Linkedin

Building Centralized Caching Infrastructure at Scale

James Won

Staff Site Reliability Engineer

June 14, 2019

@jwon_me

Project
uth Sudan Health Pooled Fund
anitarian Response / Fund Management
ign, Delivery and Evaluation /

Question #1

Join at

slido.com

#CACHE

$ whoami

> jwon

- Joined LinkedIn in 2013
- Site Reliability Engineer for multiple teams, but most recently the Caching-as-a-Service team.
- Python Developer
- Vim user

@jwon_me

Project
uth Sudan Health Pooled Fund
nitarian Response / Fund Management
sign, Delivery and Evaluation /

Question #2

Join at

slido.com

#CACHE

🚫 This is NOT…

a talk about **Couchbase**, but a talk about **techniques** that can be applied anywhere.

Though if you are interested in learning more about how we’re using Couchbase, check out our blog post: https://engineering.linkedin.com/blog/2018/05/evolution-of-couchbase-at-linkedin

Agenda

1. History of Caching @ LinkedIn
2. Offering Caching as a Service
3. Challenges/ Takeaways
4. Future work

😣 Teams were frustrated with operating Memcached

- ➢ Losing the cache when nodes died
- ➢ Resizing & Replacing hosts was difficult
- ➢ Cache copying was difficult

🚪 Enter Couchbase

- Drop-in replacement
- Fast
- Persistent
- Replicas
- Resizing

Couchbase adoption exploded at LinkedIn 🧨

- Creation of a virtual team/working group to share learnings
- Over 2000 hosts in production
- Over 300 unique clusters

Growing Too Fast?

Lack of Operations Interest

Custom Deployment

Runaway Hardware Growth

@jwon_me

Question #3

Join at

slido.com

#CACHE

Project
uth Sudan Health Pooled Fund
anitarian Response / Fund Management
ign, Delivery and Evaluation /

Agenda

1 *History of Caching @ LinkedIn*

2 Offering Caching as a Service

3 Challenges/ Takeaways

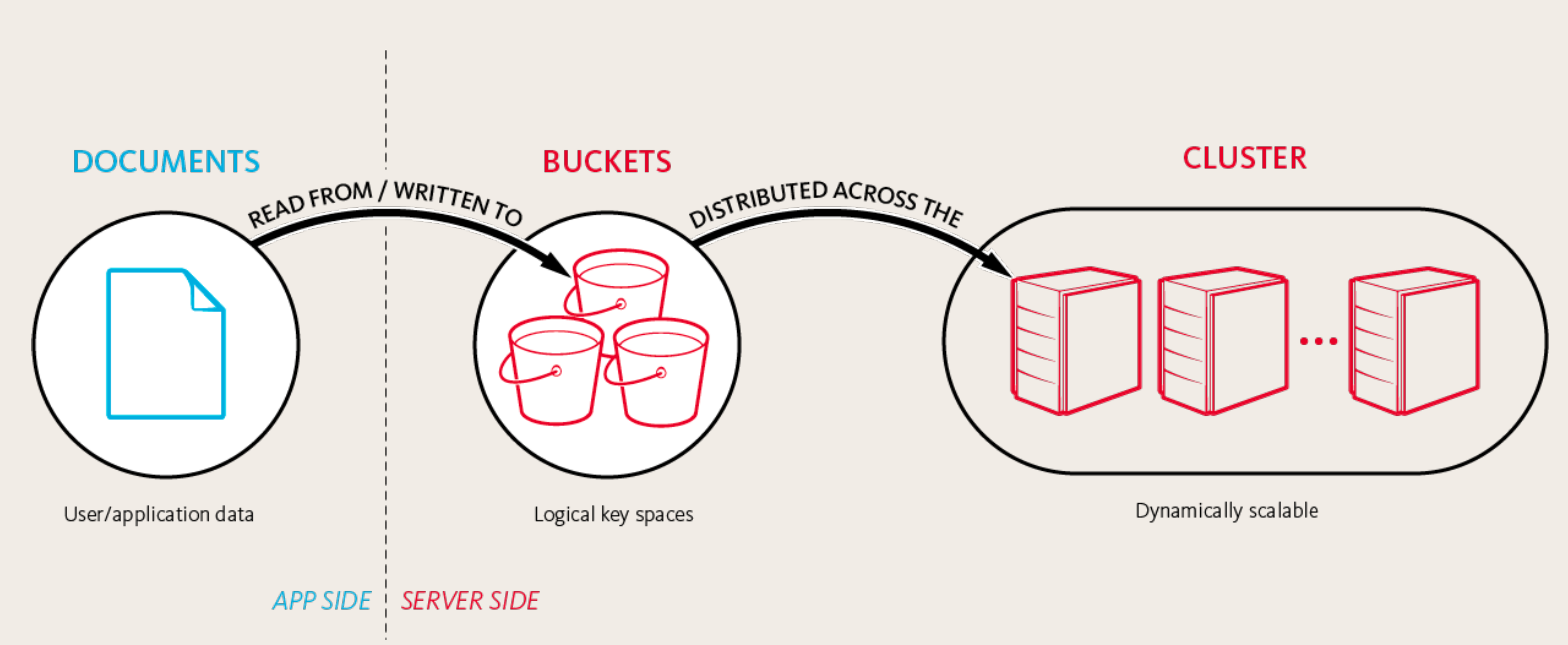
4 Future work

@jwon_me

Provide Caching as a Service (CaaS)

Centralize caching infrastructure to a team that is passionate about caching and wants to solve caching at scale

Couchbase Terminology



Source: https://docs.couchbase.com/server/4.1/concepts/buckets-vbuckets.html

@jwon_me

Caching as a Service

Build & Manage @ scale

Improve hardware efficiency

Improve security

CaaS Mission

“Provide **secure, high performance,** and **cost effective** caching to all teams within LinkedIn”

Project
uth Sudan Health Pooled Fund
anitarian Response / Fund Management
sign, Delivery and Evaluation /

Question #4

Join at

slido.com

#CACHE

Project
uth Sudan Health Pooled Fund
anitarian Response / Fund Management
sign, Delivery and Evaluation /

Question #5

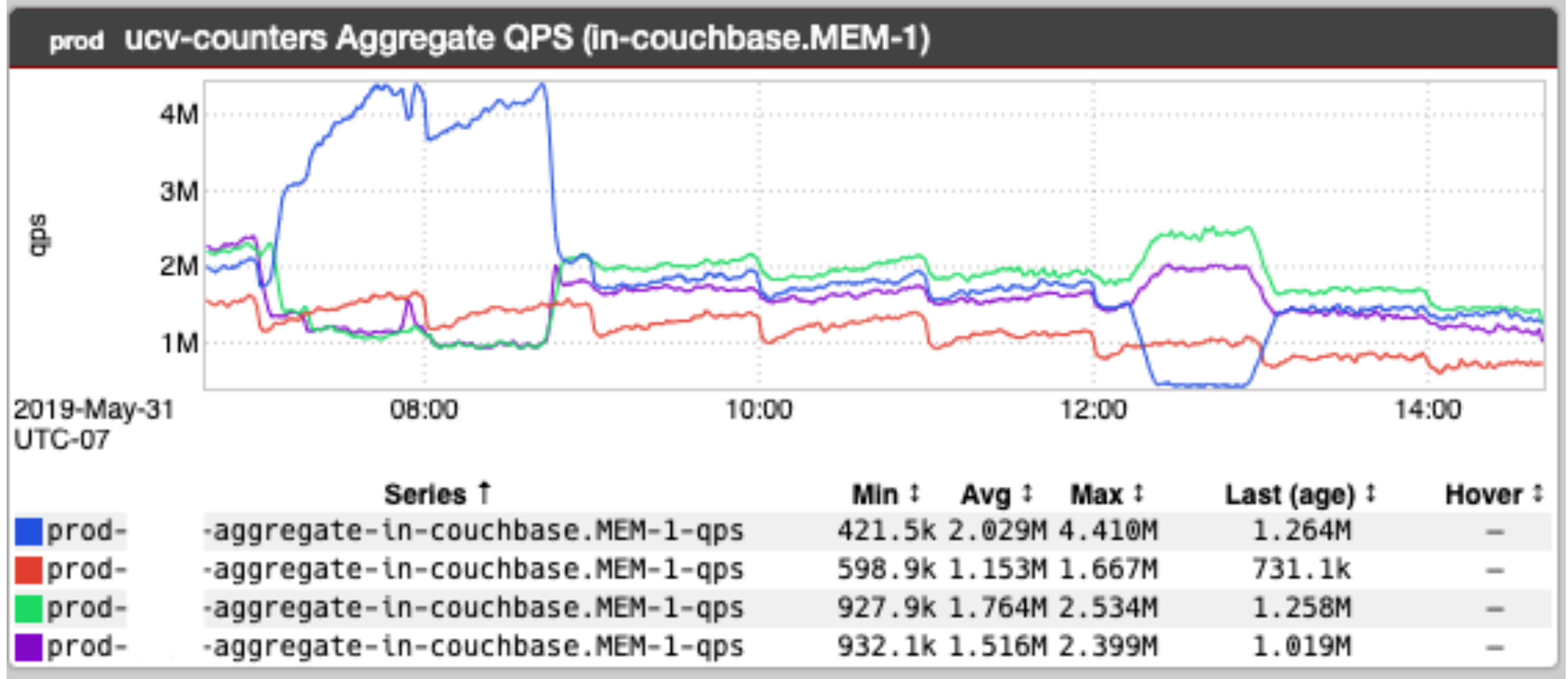
Join at

slido.com

#CACHE

What We Offer

- 0-1ms 95Pct latency for GETs/SETs
- 10ms SLO
- New!
  - Replication
  - Indexing



prod ucv-counters Aggregate QPS (in-couchbase.MEM-1)

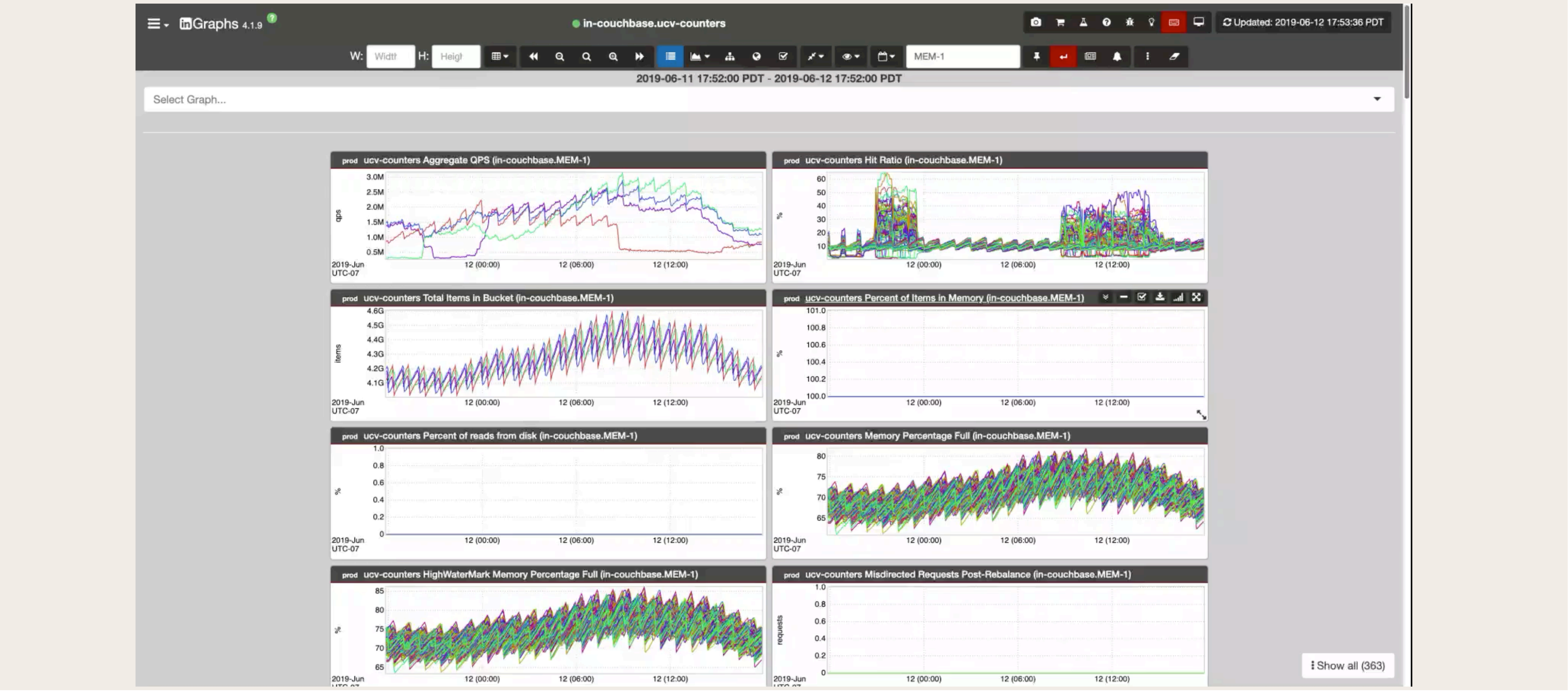
	Series ↑	Min ↕	Avg ↕	Max ↕	Last (age) ↕	Hover ↕
🟦	prod- -aggregate-in-couchbase.MEM-1-qps	421.5k	2.029M	4.410M	1.264M	–
🟥	prod- -aggregate-in-couchbase.MEM-1-qps	598.9k	1.153M	1.667M	731.1k	–
🟩	prod- -aggregate-in-couchbase.MEM-1-qps	927.9k	1.764M	2.534M	1.258M	–
🟪	prod- -aggregate-in-couchbase.MEM-1-qps	932.1k	1.516M	2.399M	1.019M	–

What We Offer

- SSD clusters, HDD clusters, Pure Memory clusters
- Informational dashboards for free
- Alerts on health and availability of cluster
- Handle hardware failures (host/switch/etc)
- Software & OS upgrades

@jwon_me

Informational dashboard auto-generated


inGraphs 4.1.9
in-couchbase.ucv-counters
Updated: 2019-06-12 17:53:36 PDT
W: Width
H: Height
MEM-1
2019-06-11 17:52:00 PDT - 2019-06-12 17:52:00 PDT
Select Graph...
prod ucv-counters Aggregate QPS (in-couchbase.MEM-1)
qps
3.0M
2.5M
2.0M
1.5M
1.0M
0.5M
prod ucv-counters Hit Ratio (in-couchbase.MEM-1)
%
60
50
40
30
20
10
prod ucv-counters Total Items in Bucket (in-couchbase.MEM-1)
items
4.6G
4.5G
4.4G
4.3G
4.2G
4.1G
prod ucv-counters Percent of Items in Memory (in-couchbase.MEM-1)
%
101.0
100.8
100.6
100.4
100.2
100.0
prod ucv-counters Percent of reads from disk (in-couchbase.MEM-1)
%
1.0
0.8
0.6
0.4
0.2
0
prod ucv-counters Memory Percentage Full (in-couchbase.MEM-1)
%
80
75
70
65
prod ucv-counters HighWaterMark Memory Percentage Full (in-couchbase.MEM-1)
%
85
80
75
70
65
prod ucv-counters Misdirected Requests Post-Rebalance (in-couchbase.MEM-1)
requests
1.0
0.8
0.6
0.4
0.2
0
2019-Jun
UTC-07
12 (00:00)
12 (06:00)
12 (12:00)
Show all (363)

@jwon_me

What We DON'T Offer

- Ownership of your data
- Backups

How much are we caching?

- >200 unique use cases in PROD of varying sizes
- >2000 hosts
- >10M qps across multiple clusters

@jwon_me

Agenda

1 *History of Caching @ LinkedIn*

2 *Offering Caching as a Service*

3 Challenges/ Takeaways

4 Future work

Project
uth Sudan Health Pooled Fund
nitarian Response / Fund Management
ign, Delivery and Evaluation /

Question #6

Join at

slido.com

#CACHE

Challenge #1

GDPR

- Needed to ensure GDPR compliance for over 200 unique caching use cases
- **Win:** The creation of a dedicated caching team allowed us to drive this initiative forward
- Most use cases did a data migration to our managed platform; built tooling around sizing and actual migration

@jwon_me

Challenge #2

Reconceptualize deployments

The old way

1. Hand-edit configuration files
2. Wait for them to sync (30m – 1h).
3. Run non-standard Salt runner to build cluster and hope it works.
4. If you made a mistake, go to step 1.

Challenge #2

Reconceptualize deployments

Solution

- Create wrapper around Couchbase Server to use supported standard deployment infrastructure
- Allow us to use standard deployment tooling for actions like upgrades, cluster expansions, etc.

Challenge #3

Getting off running as the root user

Why didn't we just run as non-root from the beginning?

- Couchbase didn't officially support it
- Small team, tight on resources. Ship something quickly as first iteration.
- RPM-based deployments was the path of least resistance to get started. Start with vendor-supported path.

@jwon_me

Challenge #3

Getting off running as the root user

Changing an engine of a car going 100 MPH

- Did in-place conversions to ”convert” a root node to a non-root node.
- Lot of code regarding file permissions as well as symlinking
- Switched to tarball-based deployments
- Removed conversion code and root code; removed almost 1000 lines of code!

@jwon_me

Takeaways

Treat your clusters as cattle 🐮, not pets 😾

This affects the way you think about automation and scale

Start with a core offering and iterate

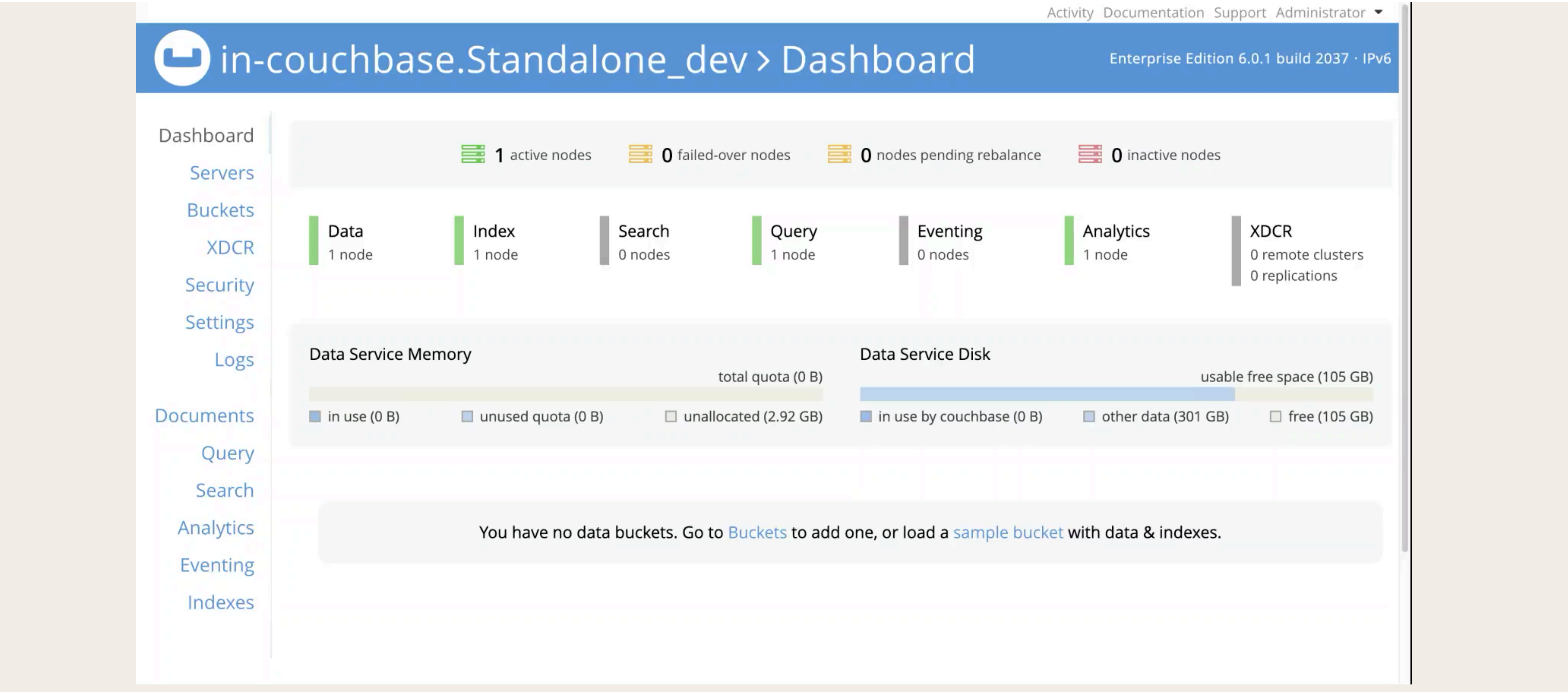
Intentionally started with a simple KV offering.

Expanded feature set (replication, indexing) as business needs grew

Codify checklists ✔️ into automation ⚙️

Runbooks/checklists are OK at first. Once process is documented, strive towards automation.

Adding a bucket manually



in-couchbase.Standalone_dev > Dashboard

Enterprise Edition 6.0.1 build 2037 · IPv6

1 active nodes **0** failed-over nodes **0** nodes pending rebalance **0** inactive nodes

Data	Index	Search	Query	Eventing	Analytics	XDCR
1 node	1 node	0 nodes	1 node	0 nodes	1 node	0 remote clusters 0 replications

Data Service Memory

total quota (0 B)

▨ in use (0 B) ▨ unused quota (0 B) □ unallocated (2.92 GB)

Data Service Disk

usable free space (105 GB)

▨ in use by couchbase (0 B) ▨ other data (301 GB) □ free (105 GB)

You have no data buckets. Go to Buckets to add one, or load a sample bucket with data & indexes.

Bad Design Decision

Bucket configuration in configs

- Originally put bucket configuration in application configs

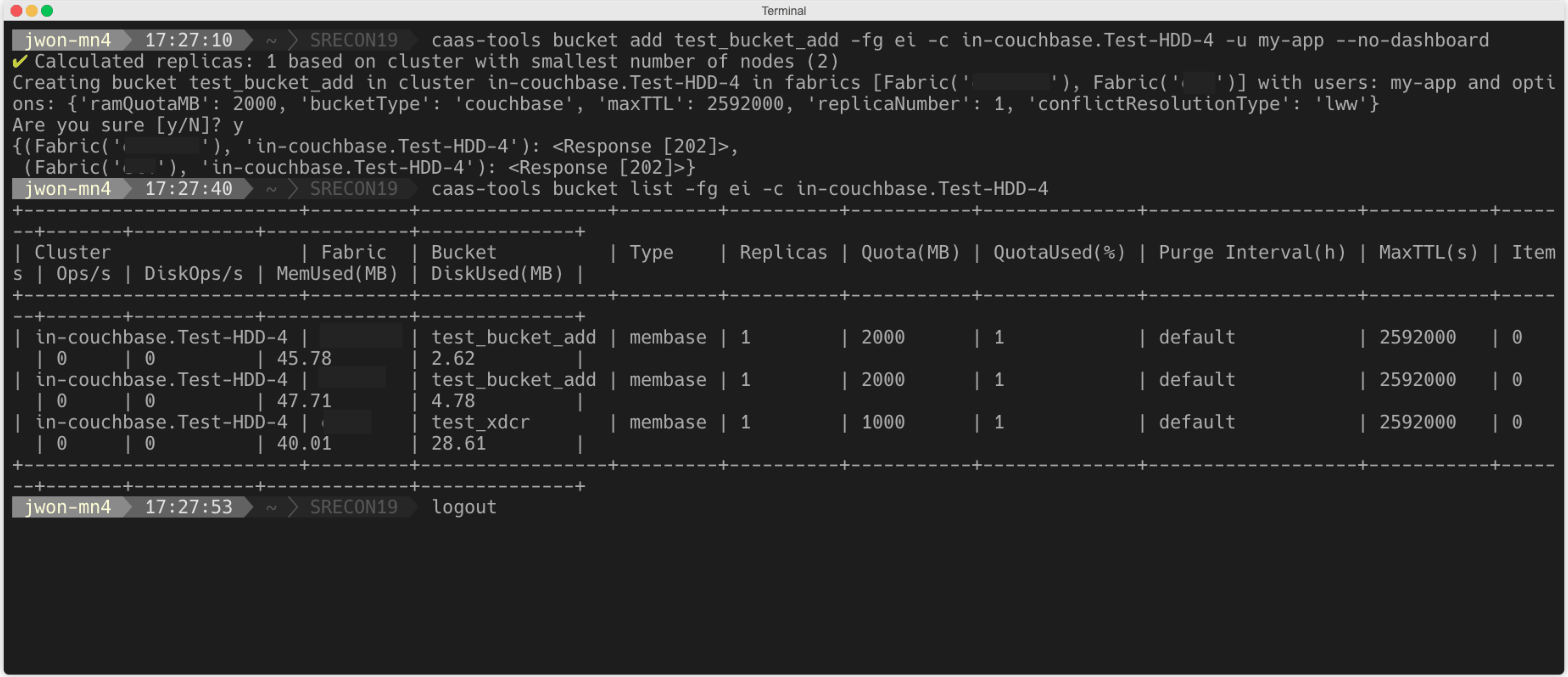
```
<entry key="tscp-tracking-fast-dedupe">
  <map>
    <entry key="bucket-ramsize" value="1000" />
    <entry key="bucket-replica" value="1" />
    <entry key="bucket-type" value="ephemeral" />
    <entry key="eviction-policy" value="nruEviction" />
    <entry key="authorized-users" value="tscp-tracking" />
  </map>
</entry>
```

Bad Design Decision

Bucket configuration in configs

- Need to deploy configs every time we needed to add/remove buckets
- Lot of unnecessary repetition -- change only needed to be applied on one host of the cluster

Add bucket using caas-tools



```
jwon-mn4  17:27:10  ~ 〉 SRECON19  caas-tools bucket add test_bucket_add -fg ei -c in-couchbase.Test-HDD-4 -u my-app --no-dashboard
✔ Calculated replicas: 1 based on cluster with smallest number of nodes (2)
Creating bucket test_bucket_add in cluster in-couchbase.Test-HDD-4 in fabrics [Fabric('[redacted]'), Fabric('[redacted]')] with users: my-app and opti
ons: {'ramQuotaMB': 2000, 'bucketType': 'couchbase', 'maxTTL': 2592000, 'replicaNumber': 1, 'conflictResolutionType': 'lww'}
Are you sure [y/N]? y
{(Fabric('[redacted]'), 'in-couchbase.Test-HDD-4'): <Response [202]>,
 (Fabric('[redacted]'), 'in-couchbase.Test-HDD-4'): <Response [202]>}
jwon-mn4  17:27:40  ~ 〉 SRECON19  caas-tools bucket list -fg ei -c in-couchbase.Test-HDD-4
+-------------------------+----------+-----------------+---------+----------+-----------+--------------+-------------------+-----------+-----
--+-------+-----------+-------------+--------------+
| Cluster                 | Fabric   | Bucket          | Type    | Replicas | Quota(MB) | QuotaUsed(%) | Purge Interval(h) | MaxTTL(s) | Item
s | Ops/s | DiskOps/s | MemUsed(MB) | DiskUsed(MB) |
+-------------------------+----------+-----------------+---------+----------+-----------+--------------+-------------------+-----------+-----
--+-------+-----------+-------------+--------------+
| in-couchbase.Test-HDD-4 | [redacted] | test_bucket_add | membase | 1        | 2000      | 1            | default           | 2592000   | 0
  | 0     | 0         | 45.78       | 2.62         |
| in-couchbase.Test-HDD-4 | [redacted] | test_bucket_add | membase | 1        | 2000      | 1            | default           | 2592000   | 0
  | 0     | 0         | 47.71       | 4.78         |
| in-couchbase.Test-HDD-4 | [redacted] | test_xdcr       | membase | 1        | 1000      | 1            | default           | 2592000   | 0
  | 0     | 0         | 40.01       | 28.61        |
+-------------------------+----------+-----------------+---------+----------+-----------+--------------+-------------------+-----------+-----
--+-------+-----------+-------------+--------------+
jwon-mn4  17:27:53  ~ 〉 SRECON19  logout
```

@jwon_me

Build platforms , not tools 🛠️

Rather than building tools for specific issues, build general APIs that can be pieced together to solve specific problems.

Trust your automation 🤖

It can be tempting to fall back to doing things the manual way when your automation/tooling fails. Don't. Instead use it as an opportunity to figure out root cause and improve it!

Agenda

1 *History of Caching @ LinkedIn*

2 *Offering Caching as a Service*

3 *Challenges/Take aways*

4 Future work

@jwon_me

Future work

- ✓ Self-Service provisioning
- ✓ Transparent migration of buckets across clusters
- ✓ Ease of shuffling nodes across clusters
- ✓ Better resource/hardware utilization

Thank you

@jwon_me LinkedIn