GRAFANACON
AMS2018

GrafanaCloud

Anthony Woods
co-founder/cto

GrafanaCloud platform requirements

- Scalable for customers, but also scalable for our SRE!
- Fault tolerance and automated recovery
- Service discovery
- Horizontal Scaling
- Resource management
- Isolation between tenants

…. Kubernetes to the rescue; we’re all in!

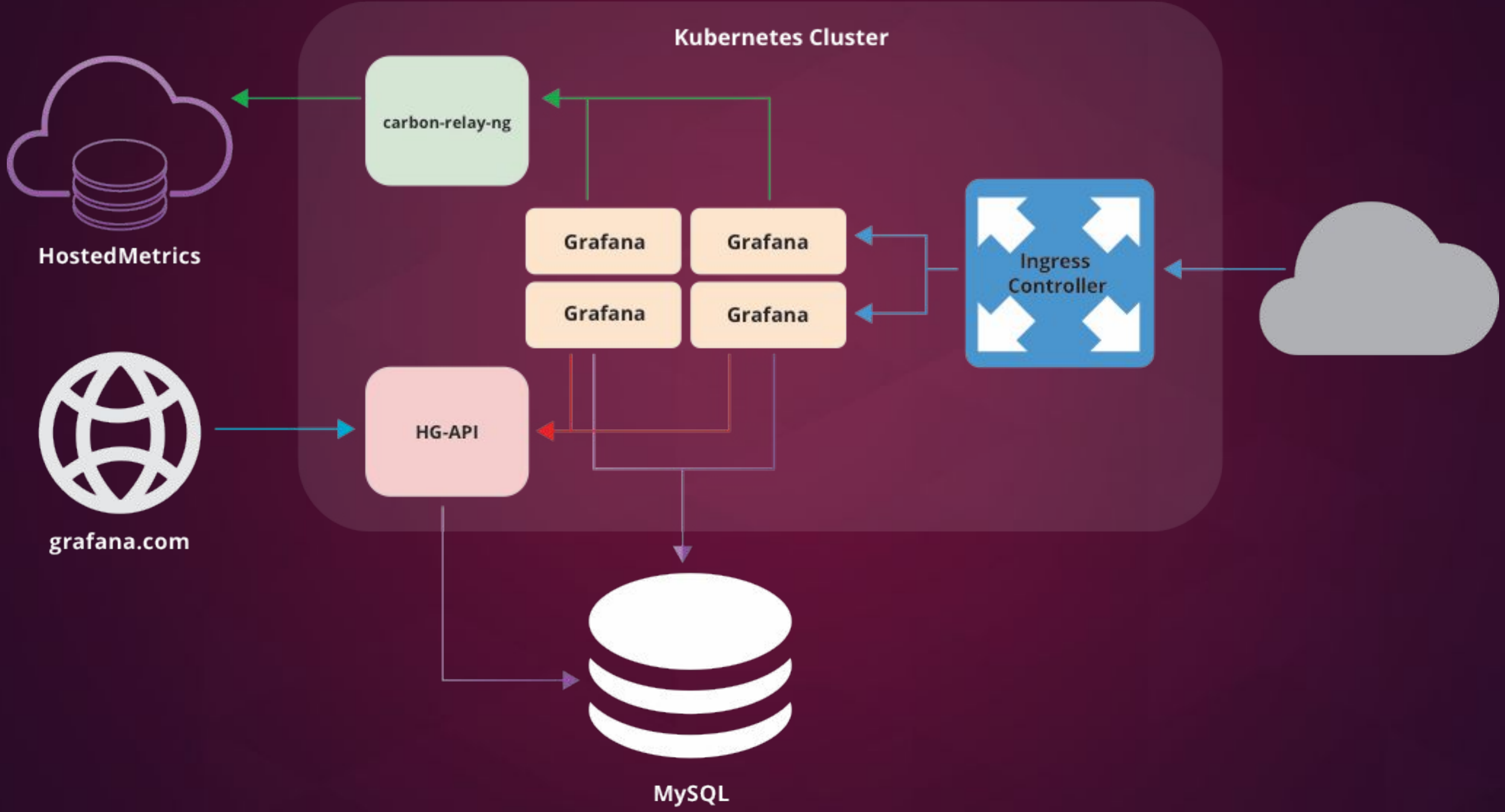
Kubernetes: our not so secret weapon

- A consistent platform for on-prem and SaaS deployments
  - Shippable SaaS
- Fully managed options reduce SRE burden
  - GKE (Google Kubernetes Engine)
- Also run vanilla K8s on bare metal
  - Packet.net
- Or wherever our customers want us to be
  - Eg. Azure AKS, AWS EKS, colo, for GrafanaCloud Private Deployments

Hosted Grafana

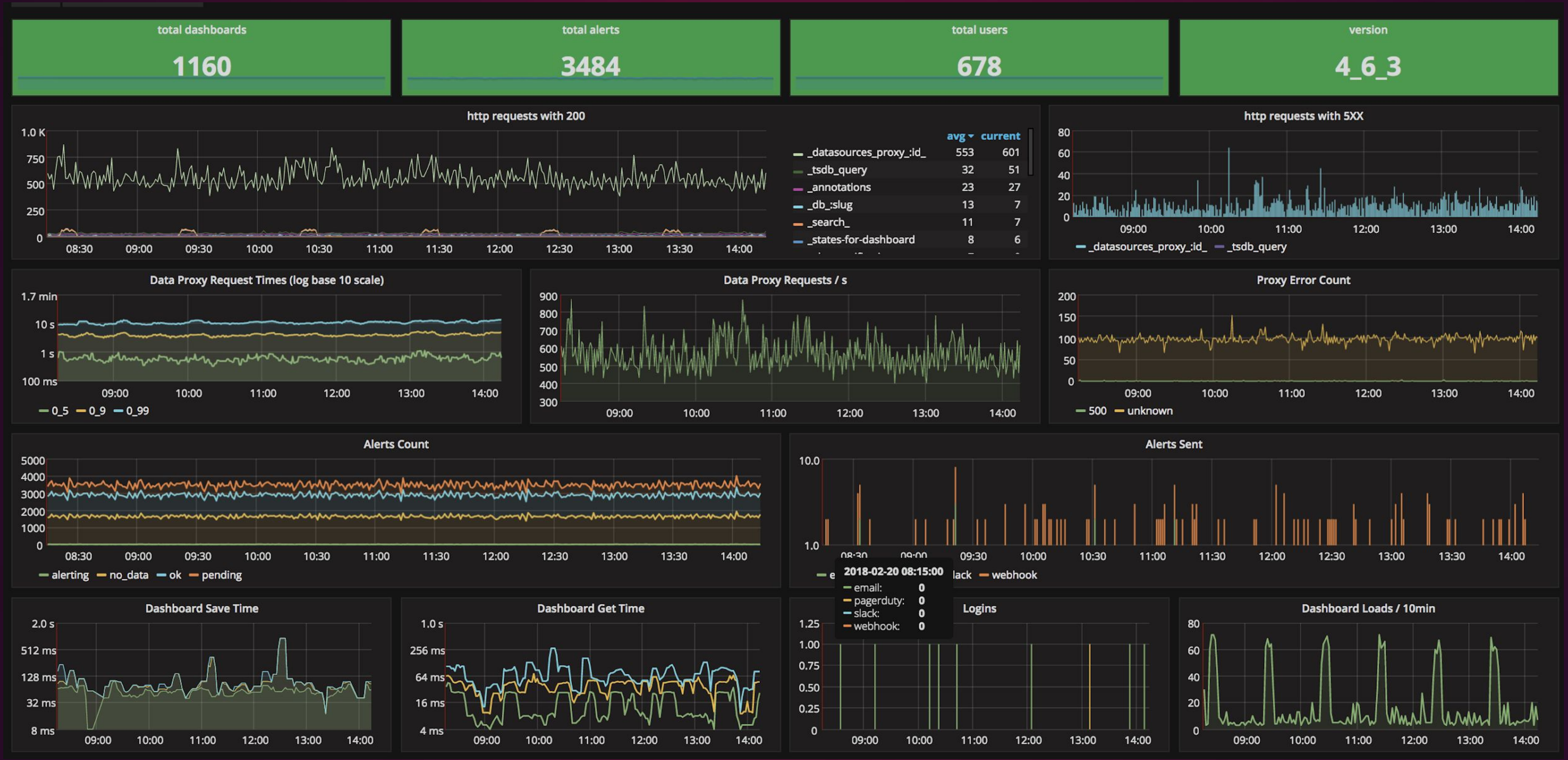
- A fully dedicated Grafana instance running the latest stable release
- One-Click installation of plugins from grafana.com
- Custom domain and authentication
- Anything config setting possible
- Who better to support it than the core Grafana team?

Hosted Grafana



GRAFANACON AMS2018

Hosted Grafana Instance Dashboard



total dashboards
1160

total alerts
3484

total users
678

version
4_6_3

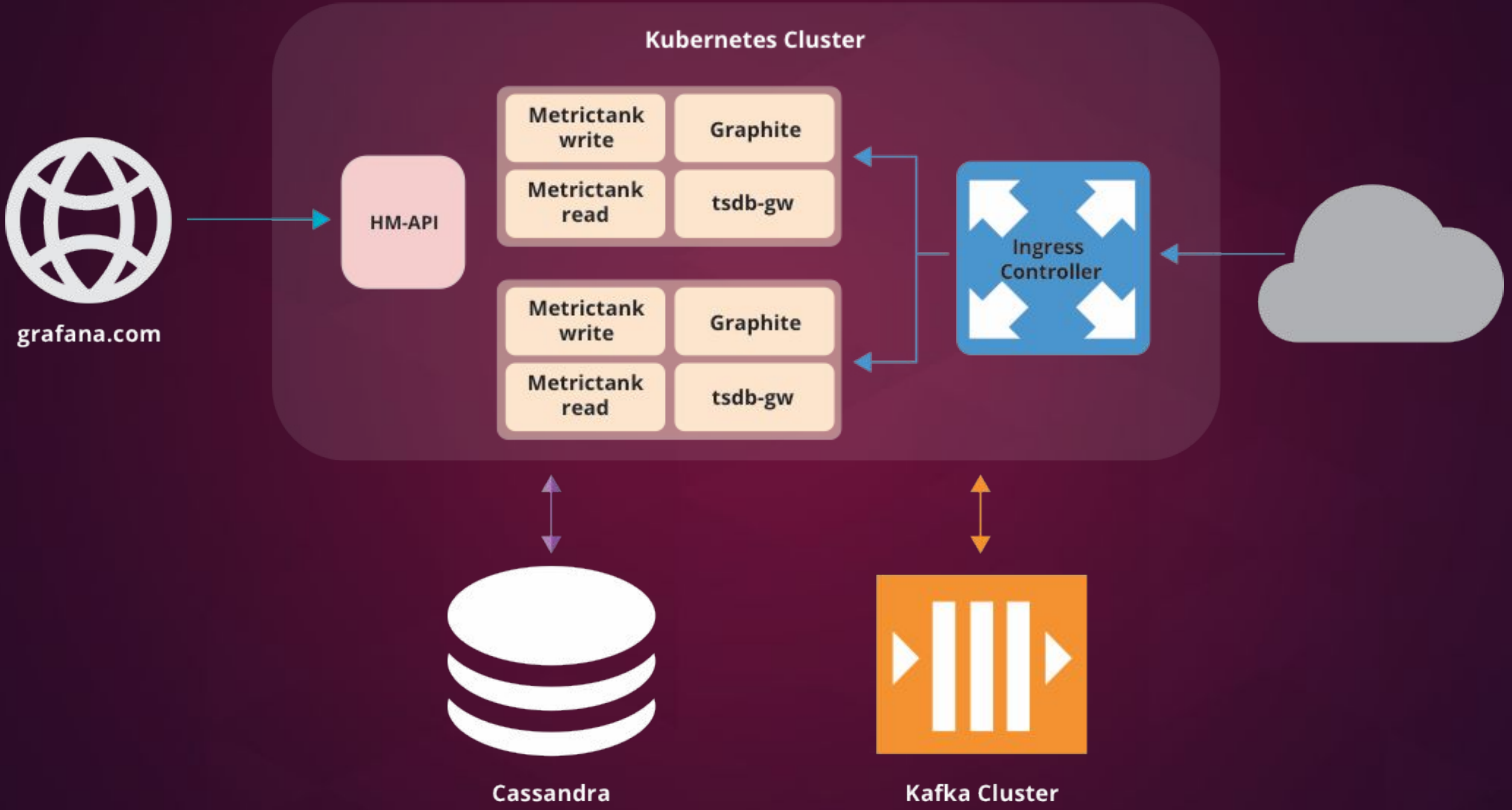
	avg ▾	current
_datasources_proxy_:id_	553	601
_tsdb_query	32	51
_annotations	23	27
_db_:slug	13	7
search	11	7
_states-for-dashboard	8	6

2018-02-20 08:15:00
email: 0
pagerduty: 0
slack: 0
webhook: 0

Hosted Metrics

- Unlimited* Scale
- Support for large metric volume (hundreds of millions of DPM)
- Fast query response times to support alerting
- Tunable for different workloads (eg. retention, cache, redundancy)
- Fault tolerant

Hosted Metrics



Hosted Metrics - core components

GrafanaLabs metrictank: https://github.com/grafana/metrictank

- Query engine compatible with Graphite and PromQL
  Keeps most data cached in memory for exceptionally fast query times
- Compresses and aggregates data then saves it to the backend store
  Inspired by Facebook Gorilla (similar algo as Prometheus and InfluxDB) < 2 bytes per point

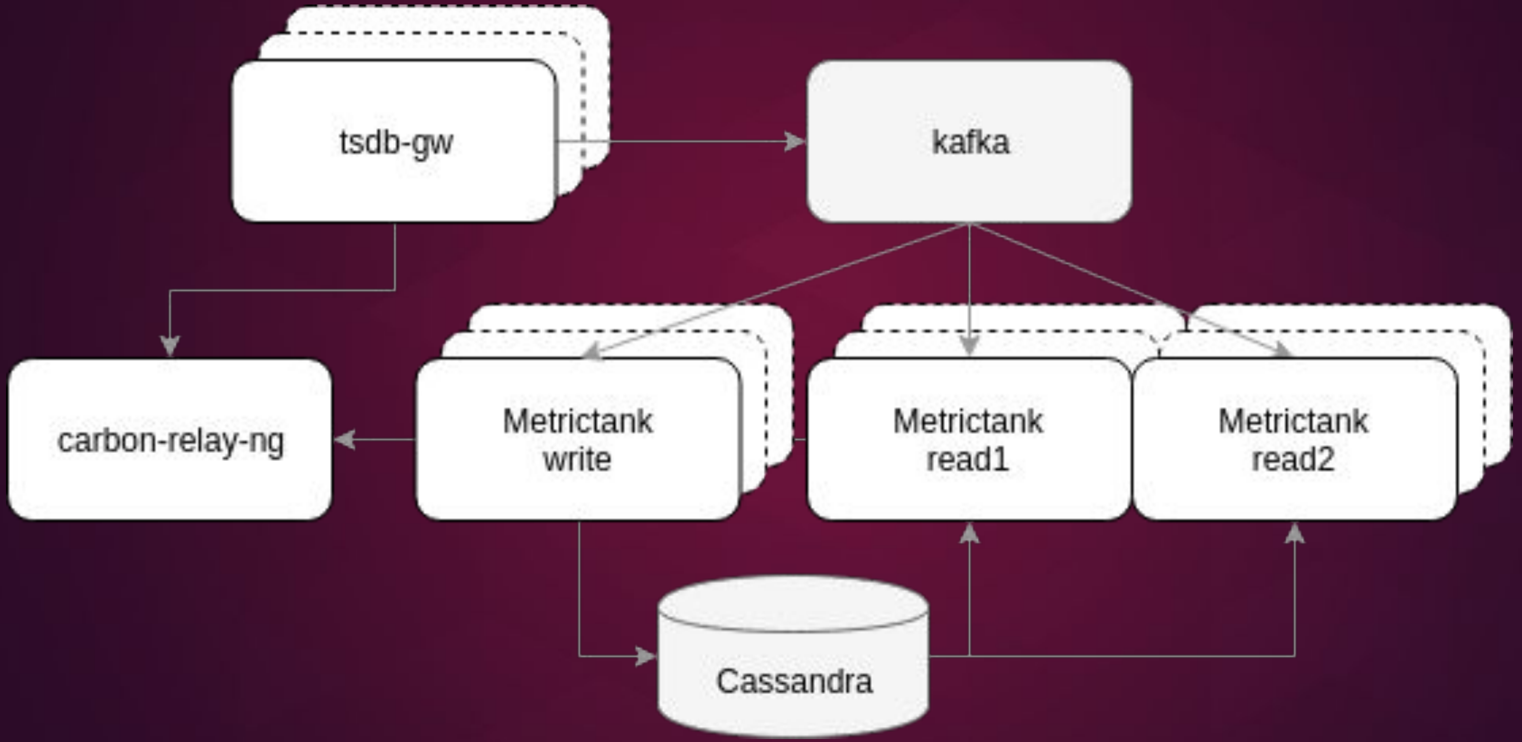
Apache Kafka: https://kafka.apache.org/

- Distributed Queue
  Provides resilience; we always need to accept data

Apache Cassandra: http://cassandra.apache.org/ or Google Bigtable

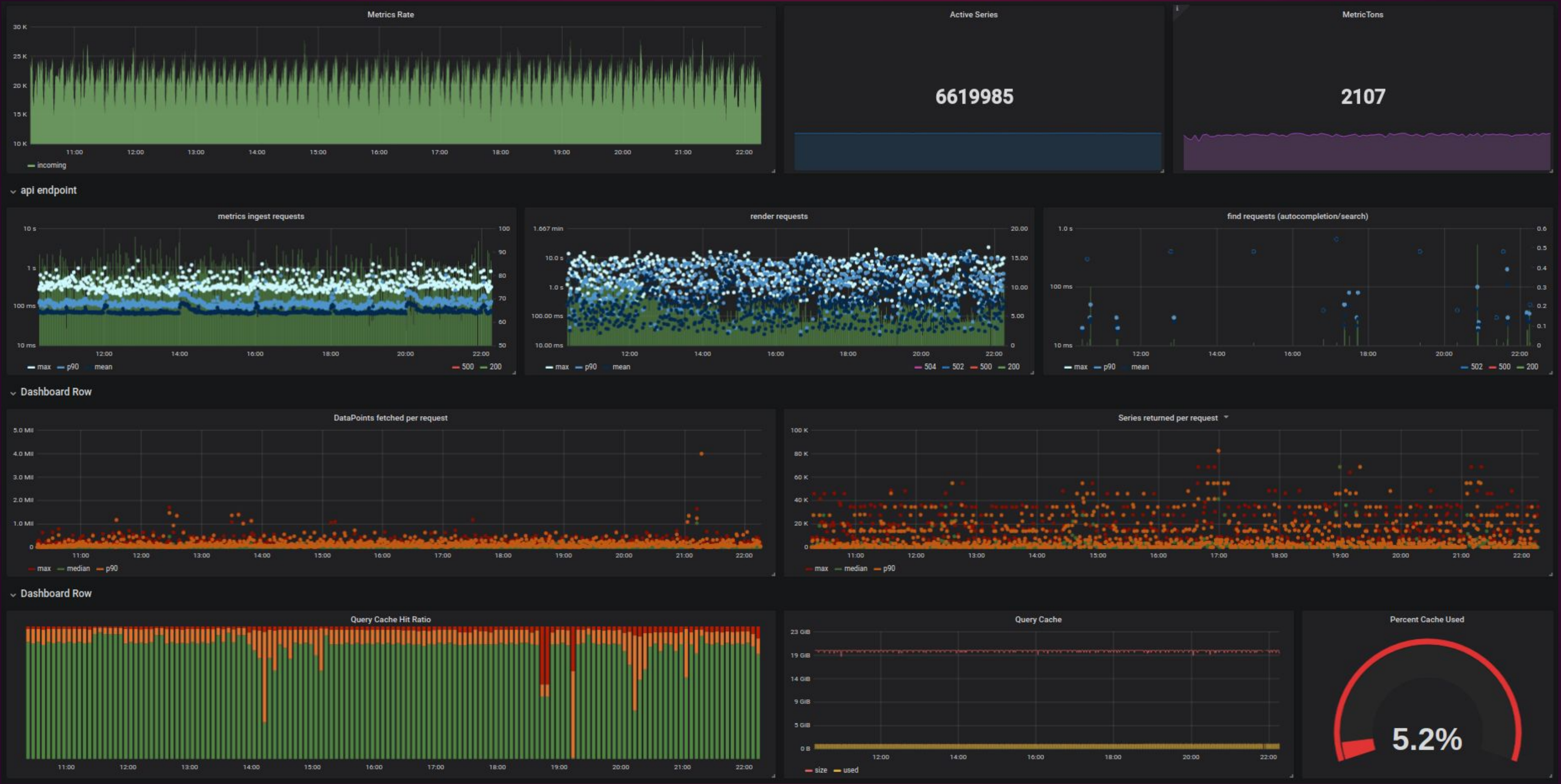
- Long term storage of metric data.
- Horizontally scalable

Hosted Metrics - Components



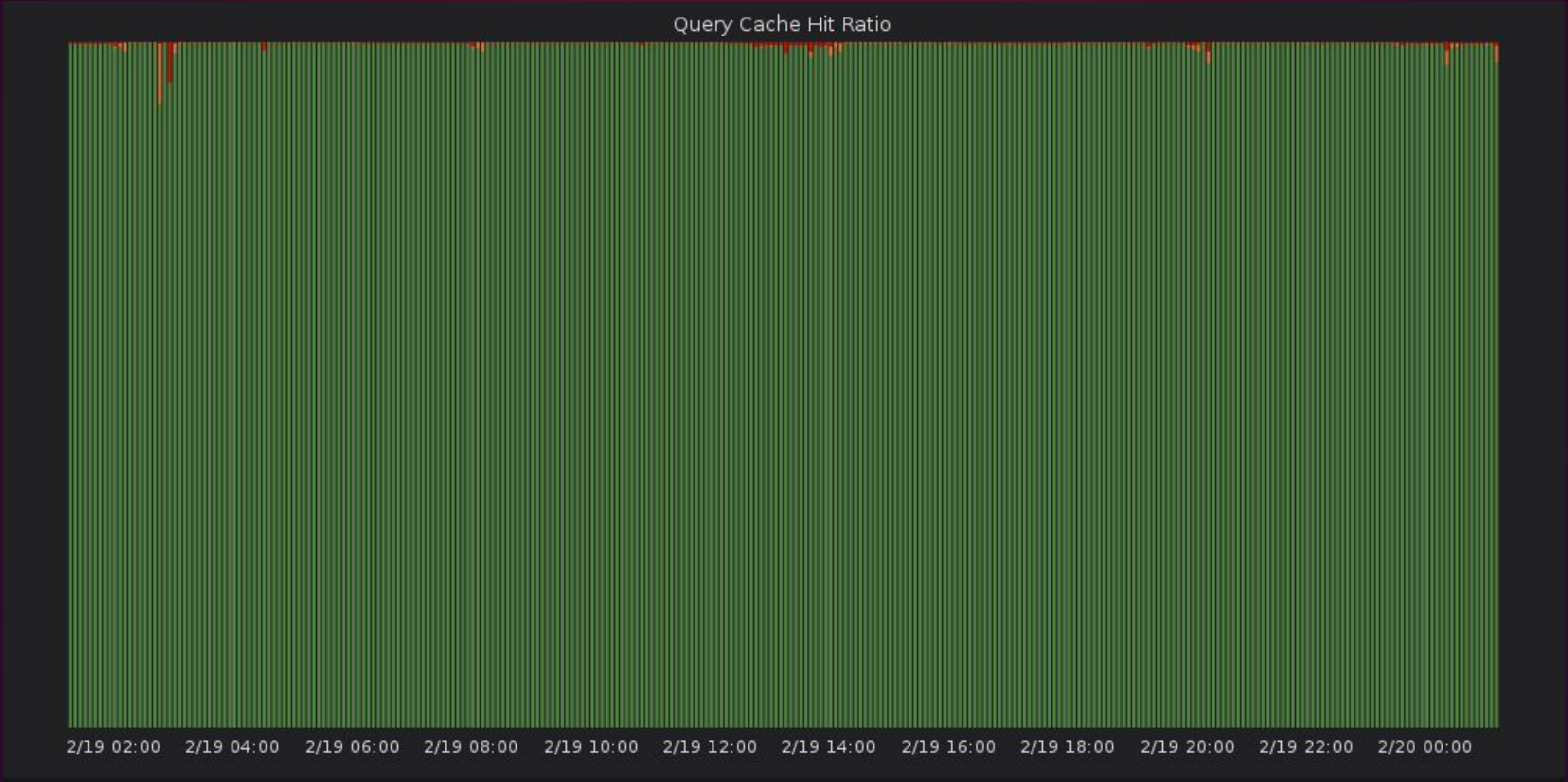
GRAFANACON AMS2018

Hosted Metrics Customer Dashboard



GRAFANACON
AMS2018

Cache Performance



Kubernetes

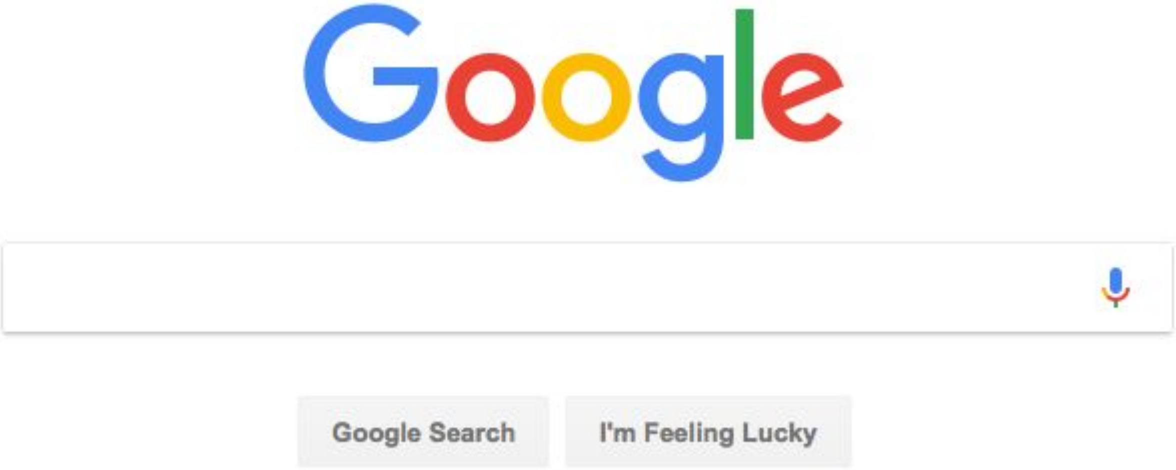
+

Bigtable

GRAFANACON
AMS2018

Google Cloud Bigtable

Misha Brukman
Product Manager


Google
Google Search
I'm Feeling Lucky

YouTube

How do we …

… run containerized workloads at scale?

Need: Deploy, scale and upgrade microservices quickly and efficiently

Solution: Borg, Kubernetes (open source)

Google Kubernetes Engine

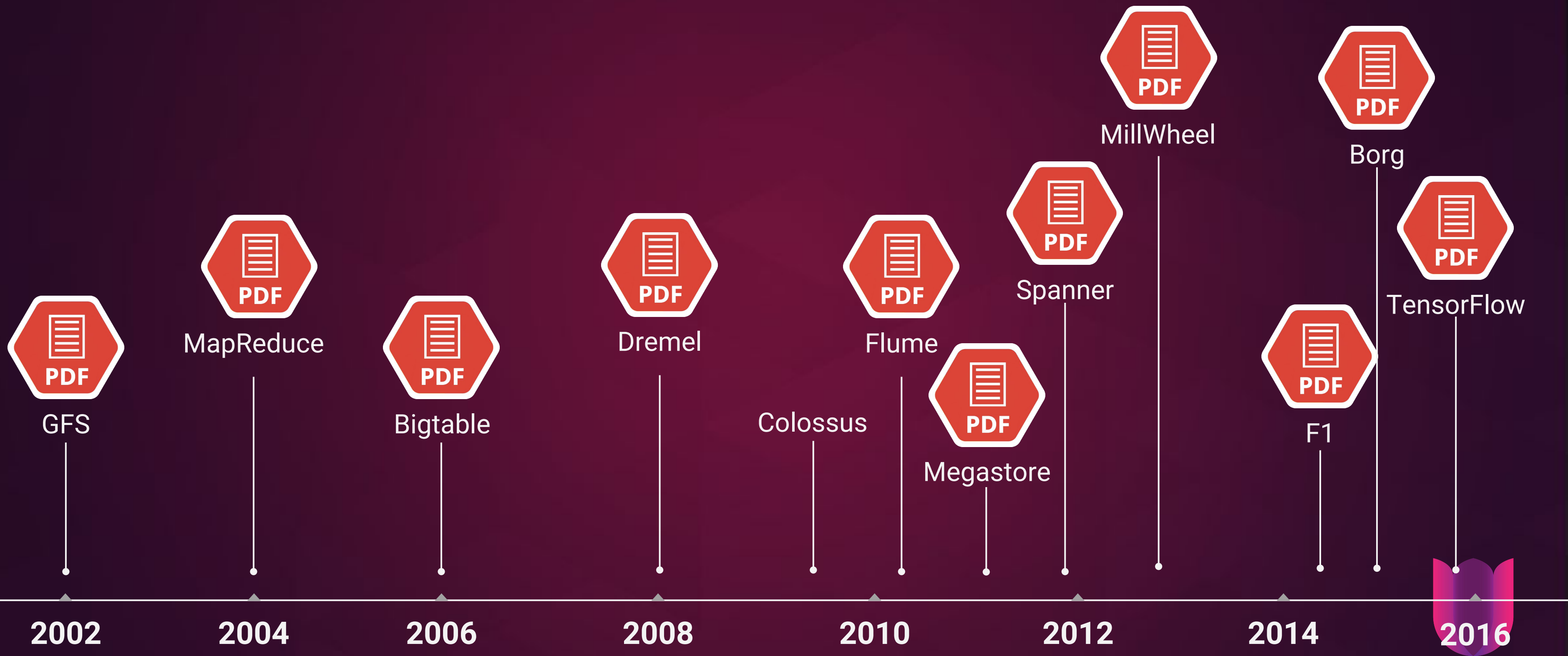
… build a petabyte-scale **analytics database**?

Need: Massive data index files took weeks to rebuild. We needed random read/write access

Solution: Bigtable

Google Cloud Bigtable

Technologies to support Google products



GFS

MapReduce

Bigtable

Dremel

Colossus

Flume

Megastore

Spanner

MillWheel

F1

Borg

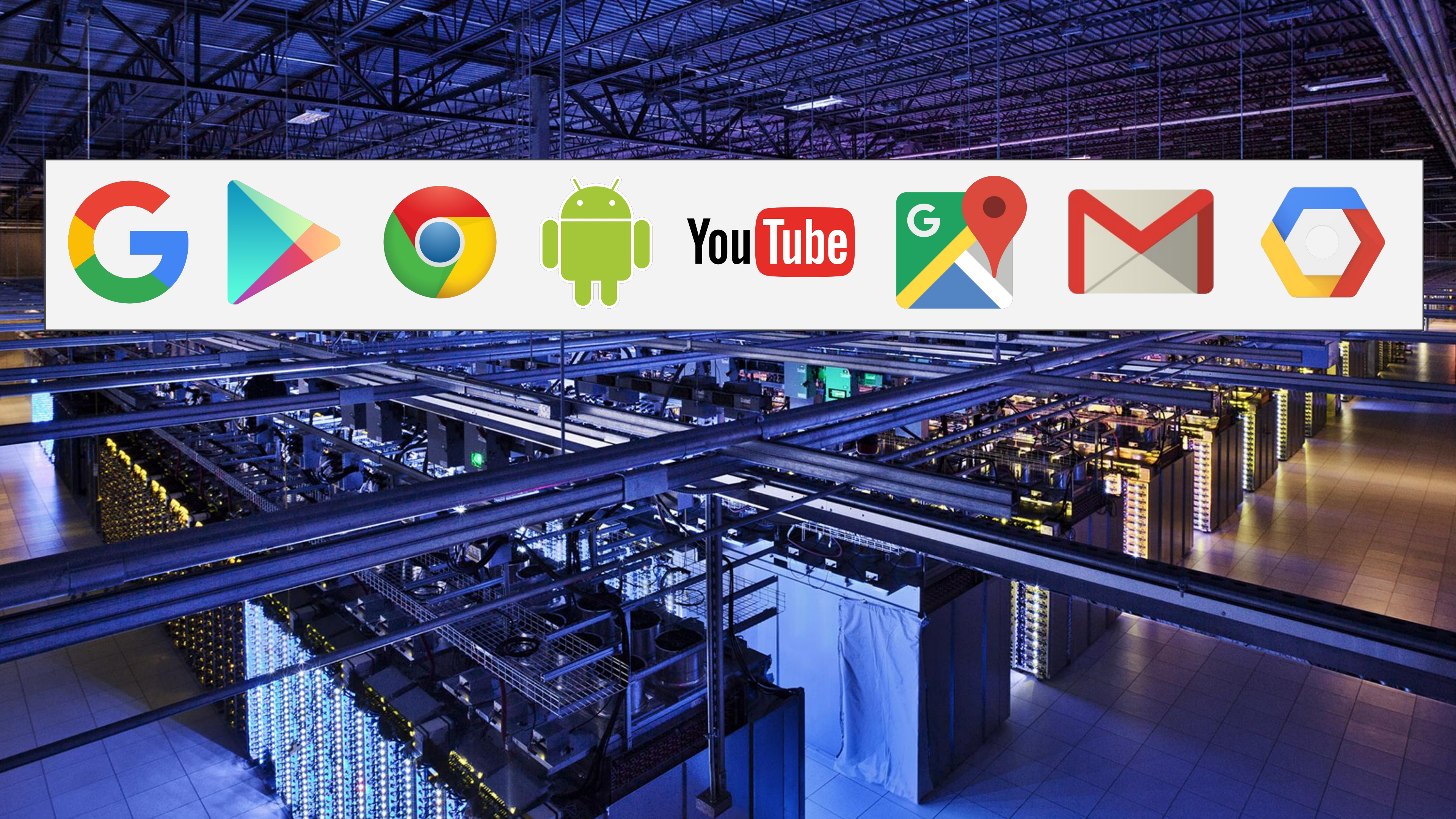
TensorFlow

2002 2004 2006 2008 2010 2012 2014 2016

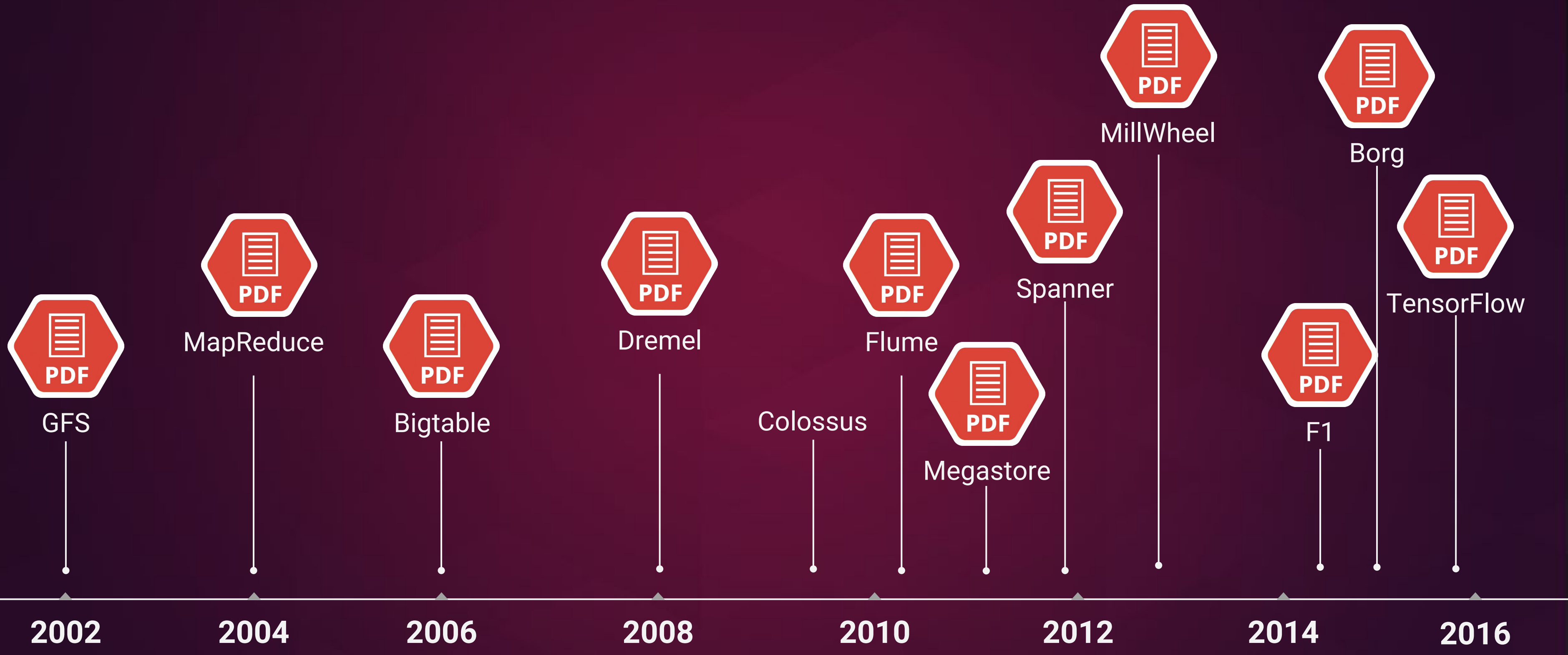
Imagine what you can build ...

... when scale is a solved problem

1 Billion users


YouTube

Technologies to support Google products



GFS

MapReduce

Bigtable

Dremel

Colossus

Flume

Megastore

Spanner

MillWheel

F1

Borg

TensorFlow

2002

2004

2006

2008

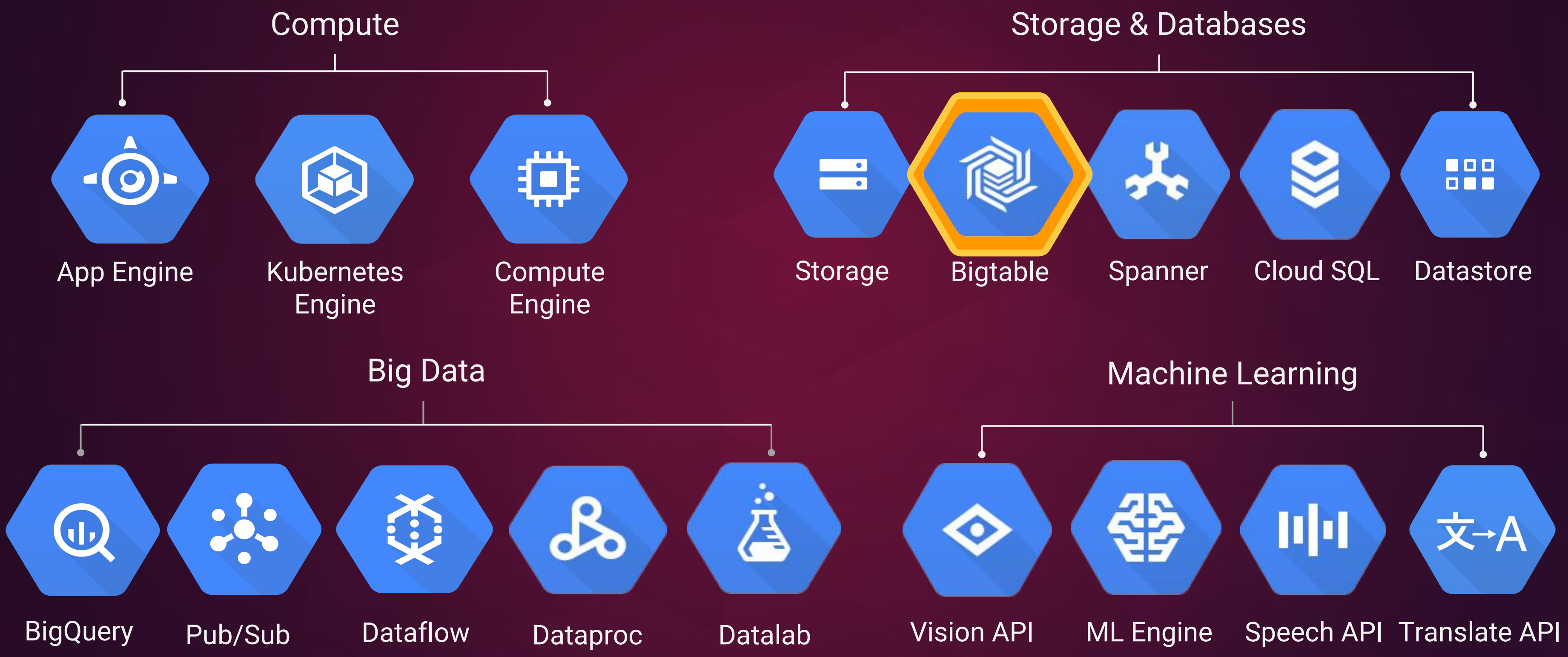
2010

2012

2014

2016

Now available on Google Cloud Platform



Compute

App Engine

Kubernetes Engine

Compute Engine

Storage & Databases

Storage

Bigtable

Spanner

Cloud SQL

Datastore

Big Data

BigQuery

Pub/Sub

Dataflow

Dataproc

Datalab

Machine Learning

Vision API

ML Engine

Speech API

Translate API

Google Cloud Bigtable

Google Cloud Bigtable

- Fully-managed NoSQL database
- **Built-in support for time series**
- Seamless scalability for throughput
- Learns and adjusts to access patterns

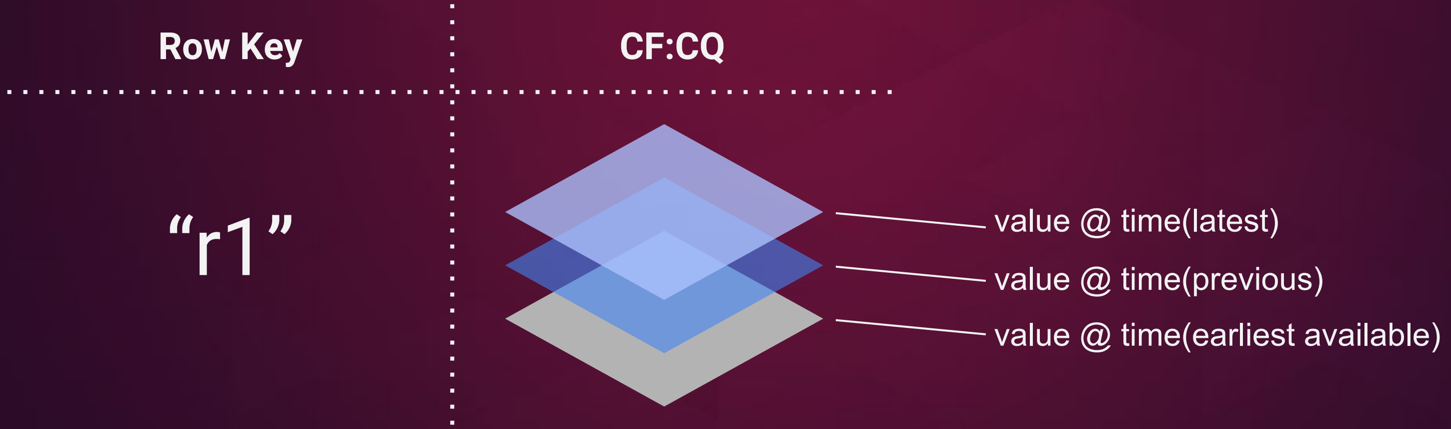
Bigtable data model

- NoSQL (no-join) distributed key-value store, designed to scale-out
- has only one index (the row-key)
- supports atomic single-row transactions
- unwritten cells in do not take up any space

	Column-Family-1		Column-Family-2	
Row Key	*Column-Qualifier-1*	*Column-Qualifier-2*	*Column-Qualifier-1*	*Column-Qualifier-2*
`r1`	`r1, cf1:cq1`	`r1, cf1:cq2`	`r1, cf2:cq1`	`r1, cf2:cq2`
`r2`	`r2, cf1:cq1`	`r2, cf1:cq2`	`r2, cf2:cq1`	`r2, cf2:cq2`

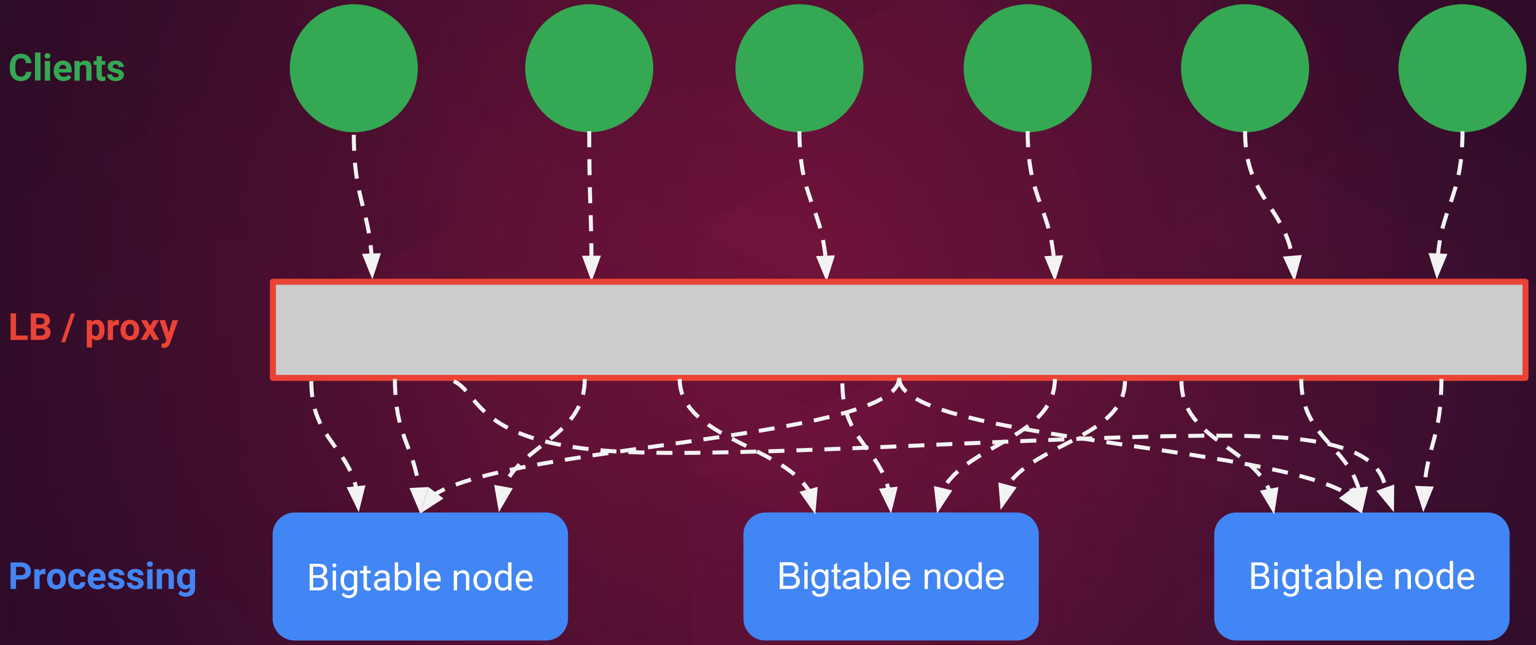
3D database structure enables time series

- every cell is ***versioned*** (default is timestamp on server)
- garbage collection retains latest version (configurable)
- expiration (optional) can be set at column-family level
- periodic compaction reclaims unused space from cells

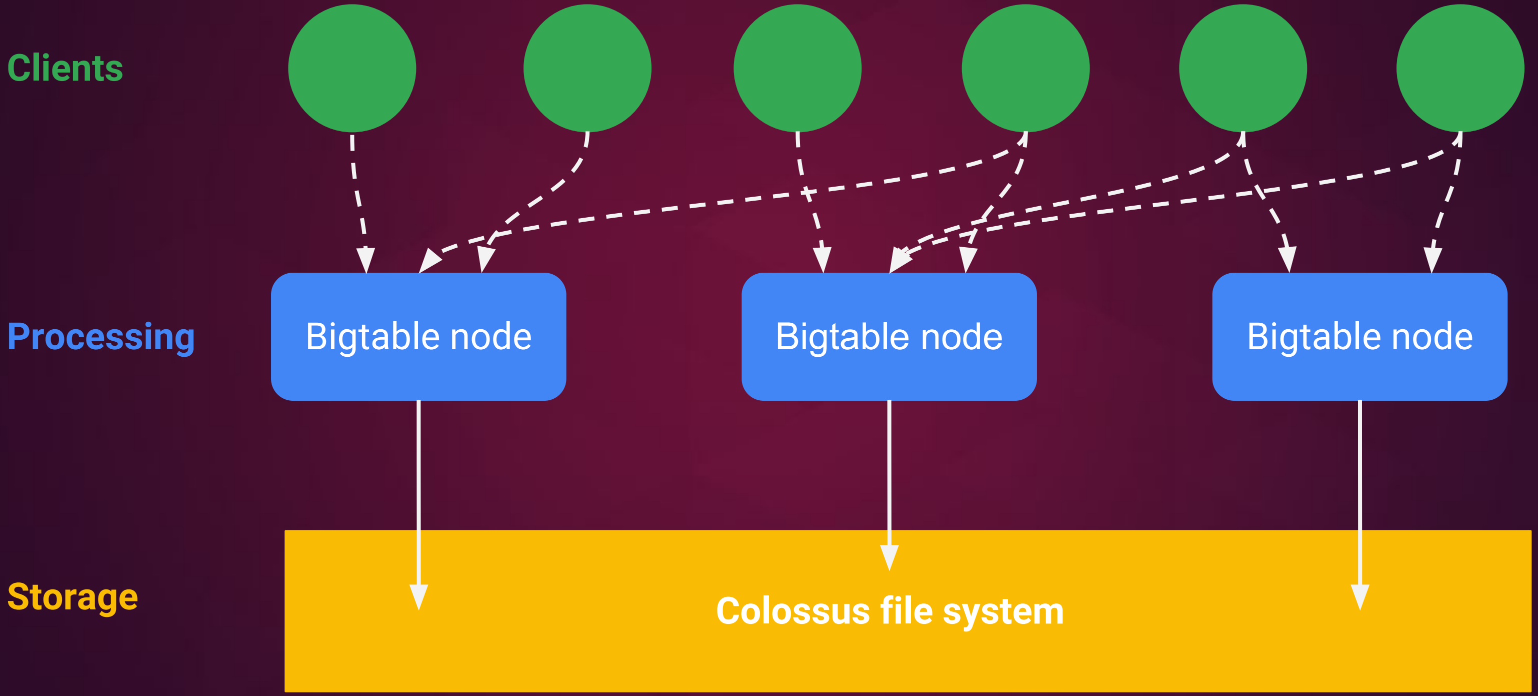


GRAFANACON AMS2018

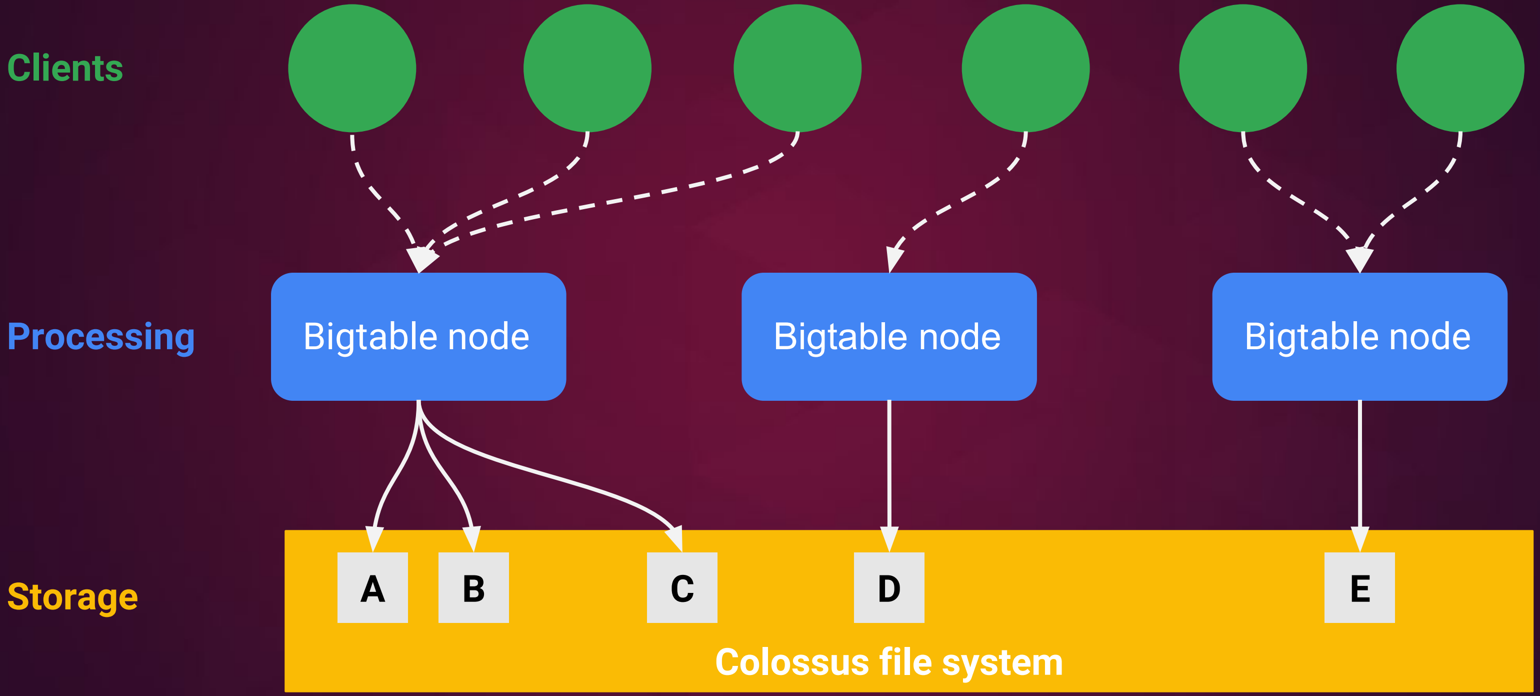
Bigtable high-level architecture



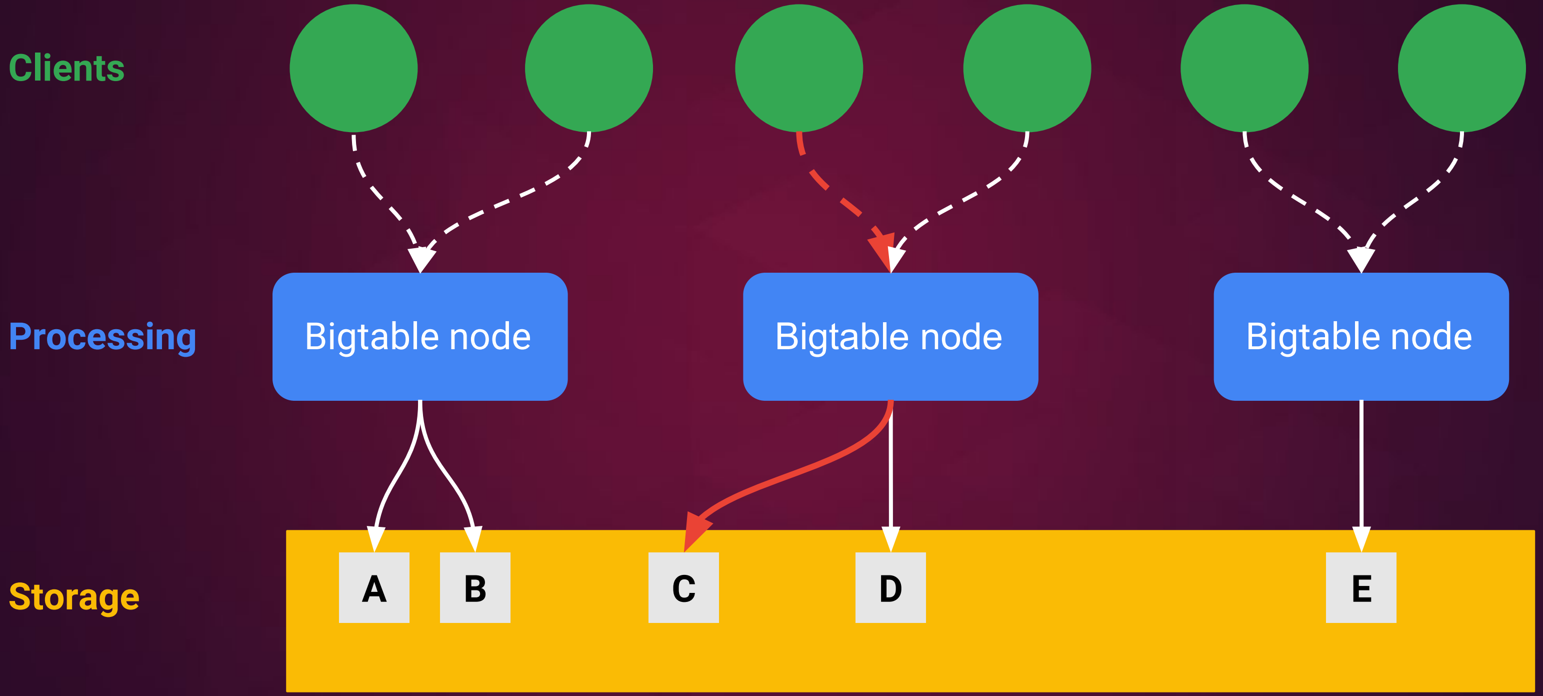
Bigtable separates processing from storage



Bigtable learns access patterns...

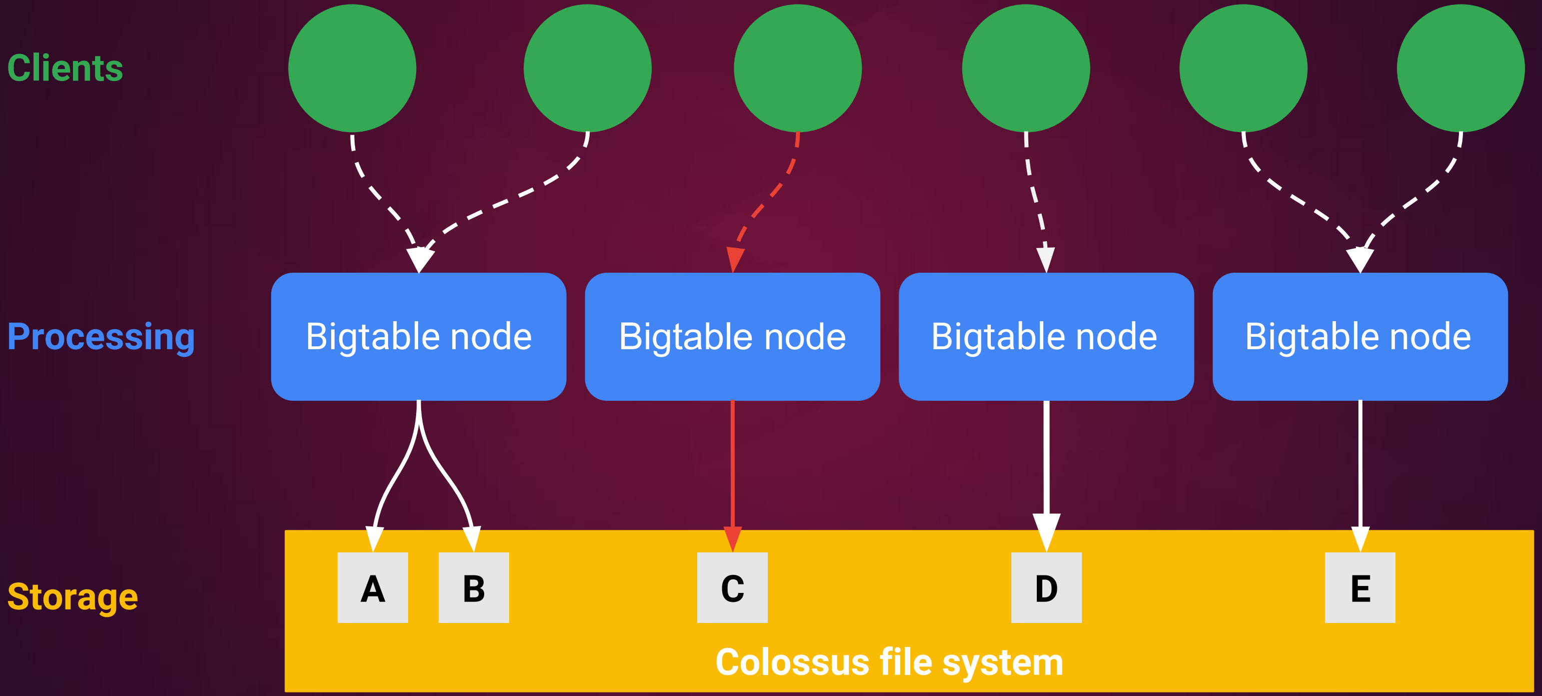


…and rebalances, without moving data



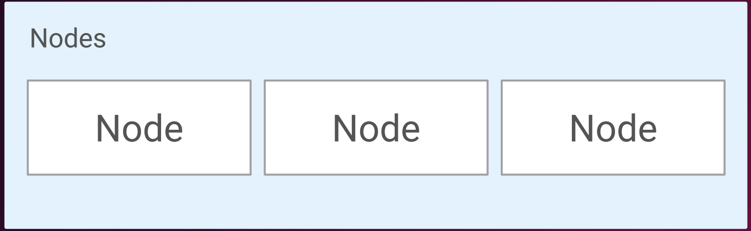
GRAFANACON AMS2018

Bigtable provides seamless resizing



GRAFANACON
AMS2018

Bigtable provides linear scalability in performance

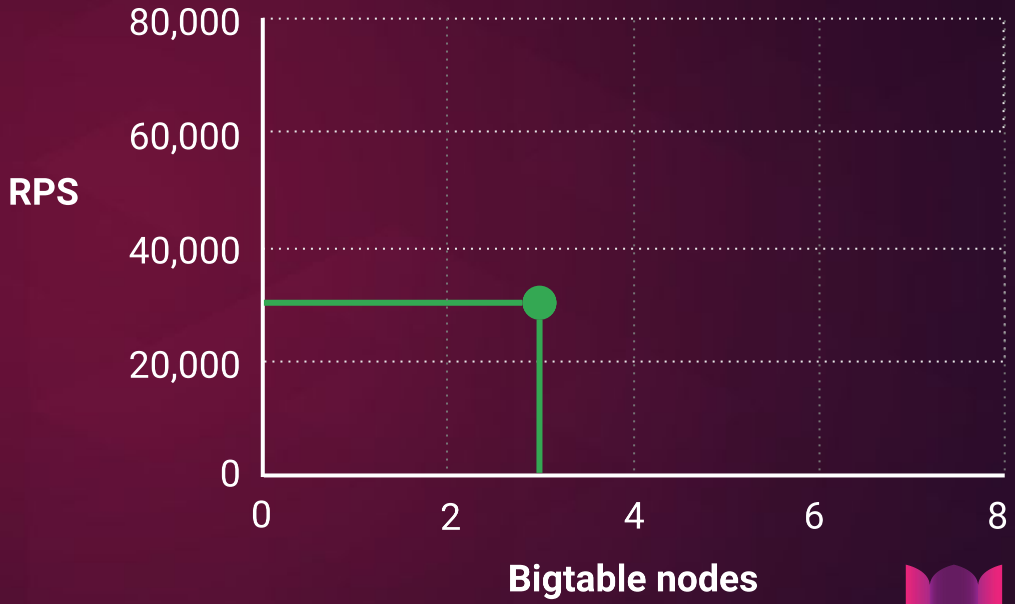


Nodes

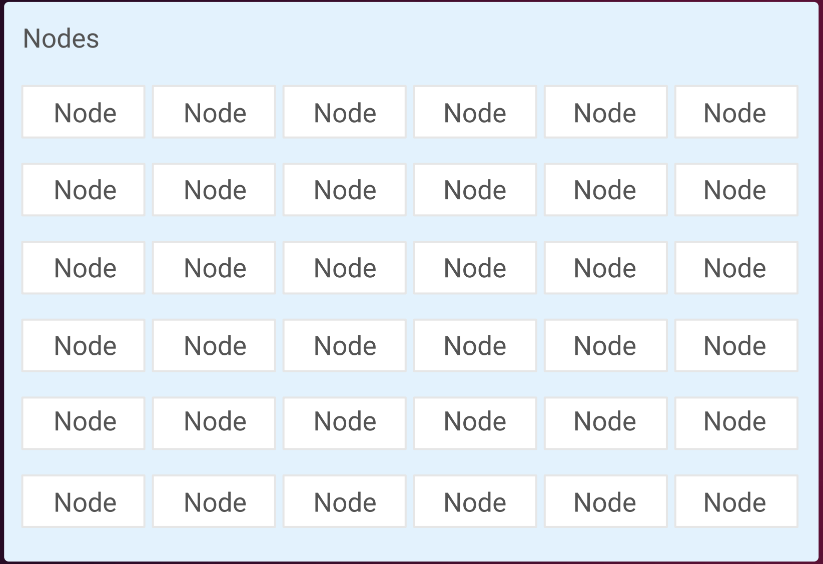
Node

Node

Node

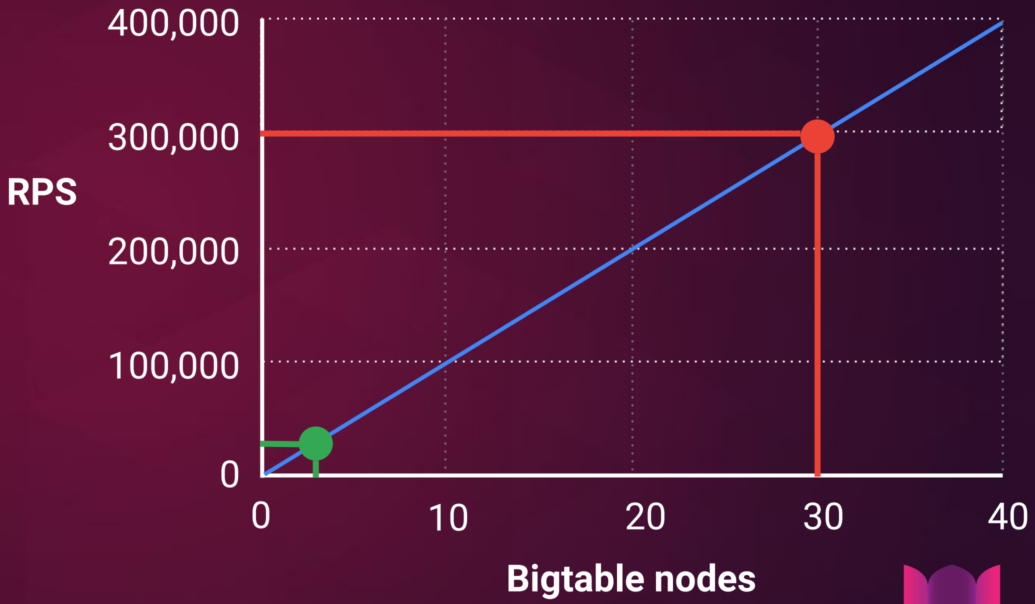


Bigtable provides linear scalability in performance

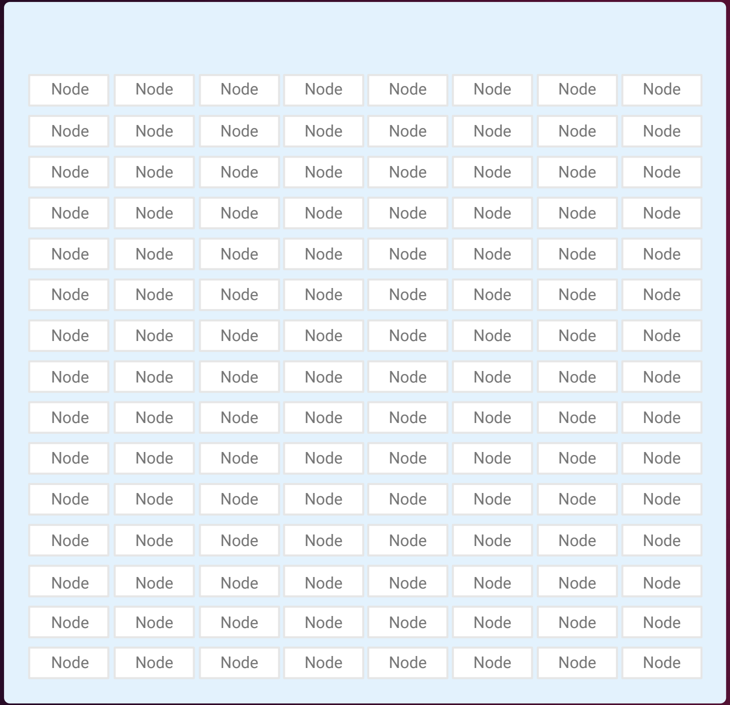


Nodes

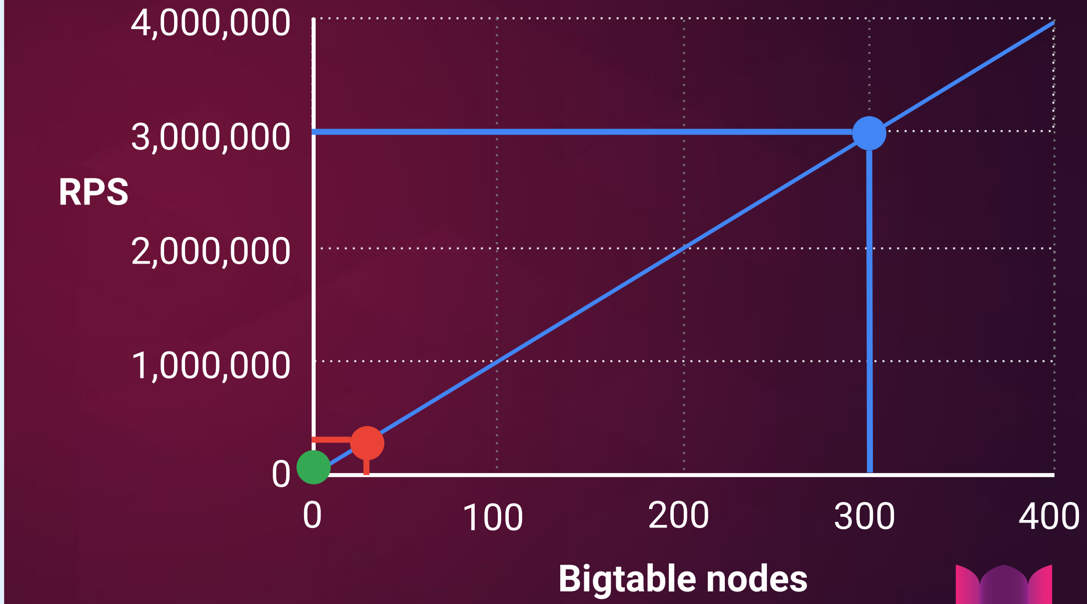
Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node



Bigtable provides linear scalability in performance



Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node
Node	Node	Node	Node	Node	Node	Node	Node



Great long tails

Single digit ms at the 99%

- Native scheduler protects serving path from compactions
- No garbage collection
- Very fast tablet reassignment

Google Cloud Bigtable

- Fully-managed NoSQL database
- **Built-in support for time series**
- Seamless scalability for throughput
- Learns and adjusts to access patterns

GRAFANACON
AMS2018

Metrictank

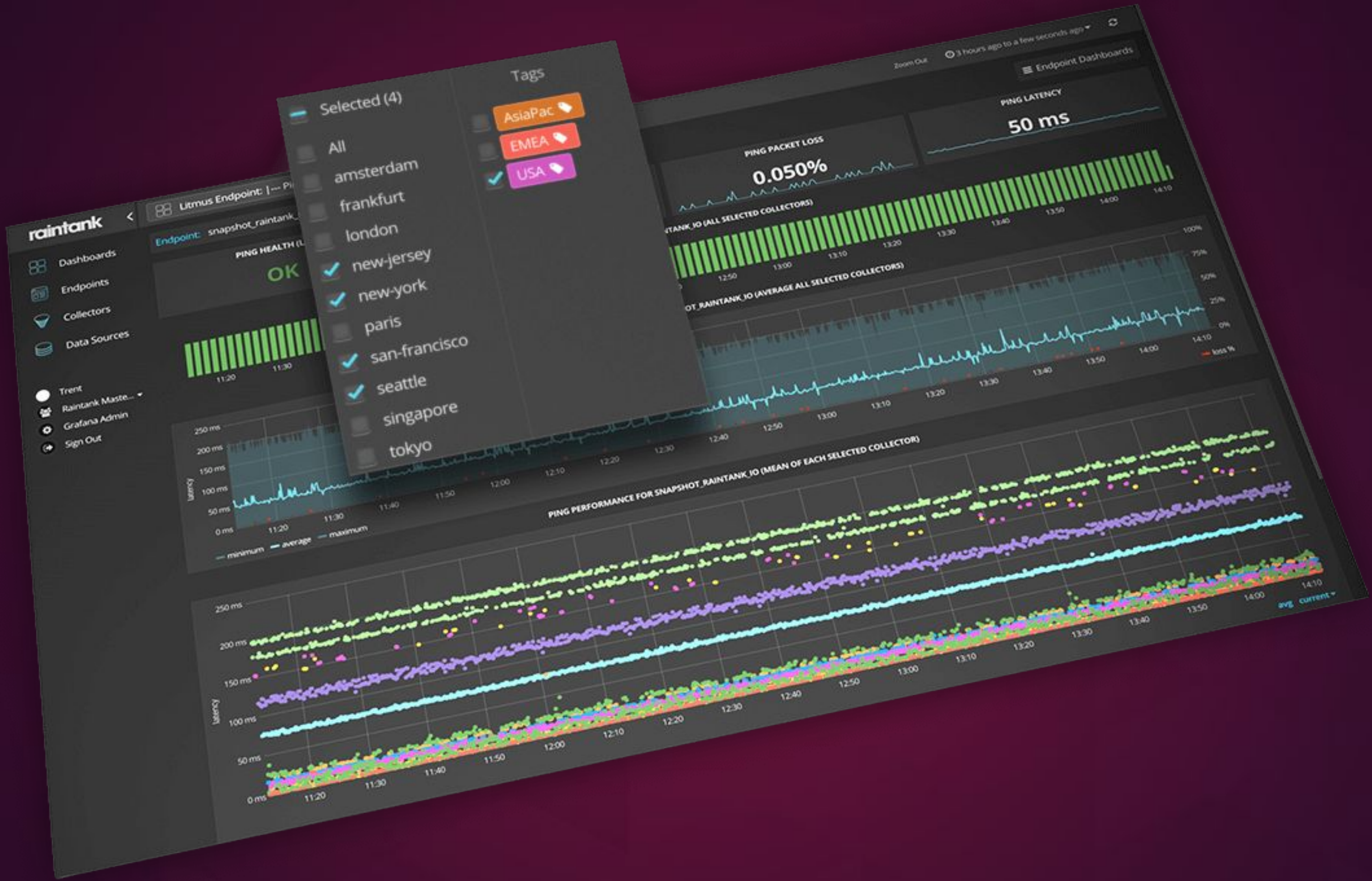
Dieter Plaetinck
Principal Engineer

Project
Not product

Data store
Not database

Genesis

(not the band)



Selected (4)

Tags

- □ All
- □ amsterdam
- □ frankfurt
- □ london
- ✓ new-jersey
- ✓ new-york
- □ paris
- ✓ san-francisco
- ✓ seattle
- □ singapore
- □ tokyo

- □ AsiaPac
- □ EMEA
- ✓ USA

Requirements for Worldping TSDB

- Large scale (millions of points per second)
- Long term storage, rollups
- Resource efficient (cpu, memory, disk)
- Multi-tenant
- Open source
- Operationally friendly
- Proven technology
- Compatible with Graphite (or pluggable into Graphite)

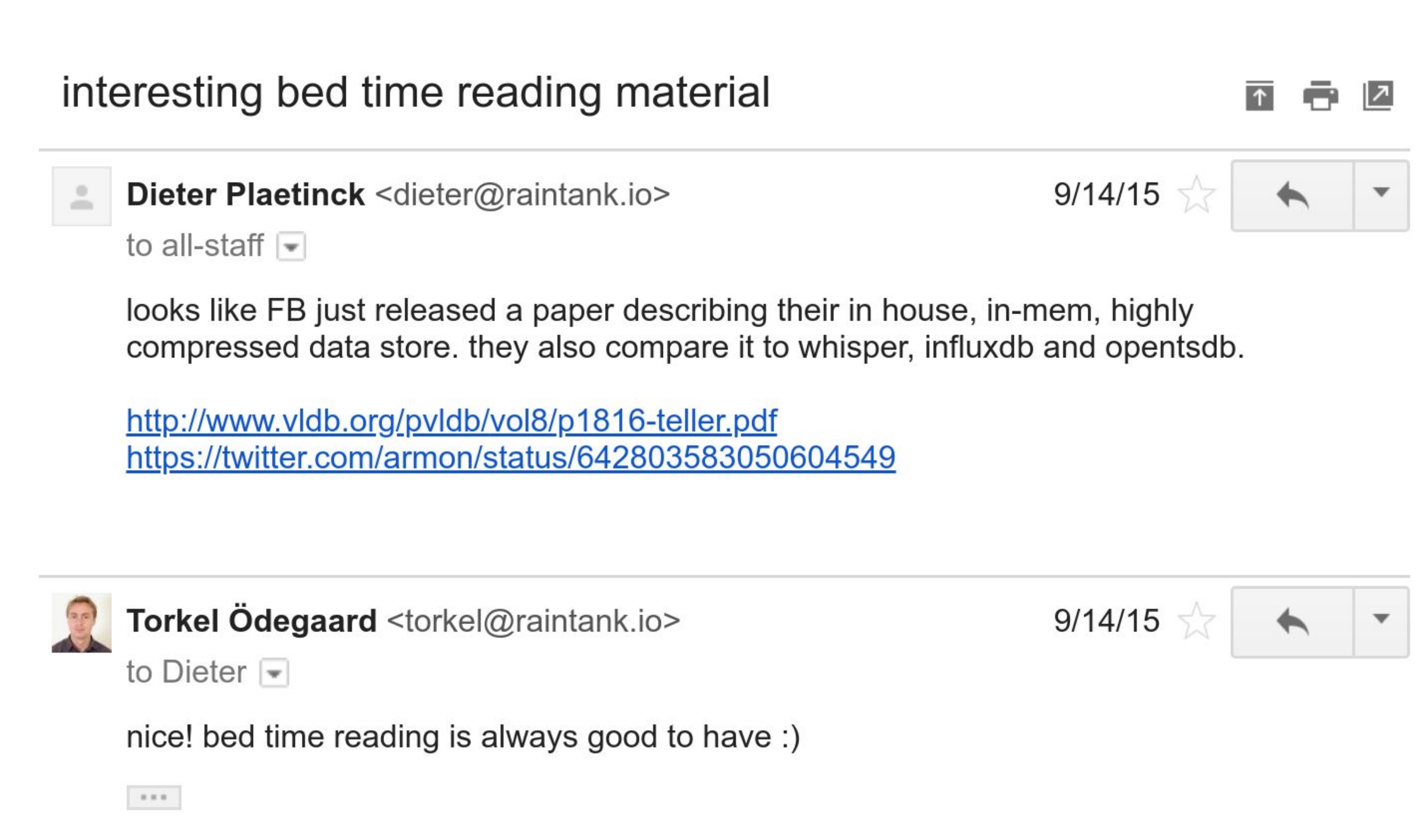
??

GRAFANACON
AMS2018

Didn’t want to write yet another TSDB

??

GRAFANACON
AMS2018



interesting bed time reading material

Dieter Plaetinck <dieter@raintank.io> 9/14/15

to all-staff

looks like FB just released a paper describing their in house, in-mem, highly compressed data store. they also compare it to whisper, influxdb and opentsdb.

http://www.vldb.org/pvldb/vol8/p1816-teller.pdf
https://twitter.com/armon/status/642803583050604549

Torkel Ödegaard <torkel@raintank.io> 9/14/15

to Dieter

nice! bed time reading is always good to have :)

- github.com/dgryski/go-tsz
- NSQ (later Kafka)
- Cassandra
- (Elasticsearch for index)

Didn’t want to write yet another TSDB

Timeline

- Sept 23, 2015 : First prototyping
- Dec 2015: Worldping production

Do we really want our own TSDB?

- 2016: Ad-hoc hosted metrics alpha’s

Do we really want our own TSDB?

- Early 2017: Grafanacloud v1

Looks like it

- Early 2018: Grafanacloud v2

OK then. Can we add prometheus?

metrictank

- service that reads from queue, compresses data to chunks. saves to DB
- Saves rollups
- Satisfies queries from memory and DB
- Input: Kafka (graphite, Prometheus, OpenTSDB, …)
- Input: direct Carbon, prometheus
- Whisper import
- Graphite function api (mix built-in and graphite-web)
- PromQL
- Can be deployed as eventually consistent cluster

Integrating
Not replacing

Input options

- Kafka (carbon-relay-ng graphite, Prometheus, OpenTSDB, …)
- Plain carbon, prometheus (!!)
- Whisper importer

Storage options

- Cassandra
- Bigtable
- (CosmosDB?)

Output options

- Graphite api
- Prometheus api
- ...

Data

- Chunk ringbuffer in memory
- LRU chunk cache in memory
- Storage plugin for persistence (Cassandra, …)
- Can reach ~100% memory hit rate

Metadata (index)

- Plugin (Cassandra, …) for persistence
- Full in-memory copy
- Built-in expression handling, searching, tag index, autocomplete, etc

Improve on Graphite

https://grafana.com/blog/2016/03/03/25-graphite-grafana-and-statsd-gotchas/

- Seamless changing of native data resolution
- Better support for churn (shortlived data)
- Multiple rollup functions, choice at query time (WIP)
- Automatic interval detection (WIP)

Worse than Graphite

- Data must be mostly-ordered. No rewrite support
- No xFilesFactor yet

Clustering

HA (replication)

&

horizontal scaling (partitioning/sharding)

Clustering: HA (replication)

- Simply run # replicas desired (via orchestrator)
- Primary role (via config/orchestrator or API, not automatic)
- kafka/NSQ for tracking save state
- Kafka data backfill reduces time-to-ready

Clustering: horizontal scaling (partitioning)

- Shard assignment tied to input (via config/orchestrator)
- Shard deterministically derived from metric name & metadata
- Index per node only for shards it “owns”
- Gossip for membership
- Queries can hit any instance, scatter+merge
- Kafka-lag based ready-state, priority, and min-available-shards

Clustering limitation 1

primary status per instance, not shard

node	A	B	C	D
shards	0 1	0 1	2 3	2 3

Clustering limitation 1

primary status per shard

node	A	B	C	D
shards	**0** 1	0 **2**	**1** 3	2 **3**

Clustering limitation 2

- Rigid sharding scheme. Can’t add/remove shards at will.
- => (live) cluster migration

Clustering trade-offs

- https://martin.kleppmann.com/2015/05/11/please-stop-calling-databases-cp-or-ap.html
- Kafka : very tuneable. Ours tuned for consistency -> buffering client side (rare)
- Cassandra : Eventually consistent. Tunable consistency latency trade-off
- eventually consistent. Everything streams in. Even when talking to MT directly
- Don’t need transactions for monitoring data
- MT read instances depend on writers saving to Cassandra

Use whatever makes sense for you

That’s why Grafana supports graphite, influxDB, prometheus, cloudwatch, ….

That’s why metrictank supports Cassandra, Bigtable, ….

Tools

mt-aggs-explain

mt-explain

mt-index-cat

mt-index-migrate

mt-kafka-mdm-sniff

mt-kafka-mdm-sniff-out-of-order

mt-replicator-via-tsdb

mt-schemas-explain

mt-split-metrics-by-ttl

mt-store-cat

mt-update-ttl

mt-view-boundaries

mt-whisper-importer-reader

mt-whisper-importer-writer

Tools

```
mt-index-cat -prefix statsd.prod -tags none -max-age 12h cass 'GET
http://metrictank/render?target=lowestCurrent(sumSeries({{.Name |
pattern}}),2)&from=-30min\nAuthorization: Bearer foo\n\n' \

| ./vegeta attack -rate 5 | ./vegeta report
```

Fun under the hood stuff

- Golang issue #14812 GC bug
- Metrictank PR #136 Buffer reuse, custom json encoder, etc
- Golang contexts
- Jaeger tracing (opentracing)
- Automated chaos testing with docker-compose and pumba/tc
- profiletrigger

Metrictank use cases

Large scale graphite installations

Long term storage prometheus

SaaS without vendor lock-in

Favor known database

Conclusion

- Try it out, but beware
- Or try GrafanaCloud (SaaS or Private Deployment)

Integrate with ecosystem

Not divide and conquer

GRAFANACON
AMS2018

Azure Cosmos DB

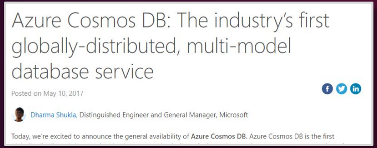
Anko Duizer
Sr. Technical Director for Global Cloud ISVS, Microsoft

Azure + OSS + Grafana: Years in Review

Dec 2016 · May 2017 · July 2017 · October 2017 · November 2017 · 2018

Dec 16

Joined Linux Foundation as a platinum sponsor & board member



Azure Cosmos DB: The industry's first globally-distributed, multi-model database service

Posted on May 10, 2017

Dharma Shukla, Distinguished Engineer and General Manager, Microsoft

Today, we're excited to announce the general availability of **Azure Cosmos DB**. Azure Cosmos DB is the first

May 17

Announcing Azure Cosmos DB

Microsoft joins Cloud Native Computing Foundation

Posted on July 26, 2017

John Gossman, Architect, Azure

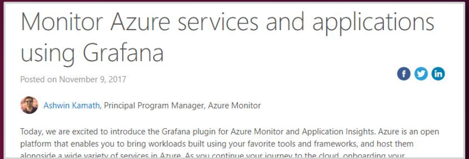
I'm excited to share that we have just joined the Cloud Native Computing Foundation (CNCF) as a Platinum member. CNCF is a part of the Linux Foundation, which helps govern for a wide range of cloud-oriented open source projects, such as Kubernetes, Prometheus, OpenTracing, Fluentd, Linkerd, containerd, Helm, gRPC, and many others.

Jul 17

Joined Cloud Native Computing Foundation as Platinum Member

Oct 17

Azure Container Service AKS (managed Kubernetes)



Monitor Azure services and applications using Grafana

Posted on November 9, 2017

Ashwin Kamath, Principal Program Manager, Azure Monitor

Today, we are excited to introduce the Grafana plugin for Azure Monitor and Application Insights. Azure is an open platform that enables you to bring workloads built using your favorite tools and frameworks, and host them alongside a wide variety of services in Azure. As you continue your journey to the cloud, onboarding your

Nov 17

Grafana plugin for Azure Monitor and Application Insights

Azure Databricks

Featured Notebooks

Introduction to Apache Spark on Databricks · Databricks for Data Scientists · Introduction to Structured Streaming

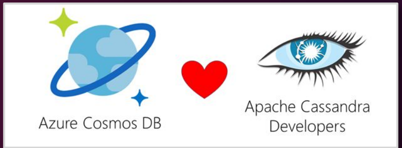
Nov 17

Azure Databricks

Apache Cassandra API in Azure Cosmos DB

Joined MariaDB Foundation

MariaDB, PostgreSQL & MySQL on Azure



Microsoft joins the MariaDB Foundation as a Platinum level sponsor

2017-11-15 | Written by Otto Kekäläinen

3 Comments

MariaDB Foundation today announced that Microsoft has become a platinum sponsor. The sponsorship will help the Foundation in its goals to support continuity and open collaboration in the MariaDB ecosystem, and to drive adoption, serving an ever growing community of users and developers.

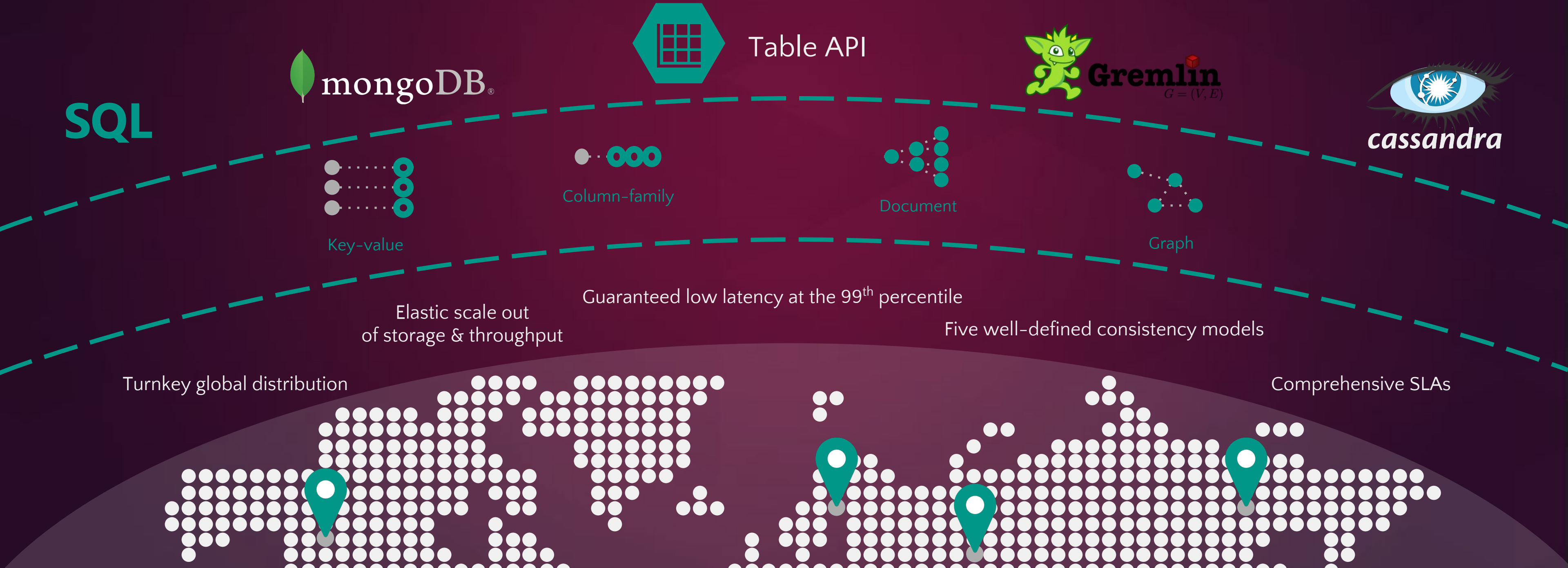
Azure

Upcoming

GrafanaCloud on Azure

Metrictank & Azure Cosmos DB

A globally distributed, massively scalable, multi-model database service


SQL
mongoDB
Table API
Gremlin
G = (V, E)
cassandra
Key-value
Column-family
Document
Graph
Elastic scale out
of storage & throughput
Guaranteed low latency at the 99th percentile
Five well-defined consistency models
Turnkey global distribution
Comprehensive SLAs