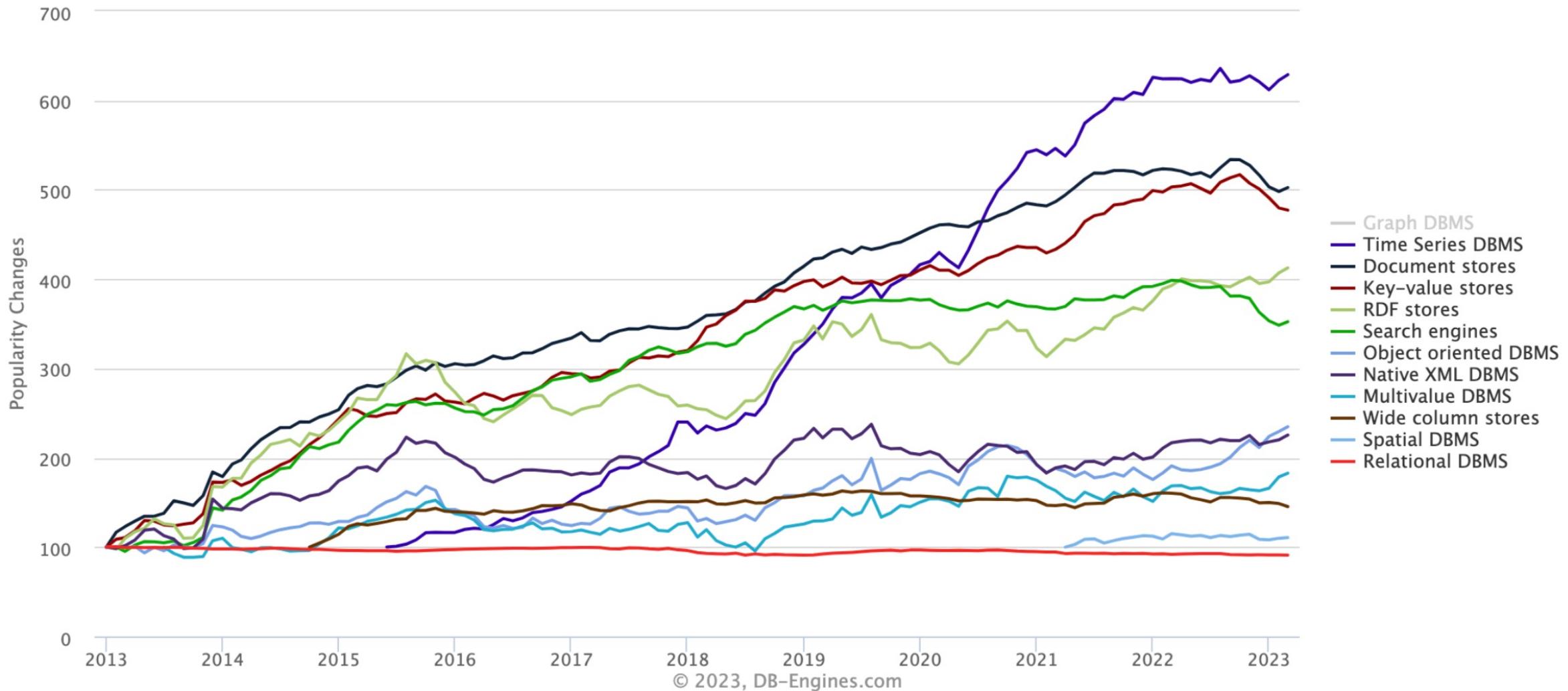
IoT devices

Time	IoT		Connected-Cars	
	Devices (billion)	Data (TB)	Devices (million)	Data (TB)
2022	13.1	14.1B	237	1.5B
2025	19.1	19.1B	402	2.6B

Source: Statista

What is Time Series Data?

Complete trend, starting with January 2013



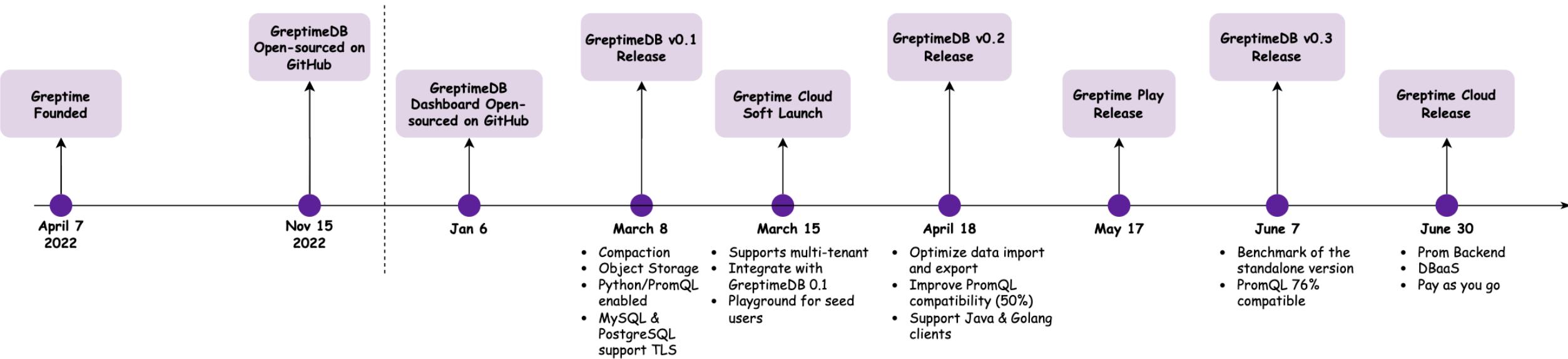
About me



Requirements for a TSDB

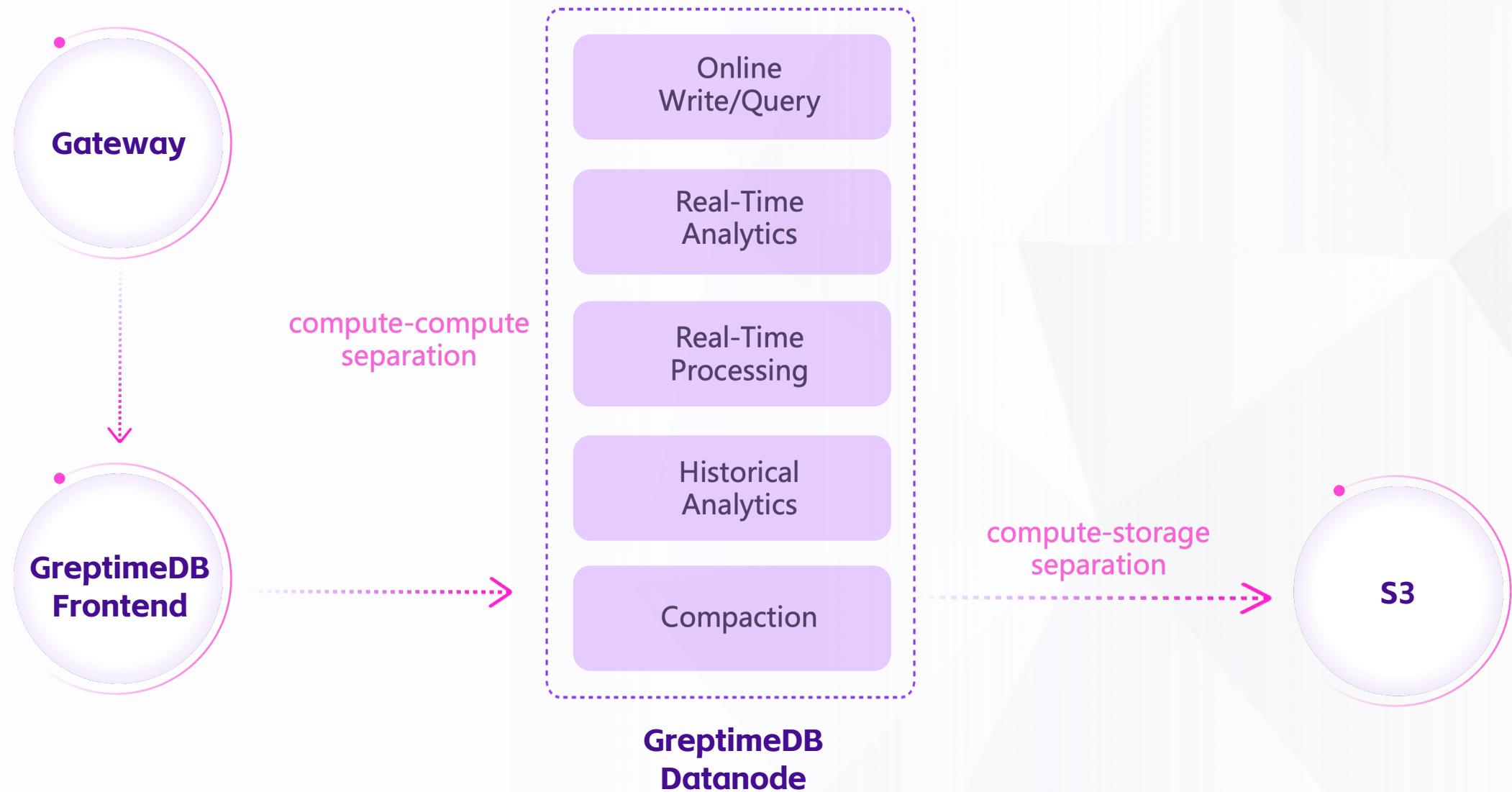
- Process massive, high-cardinality time series data points at a reasonable cost
- Handle at least 100K queries and 500M data points per second in real-time
- Meet various high requirements for data analytics

Milestones

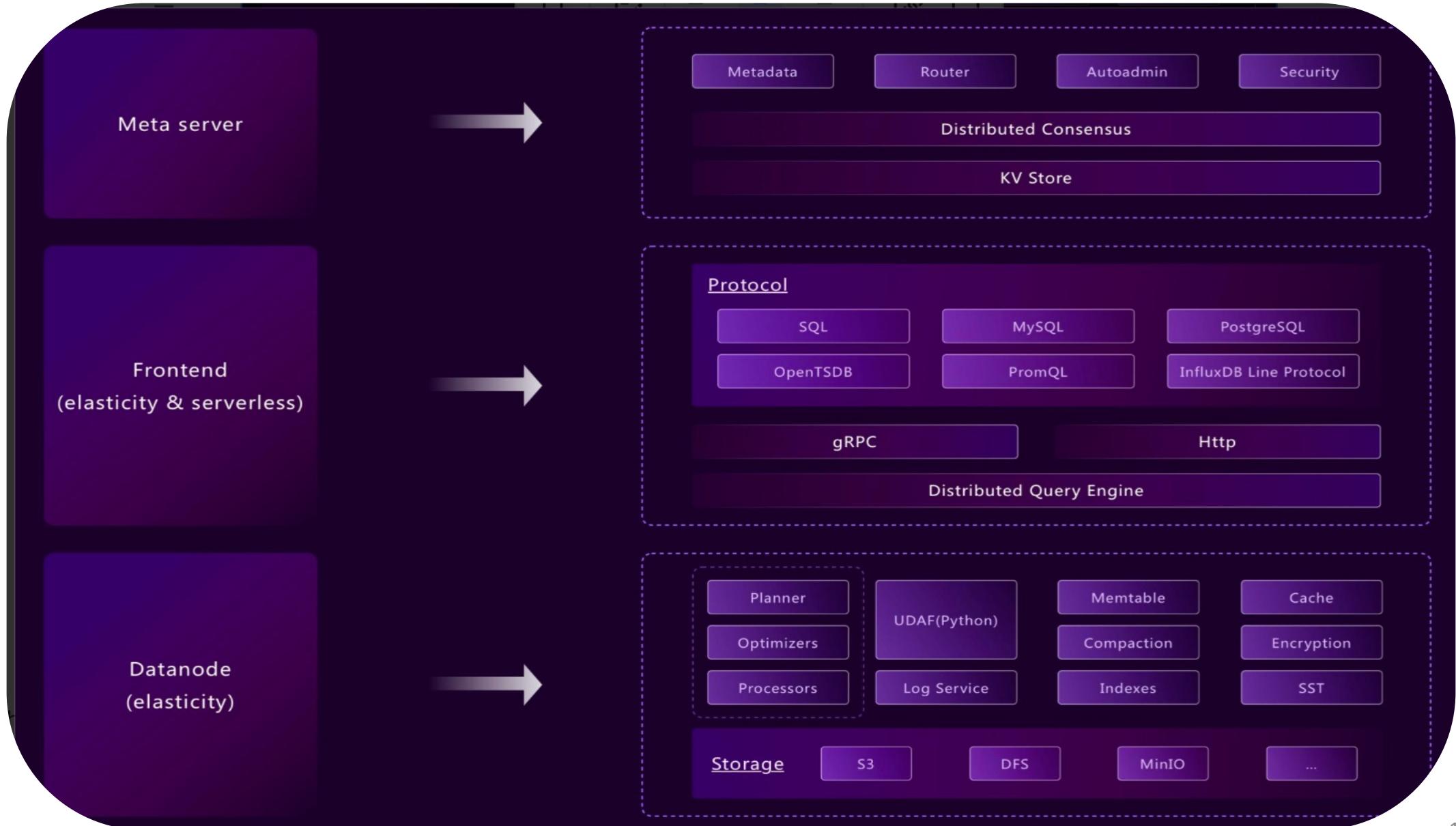


Why Greptime

Our Architecture



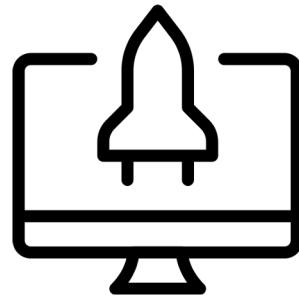
Flexible Architecture Supports Various Deployment Strategies





Cost effective

- AWS S3 and Azure Blob
- Adaptive compression
- Dictionary compression
- The Chimp algorithm



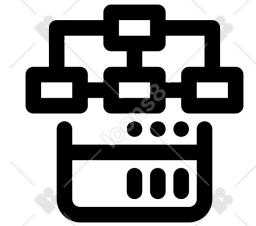
High Performance

- LSM Tree
- Data Sharding
- WAL
- Vectorized execution
- Distributed parallel processing
- Smart indexing
- Massively Parallel Processing (MPP)
- Apache Parquet
- Combining Cost-Based Optimization(CBO)
- User-defined hints



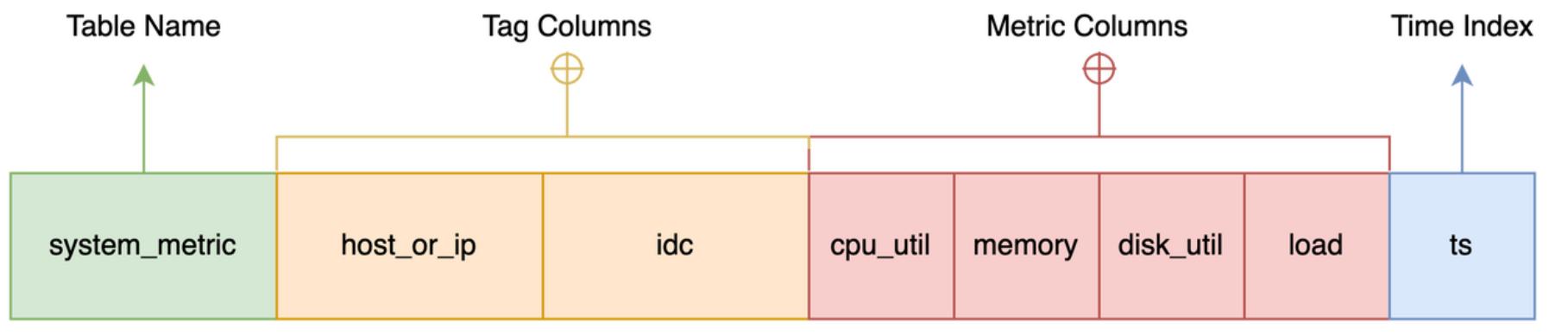
Easy to use

- K8s operator
- Command-line tool
- Embedded dashboard
- Various Protocols
- SDKs for different language
- PromQL
- SQL
- Python



Automatic Schema

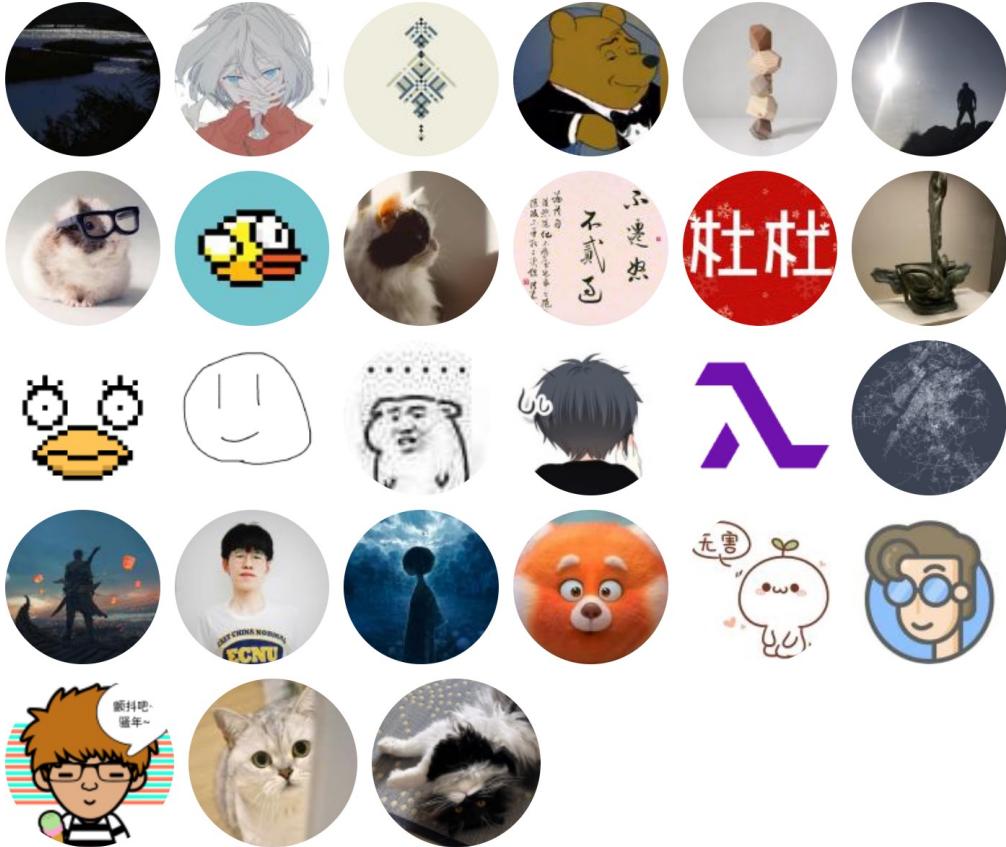
- Time-series table (see below)
- Rows and columns
- Tags and fields
- Timestamp



- **Table name:** often the same as indicator name, such as system_metric here.
- **Time index column:** required and normally used to indicate the data generation time in this row.
- **Metric Column:** data indicators collected, generally change with time, such as cpu_util and memory_util, etc. GreptimeDB adopts a multi-value .
- **Tag Column:** labels attached to the collected indicators, generally to describe a particular characteristic of these indicators.

GreptimeDB v0.3

Contributors



200⁺ PRs

120⁺ Feats

20⁺ Refactors

Three Main Enhancements

All performance improvement and function updates are based on the distributed version, offering qualities that the standalone version does not possess, such as scalability, high availability, and fault tolerance.

1. Performance Optimization

Supports Region-level high availability and provides fast fault-tolerant switchover scheduling. The write performance of the distributed version is also optimized.

2. Querying Optimization

Query optimization, improvement of important SQL queries (e.g. TopK) and optimization of data compaction strategy to boost querying speed.

3. Stability Improvement

The Procedure framework guarantees the consistency of multi-step operations. A finer-grained Hybrid-flush strategy improves write stability, and add more metrics to improve system Observability, supporting tools such as the Tokio console.

What to expect for GreptimeDB v0.4?

1. Supports DDL statements

Create Table, Drop Table, etc. to be connected to the Procedure framework to ensure correct execution of distributed multi-step operations.

2. Querying Optimization

Asynchronous recompression and indexing to improve query performance

3. Improve SQL query performance

Pushdown of common operators in SQL to improve SQL query performance

GreptimeCloud Demo



AMA
ask me anything

Join us



Slack



JiachunFeng





Thanks for joining