Agenda

- Who we are

- Learnings

- Where we are coming from

- Hints and wishes

- Where we are aiming for (with the help of Aerospike)

VIRTUALMINDS

About Virtual Minds

PRIVACY BY DESIGN

CUTTING EDGE TECHNO-LOGY

Programmatic Supply (SSP)

YL Yieldlab
Premium Programmatic Advertising

Attribution & Marketing Automation

ADCLEAR

Adserving, Targeting, Video & Creative

ADITION technologies AG

Programmatic Buying (DSP)

Active Agent

DMP Verification, Marketplace

ADEX

MODULAR TECH STACK

100% MADE IN GERMANY

AEROSPIKE SUMMIT '20

VIRTUALMINDS

Lessons learned from using Aerospike

- **Use Cases** - New use cases always pop up

- **Speed** - The faster the better

- **Pure C** is a pretty good argument

- **Best practices** - Power to the data, making use of computer and programming best practices leads to good performance

- **Plan for long run** - Some competitors did a good job at first but failed long run

- **Key-value strength** - The more key-value the better Aerospike, redundant data is king

- **Volatility handling** - Small parts of data, highly volatile? -> use Aerospike

- **Efficiency** - We are driven by efficiency, Aerospike Enterprise meets the goal

But wait ... What lead to aerospike?

- We handle big data for > 10 years (started with dynomite)
- Tested different solutions, used different ones, landed at aerospike OSS once, jumped to enterprise later

write request counts with timeouts in buckets
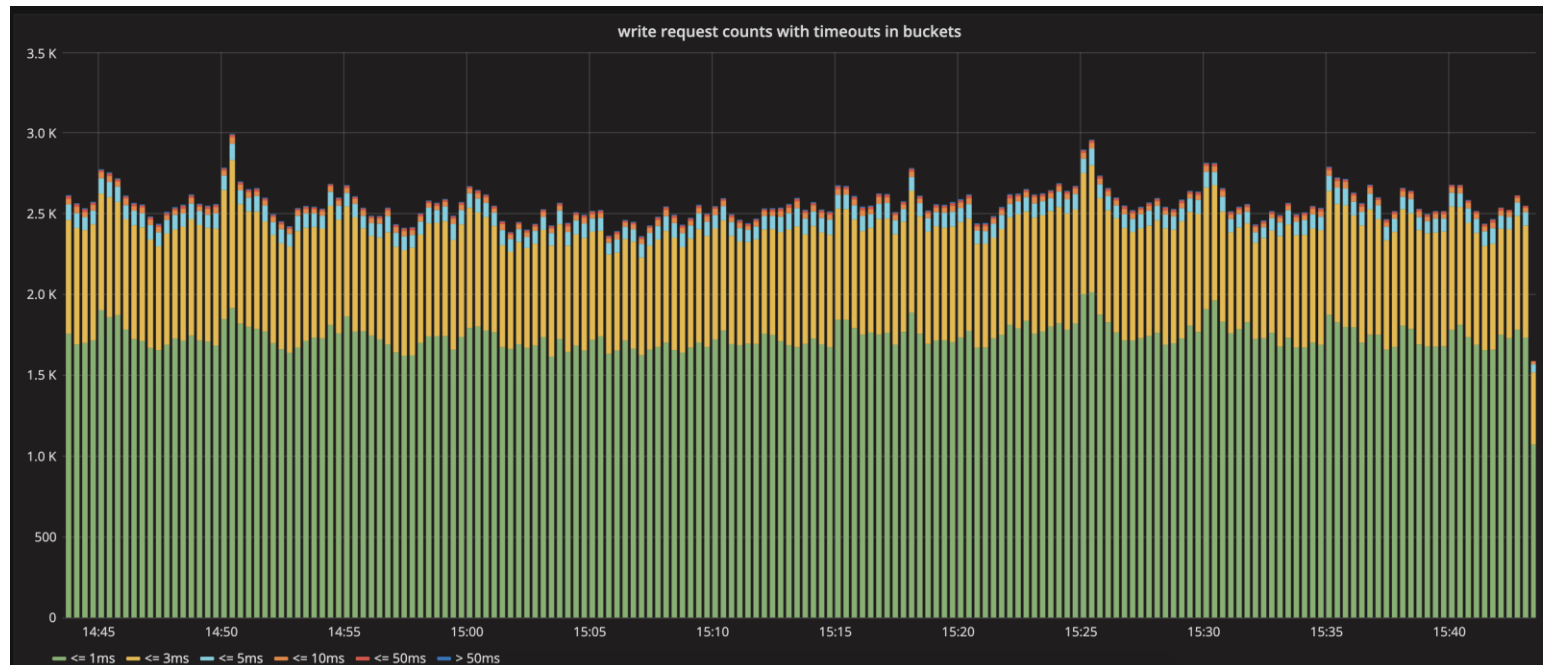
Paradigms/Best practices for data in dev/ops

## **Data development patterns in AdTech (our learnings)**

- Scaling through partitioning
- Embrace redundant/duplicated data
- Simplify queries and aggregations
- Complex queries (like SQL * , JOIN …)
  in < 5 ms on 95% avg doable
- Data on disk and memory for restart scenarios

Alternatives (tested)

- **RIAK**
  - Buggy and no chance to fix
  - Super behaviour at start, bad performance/speed of innovation in the long run
- **Cassandra**
  - Slow read/write performance, especiall when done in parallel
  - Did not meet criteria (99% of all reqs < 5ms avg)
- **Evaluated much more**
  (mongoDB, couchDB, Redis)

	Munin	Cacti	Graphite	OpenTSDB	Ganglia	InfluxDB
Scales well	no	no	sort of	yes	yes	yes
Keeps all data	no	no	no	yes	no	sort of
Creating metrics	easy	sort of	easy	easy	easy	?
Tested	no	yes	yes	yes	no	no

**Local Read Latency**
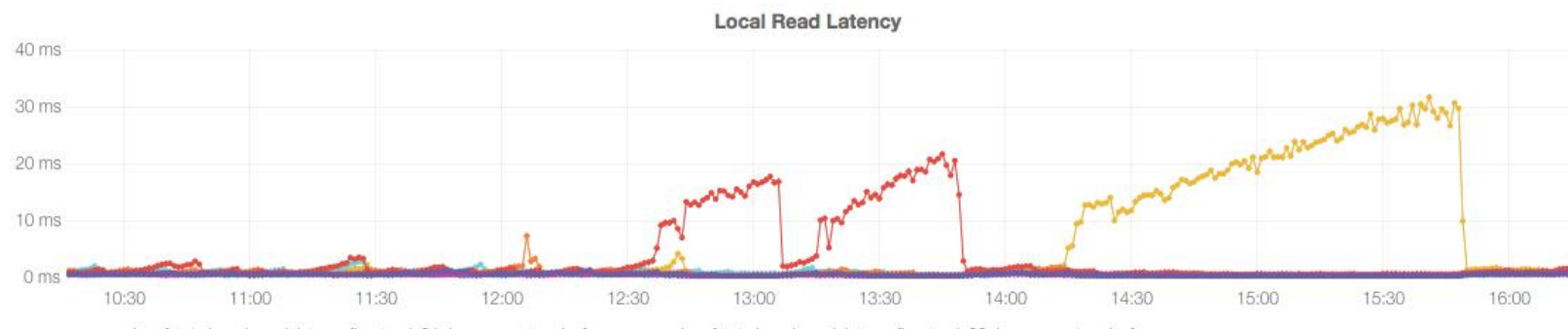
Use Cases

- **Tagging Store**
  - Automatic tagging of users by the adserver
  - Real-time information flow to data store
  - Use tags for eg. Retargeting in below 5 seconds as key KPI

- **DMP Store**
  - Zyclic load of external userdata
  - Make use of the stored attributes for targeting within campaigns

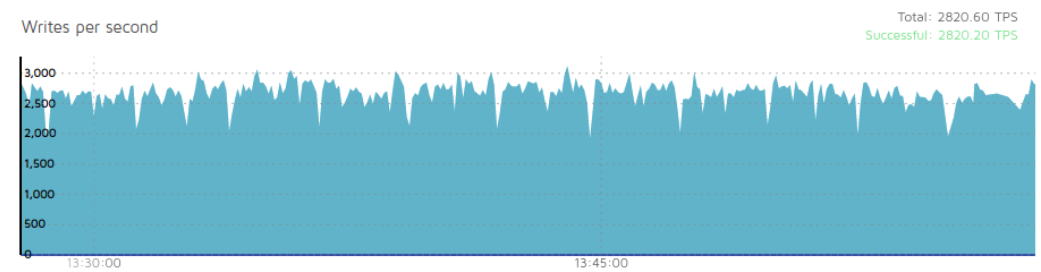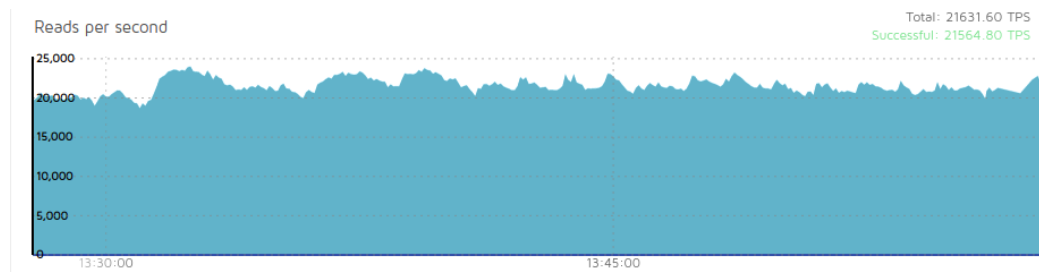- **DSP Bidding** (see next slides)

- ...and a few more

Bidder example

- Matching external to internal IDs as close to realtime as possible is crucial for the DSP bidder
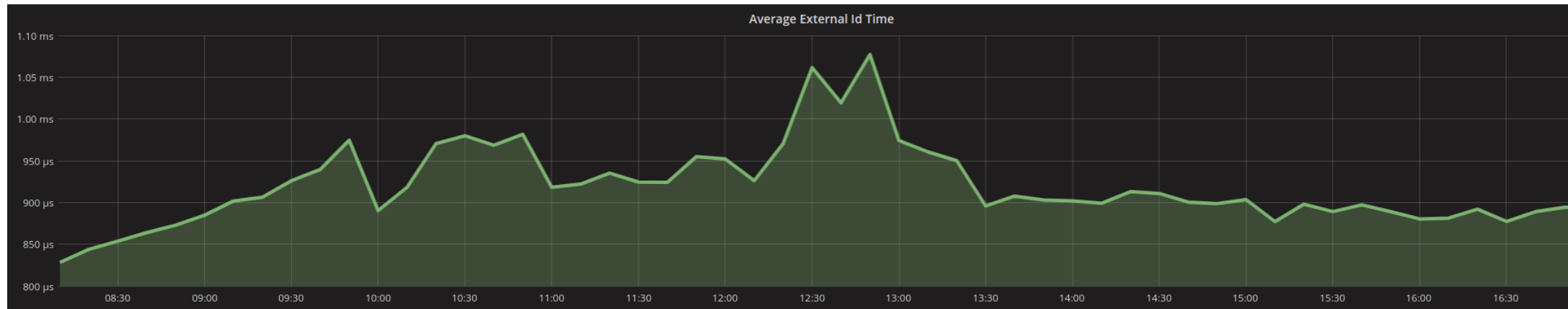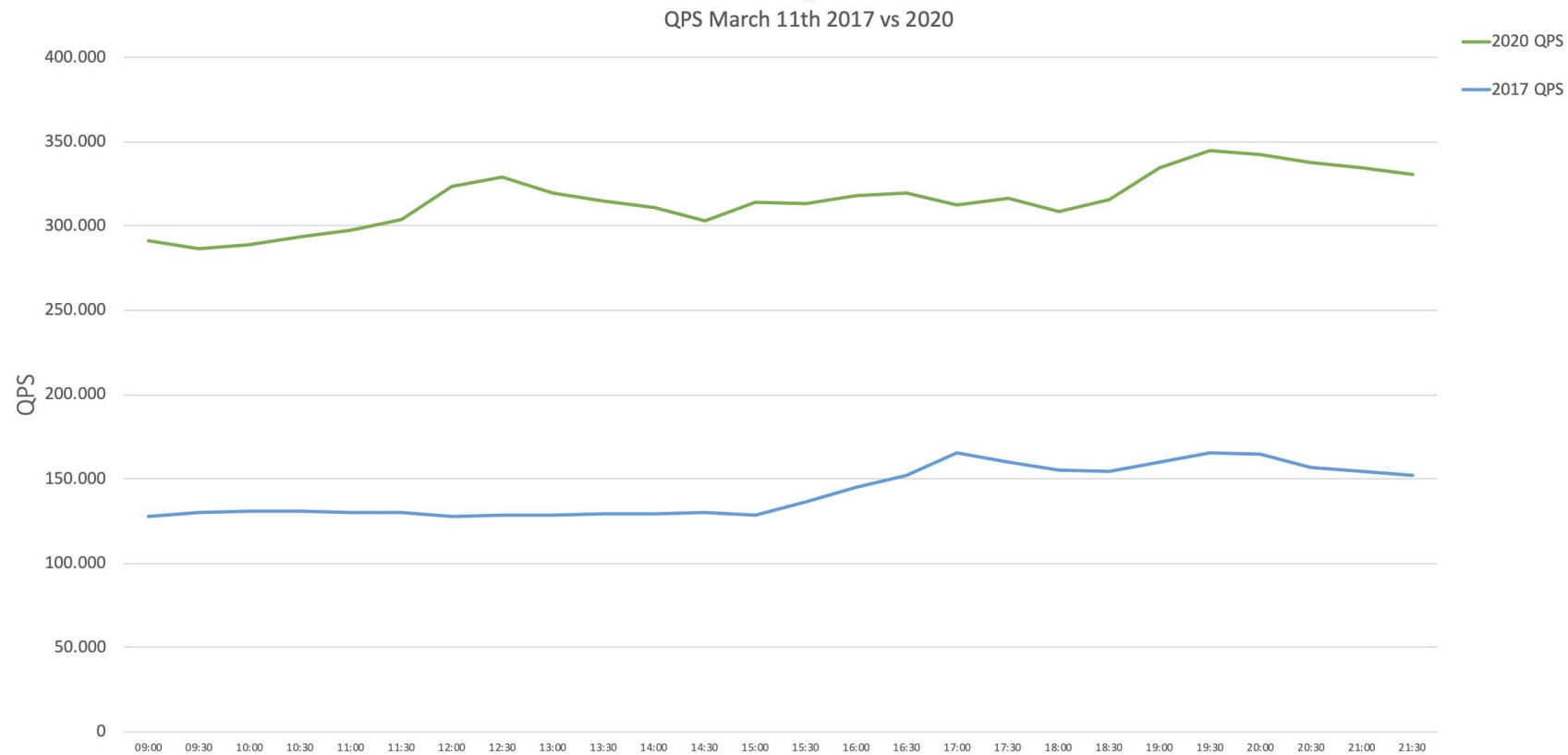
Bidder example

- Budget control in real time to perfectly match our users' targets:

Bidder example

- Especially in higher volumes and continuing growth:

QPS March 11th 2017 vs 2020

Bidder example

- And it survives Java pretty well:

```java
AerospikeClient aerospikeClient = ClientFactory.createClient("aerospikeServerAdress");
Policy policy = new Policy();
batchPolicy.timeout = 1000;
Key key = new Key("namespace", "set name", "key");
Record record = aerospikeClient.get(policy, key);
```

Bidder example

- A few more metrics about our latest deploy
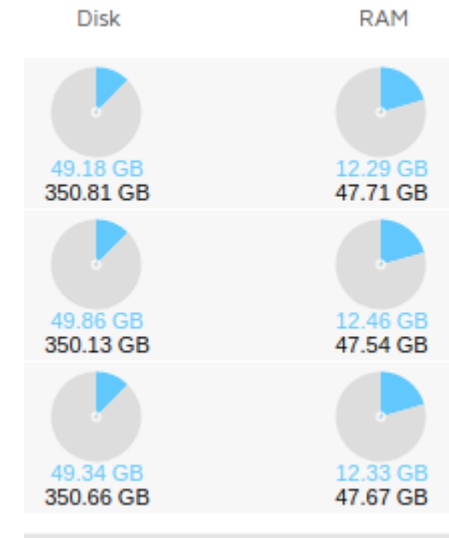
Embracing OSS

**Why we generally use OSS and why we have
chosen Aerospike Enterprise license**

Which features help us especially?

- **Unlimited nodes**
  - OSS solution would hinder our scaling needs
  - Spreading over unlimited amount of low-budget nodes made possible
- **Fast restart/start**
  - 40 min go live rather than hours meet our customer'S SLA requirements much better
- **Self recovery after loss of power**
- **Cross Datacenter Replication (XDR)** for high availability stores
- **Kafka and Spark connectors**

One more thing

- ## Aerospike is a great DB - no doubt

- ## We use Aerospike for one more topic:

  Change Management tool to switch culture and mindset from OSS and internal only to best of breed

Our Environment

- Low latency networks for us, gaming industry and more

- Enterprise hardware in use

- Automated

- Started on-prem; Ended up on-prem with own hybrid cloud solutions on top

Our network

- Redundant fibre-optic cabling

- Cross Connects between data centres

- Own backbone with 1.330 Gbits

- Peering:

  - Peering 350 Gbits

  - Transit Peering 600 Gbits

  - Private Interconnect Peering 380 Gbits

Hamburg
Düsseldorf
Amsterdam
Frankfurt
München

Overview

- 1999, founded as colocation provider in Düsseldorf

- 72 employes at two locations

- Operation of six data centers in Düsseldorf

- 24/7 on site support with our own technical specialists

- Operating of our own 1.330 Gbit backbone

- Hosting of dedicated and virtual server since 2003

- Individual hosting business specializing in media tech companies

- Provider of private cloud solutions since July, 2019

What makes the difference?

How do we interact with aerospike today, what is different to others

- No maintenance requirements (ease of use)

- No permanent tuning

- Regular calls/reminders with aerospike for configuration check

The change from OSS to Enterprise

- We got fully redundant

- We removed custom code
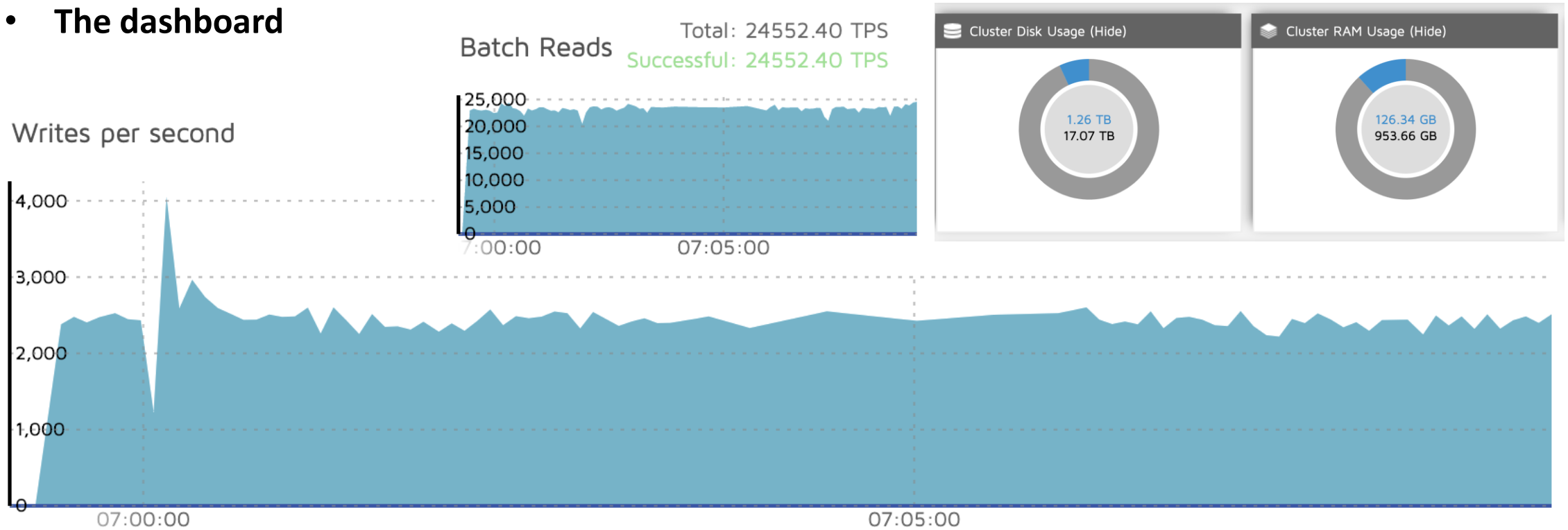
- We scaled up

- We changed scaling model

What use-cases do we see Aerospike in?

- Low latency high speed decision storing and making

- Robust redandant key-value storage

- No-ops scenarios

- Distributed locking

Which extra features do we love most and why?

- **The dashboard**

What's on our plan/roadmap for/with Aerospike?

- Aerospike plus Kubernetes for easy deploy and maintain scenarios

- Aerospike Kafka/Spark deep integration

Summing up

- Can be managed with a very small team
  - In operations
  - In development
- Fast, reliable, scalable
- It simply works as expected
- Good cooperation/partnership on tech and business level
- Supports us in speeding up on tech and organization

Thank You