

## Running Zabbix with 1M NVPS

Explore the architectural shift from traditional RDBMS to OLAP DBMS as the high performance history backend for Zabbix LTS 8.0

I'm **Edgar Akhmetshin**,  
**I solve problems** (and make new  
sometimes)

## Lead Technical Support Engineer

Zabbix Certified Expert

Zabbix Certified Trainer

## Architected Zabbix environments since

**20**  
10+ years in high-load monitoring

Predict, don't react

## Meet me

#LiveEvents #ZabbixSummit #OfficialTrainings

#Consultancy #EnvironmentReviews

#PerformanceTuning #SupportTickets

#Community #Telegram

**Reality**

**ZABBIX '26**

**CONFERENCE**

**GERMANY**

Directed by  
**ROBERT B. WEIDE**

**Zabbix with 150k+ NVPS and  
triggers**

## Zabbix LTS 6.0 / 7.0 and 7.4:

- Hybrid monitoring power house
- RDBMS (ACID) + time series
- ElasticSearch (horizontal scaling)

## We expect in Zabbix LTS 8.0?

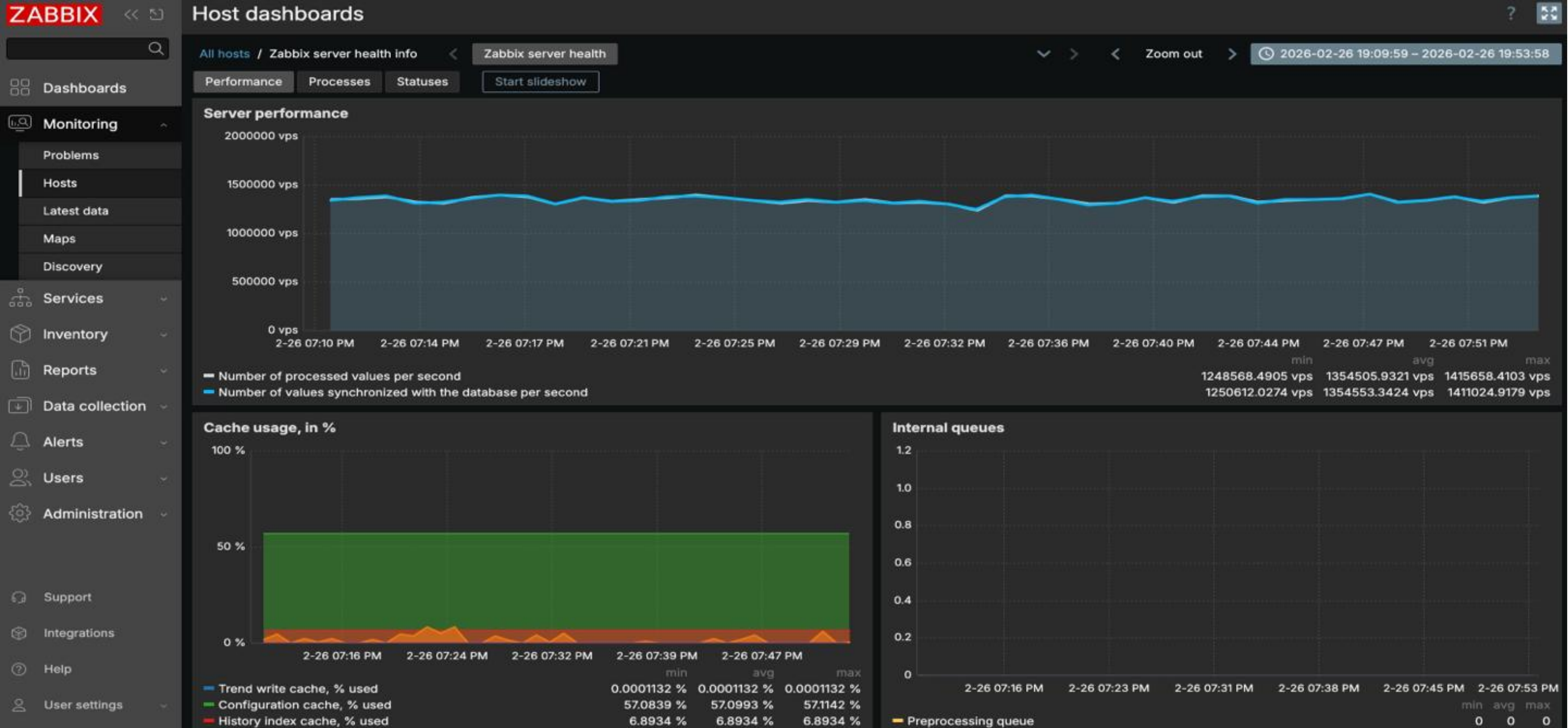
- New storage engines
- Performance

# The ascent

**ZABBIX** '26

CONFERENCE

GERMANY



# Zabbix with 1M\* NVPS and triggers

\*Verified on laptop inside VM. If it runs here, it flies in the DC

# Anatomy

**ZABBIX**'26

CONFERENCE

GERMANY

## Pluggable history backend — from hard-coded SQL/Elastic to a provider framework

Before:

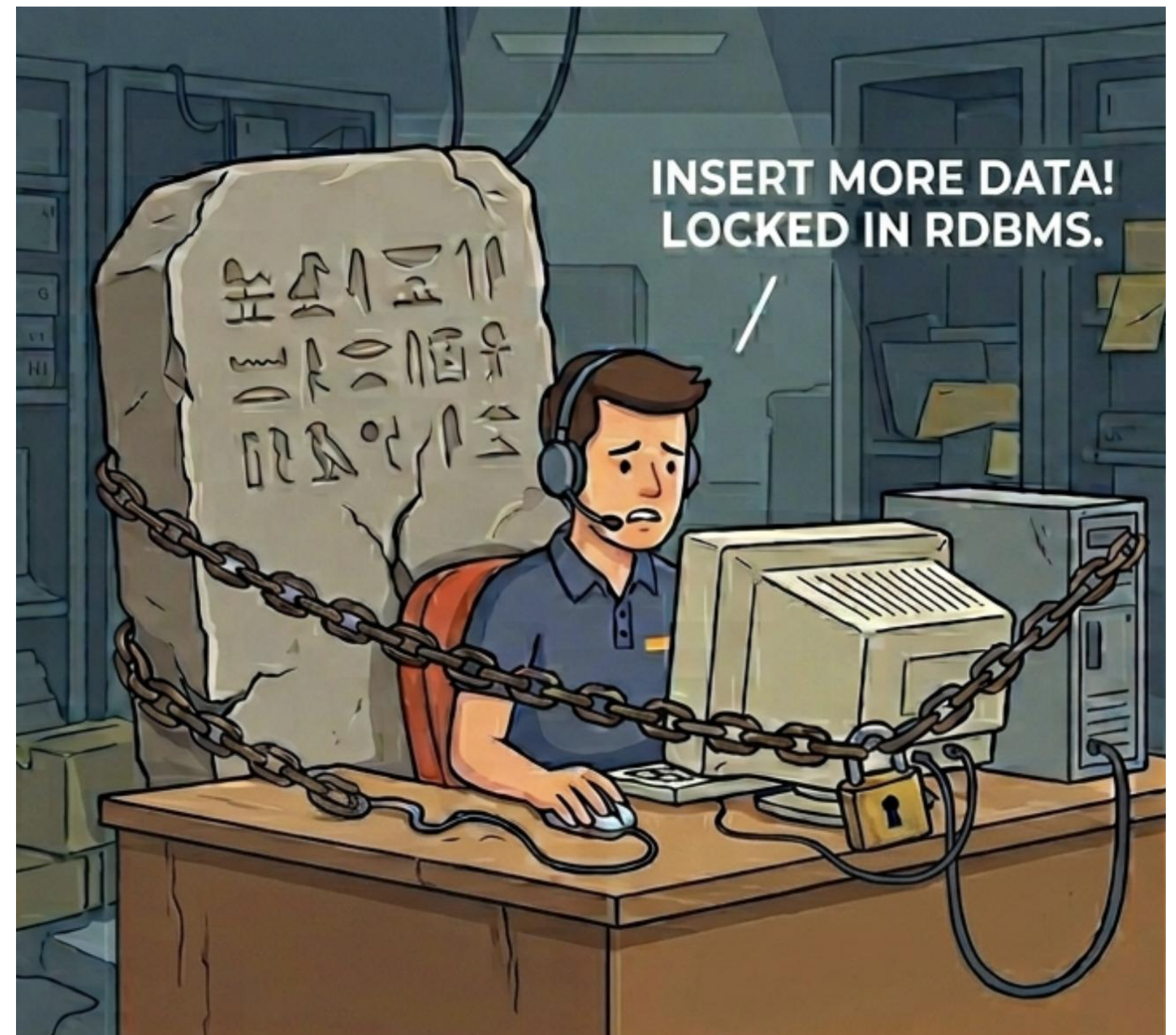
- `HistoryStorageURL=https://es.zabbix.lan:9200`
- `HistoryStorageTypes=uint,dbl,str,log,text`
- `HistoryStorageDateIndex=1`

Issues:

- **Experimental**
- Synchronous
- ElasticSearch only
- RDBMS engines + extensions only

Facts:

- HTTP(S)-based



# Pluggable history backend — from hard-coded SQL/Elastic to a provider framework

After:

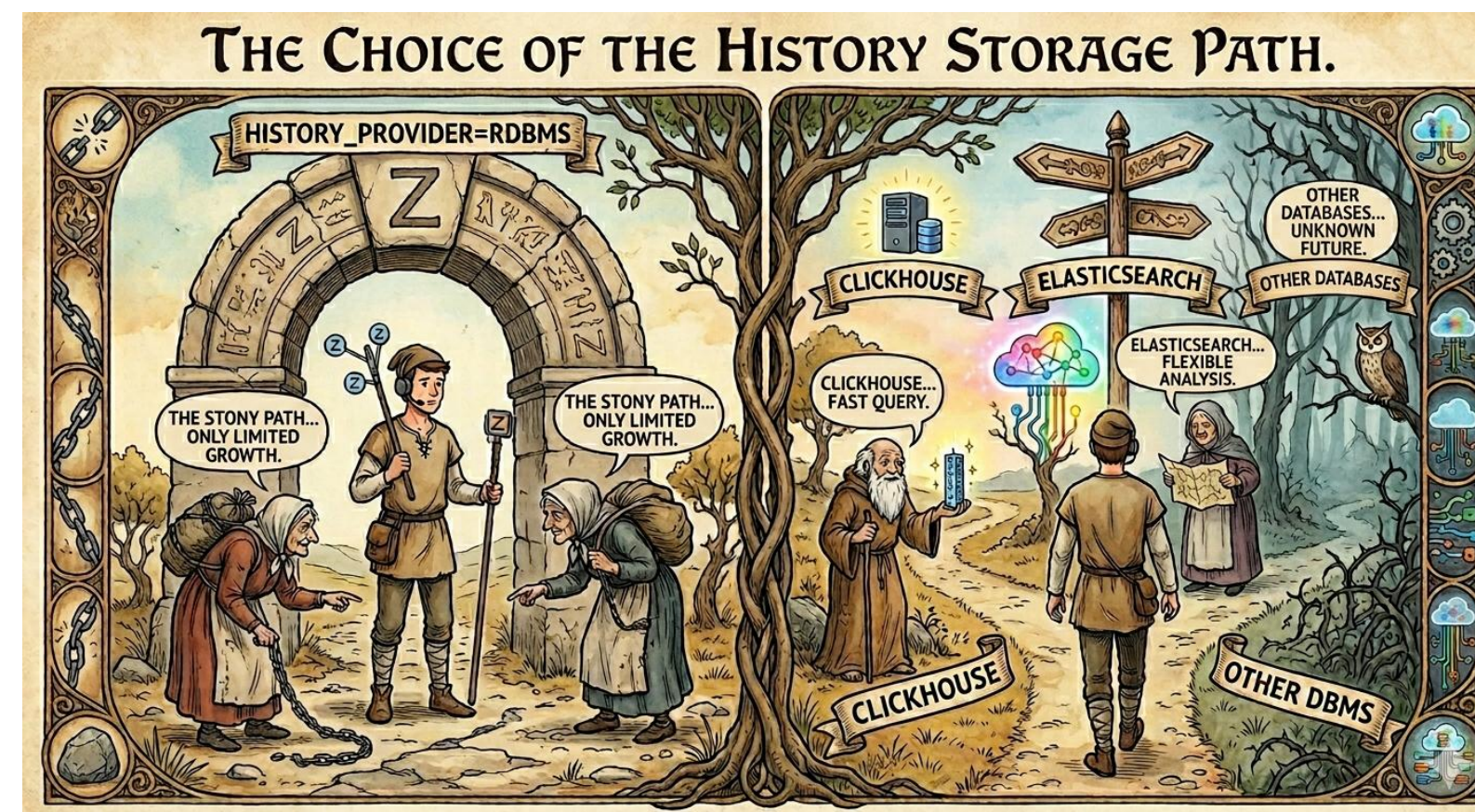
- `HistoryProvider=provider;options`

Pros:

- Asynchronous
- HTTP(S)-based

Examples:

- `HistoryProvider=clickhouse;value_types="uint,text",url=http://ch.zabbix.lan:8123,db=zabbix,username=zabbix,password=isthebest`
- `HistoryProvider=elasticsearch;value_types="uint,text",url=http://es.zabbix.lan:9200...`
- `HistoryProvider=provider;...`



## Pluggable history backend — from hard-coded SQL/Elastic to a provider framework

Why matters:

- **Scale:** future-proof scalability.
- **Speed:** blazing-fast historical queries.
- **Efficiency:** optimized resource utilization.
- **Freedom:** no vendor lock-in.
- **Control:** total data control.

Looking ahead:

- all → ClickHouse
- uint/float → ClickHouse
- str/log/text → Elasticsearch
- everything else → SQL

# FLEXIBLE

DATABASE BACKENDS



## Cache warmup

### From "sequential per-item fetch" to "batched, selector-driven, trait-gated"

- re-read the world per-item history fetches during value-cache warmup

```
[root@frikadelka zabbix]# tail -f zabbix_server.log | grep "SYNC:"  
  
3545706:20251103:123145.538 HISTORY SYNC: processed 1000000 in 320.163 (3123.4 values/sec)  
3369910:20251103:120103.383 HISTORY SYNC: processed 1000000 in 266.764 (3748.6 values/sec)  
...
```

Now:

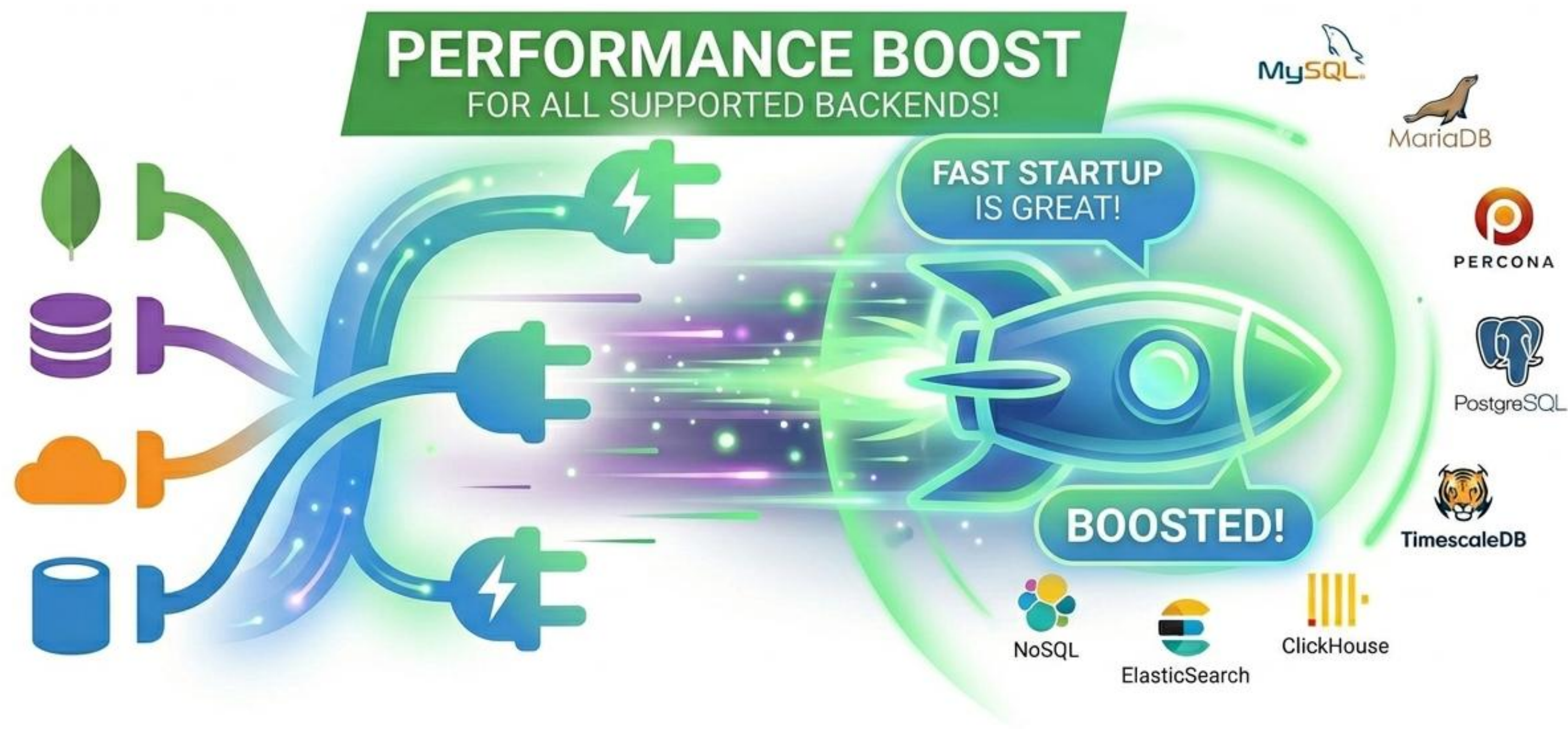
- read only what's about to be evaluated, in batches, against the right backend
- one query returns N items worth of recent rows

```
[root@frikadelka zabbix]# tail -f zabbix_server.log | grep "SYNC:"  
  
2561621:20251127:185551.694 HISTORY SYNC: processed 1000000 in 0.626 (1596557.6 values/sec)  
2561624:20251127:185556.303 HISTORY SYNC: processed 1000000 in 0.605 (1651051.1 values/sec)  
...
```

# Cache warmup

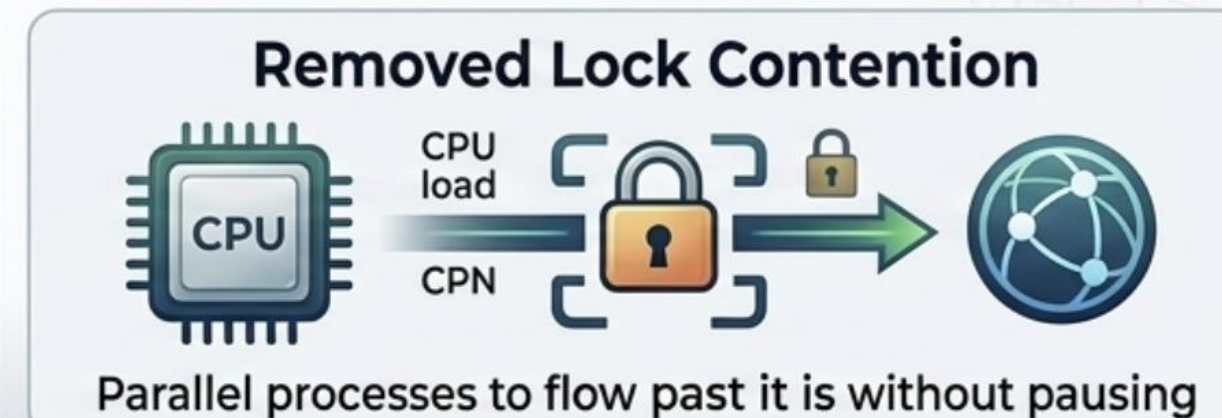
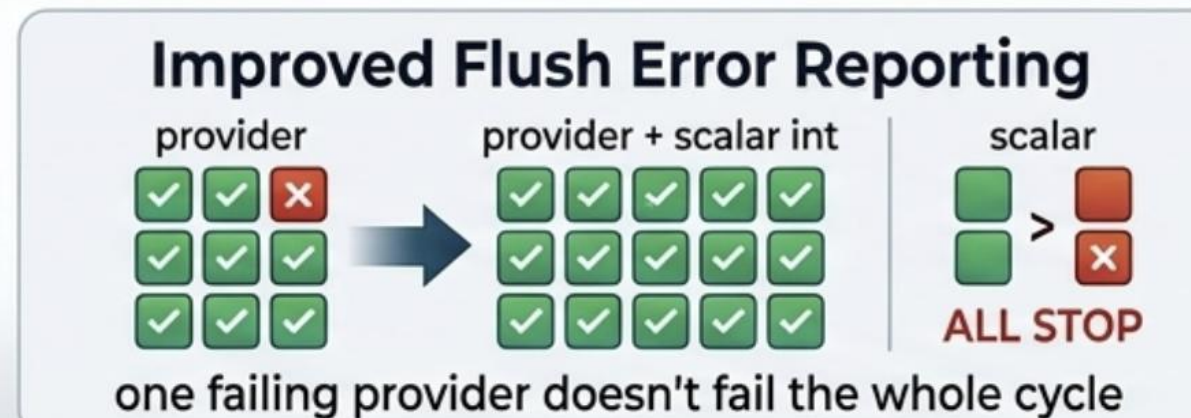
## ZABBIX: INCREASED CACHE WARMUP SPEED

FASTER STARTUP & OPTIMIZED PERFORMANCE FOR ALL BACKENDS

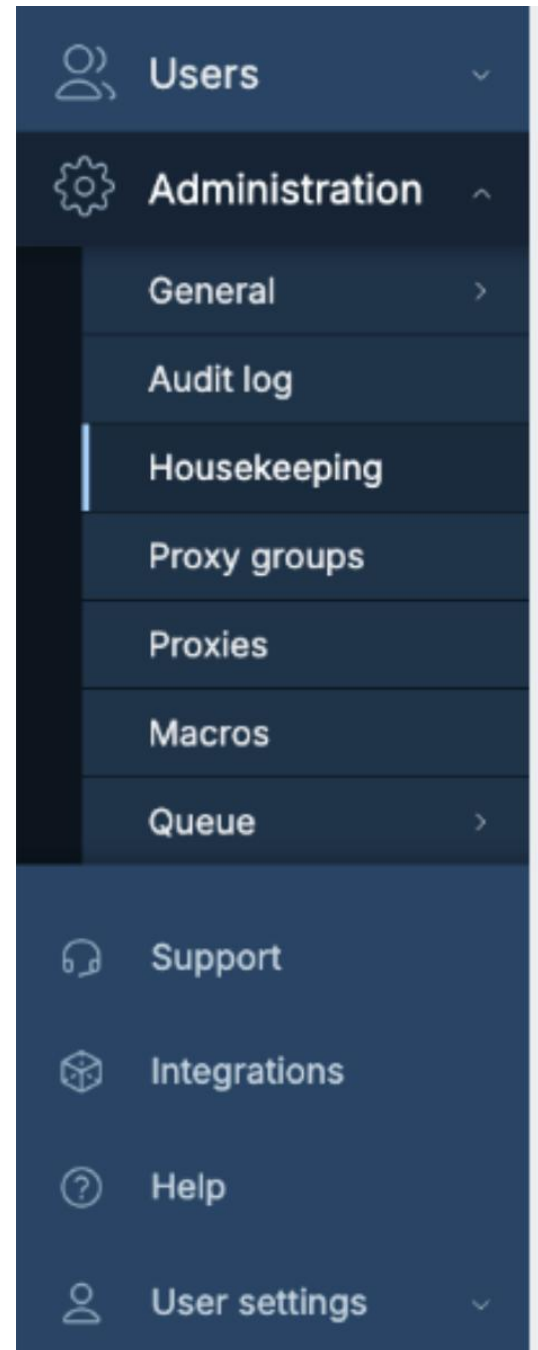


## History sync hot path cleanup

- ZBX\_HC\_SYNC\_MAX raised from 1000 → 2000 (larger per-cycle batch)
- Connection re-use for ElasticSearch(OpenSearch) and ClickHouse backends
- Flush errors now reported via a bitmask instead of a scalar int, so one failing provider doesn't fail the whole cycle
- Removed hash calculation under locks



Stop doing work someone else can do  
for you  
No trends  
No housekeeping



User sessions

Enable internal housekeeping

\* Data storage period

History

Enable internal housekeeping

Override item history period

\* Data storage period

ClickHouse

Data storage period: Numeric (unsigned) 30d

Data storage period: Text 30d

Data storage period: Character 30d

ElasticDB

Data storage period: Numeric (float) Not available

# Mechanics

**ZABBIX** '26

CONFERENCE

GERMANY

## OLAP DBMS – ClickHouse

## Search and analytics engine - ElasticSearch (OpenSearch)

- Familiar setup
- Migration tool
- TLS encrypted connections
- Support throughout the entire product lifecycle

Small



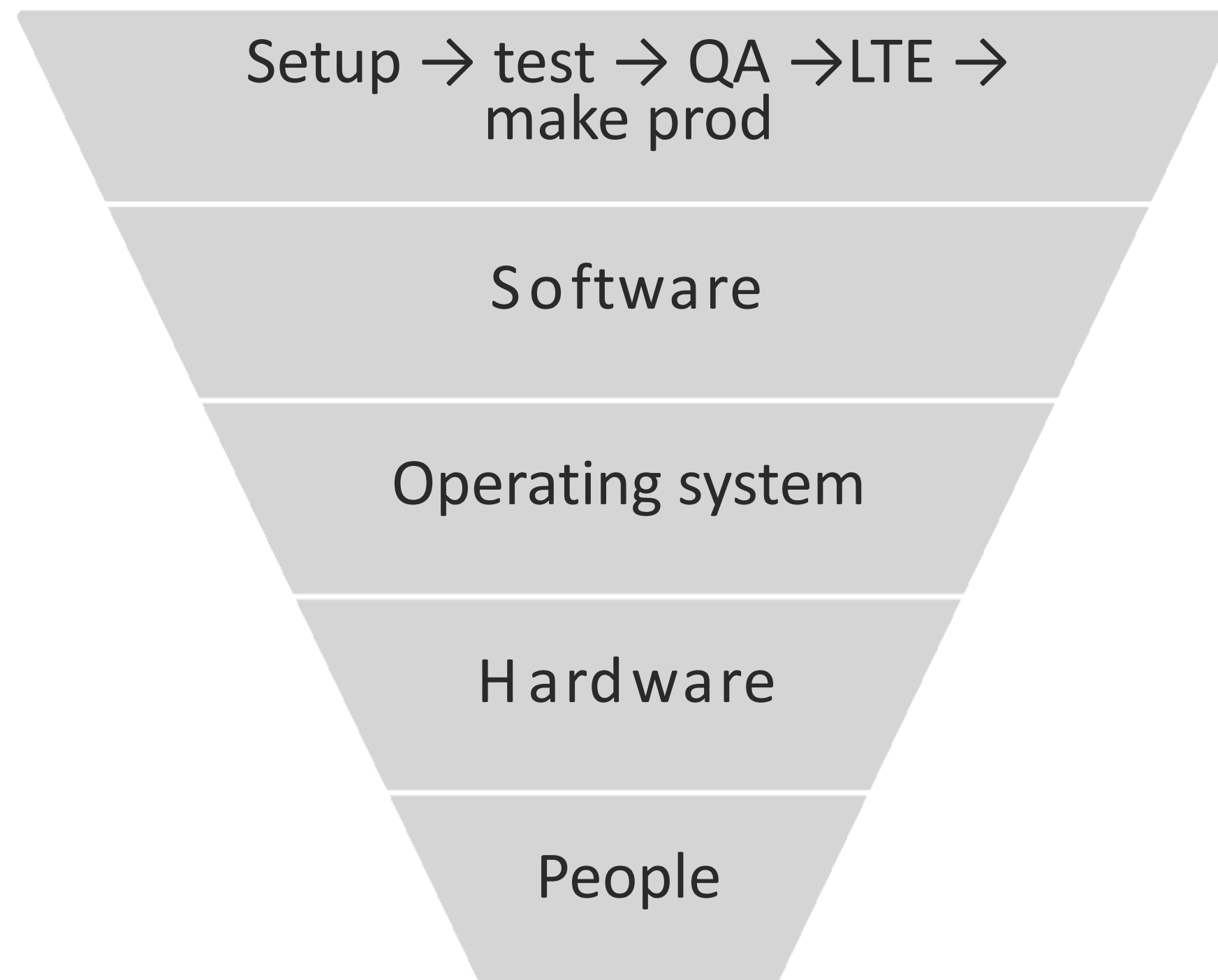
Large



## Checklist

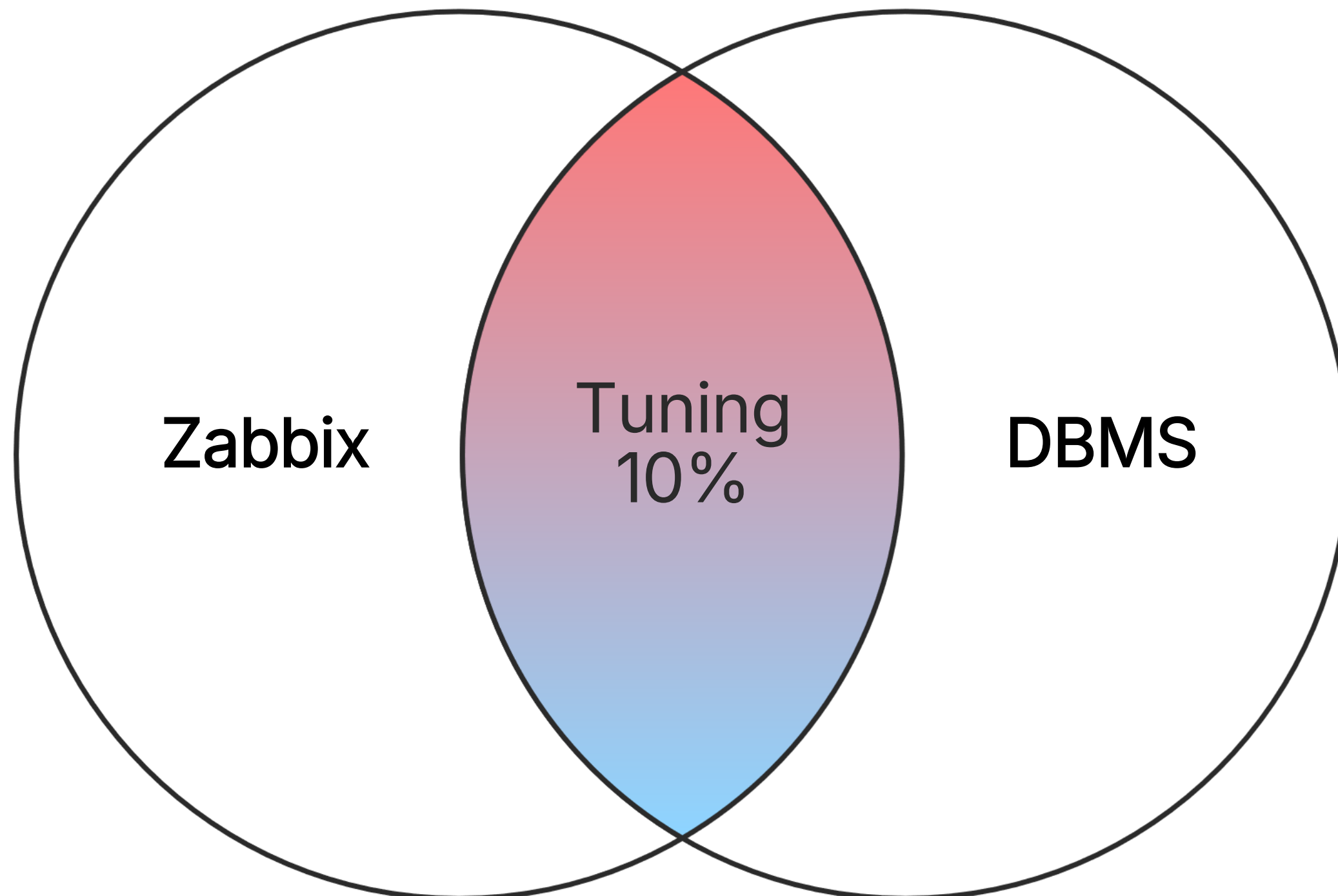
Zabbix LTS 8.0

ClickHouse 25.x+



## Performance

Hardware is the potential.  
Configuration is the permission.



**Some people talk monitoring.**

**We live it — 1 million values at a time**

**ZABBIX**'26

CONFERENCE

GERMANY

**Q&A**

**Thank you!**

**ZABBIX** '26

CONFERENCE

GERMANY