HACKATHON • CONFÉRENCES • INNOVATIONS

Monitoring OVH

300k servers, 27 DCs... and one Metrics platform

Horacio Gonzalez
@LostInBrittany

Kevin Georges
@0xd33d33

Steven Le Roux
@StevenLeRoux

Who are we?

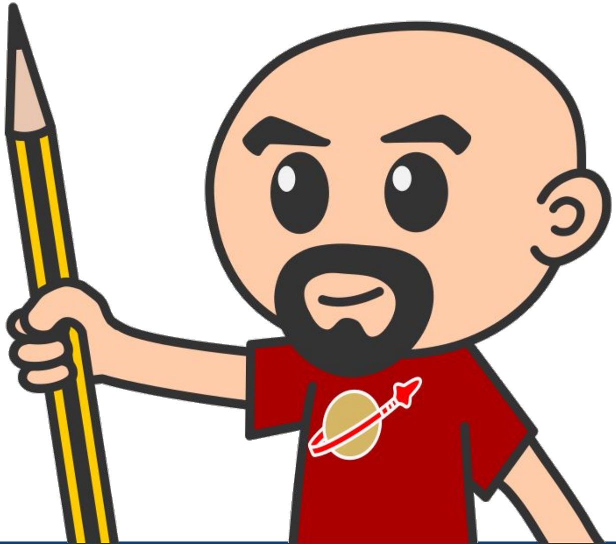
Introducing ourselves and introducing OVH

Horacio Gonzalez

@LostInBrittany

Spaniard lost in Brittany, developer, dreamer and all-around geek

OVH
Team DevRel



Breizh C@mp
Mix de technologies

Finist Devs
GDG Finistère

WARP 10 MEETUP

Google Developers Experts 2015
Web Technologies GDE

Kevin Georges

@0xd33d33

Engineering Manager

Working on Observability and Kubernetes

Distributed system addict

Warp10 / HBase / HDFS / Zookeeper / ETCD / Kubernetes

Finist Devs

WARP 10 MEETUP

Steven Le Roux

@StevenLeRoux

Principal Engineer

From networking to Distributed

Unconventional life rider

OVH : Key Figures

OVH
Innovation for Freedom

1.3M Customers worldwide in 138 Countries
1.5 Billions euros investment over five years
30 Datacenters (growing)
350k Dedicated Servers
200k Private cloud VMs running
650k Public cloud Instances created in a month
15TB bandwidth capacity
35 Points of presence
4TB Anti DDoS capacity
Hosting capacity : 1.3M Physical Servers

+ 2 500 Employees in 19 countries
19 Years of Innovation

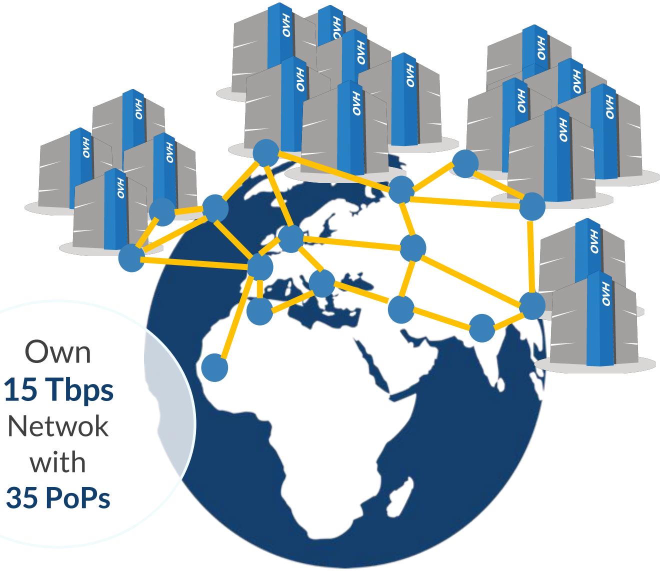
OVH: A Global Leader on Cloud

200k Private cloud VMs running

1 **Dedicated IaaS Europe**

Hosting capacity : **1.3M** Physical Servers

360k Servers already deployed



Own **15 Tbps** Netwok with **35 PoPs**

2018
27 Datacenters

2020
50 Datacenters

> **1.3M** Customers in **138** Countries

Ranking & Recognition

OVH
Innovation for Freedom

1st **European** Cloud Provider*

1st **Hosting** provider in Europe

1st **Provider** Microsoft Exchange

Certified vCloud Datacenter

Certified Kubernetes platform (CNCF)

Vmware **Global Service Provider** 2013-2016

Veeam Best Cloud Partner of the year (2018)

* Netcraft 2017 -

OVH: Our solutions

Cloud

- VPS
- Public Cloud
- Private Cloud
- Serveur dédié
- Cloud Desktop
- Hybrid Cloud

Mobile Hosting

- Containers
- Compute
- Database
- Object Storage
- Securities
- Messaging

Web Hosting

- Domain names
- Email
- CDN
- Web hosting
- MS Office
- MS solutions

Telecom

- VoIP
- SMS/Fax
- Virtual desktop
- Cloud HubiC
- Over theBox

Once upon a time...

Because we love telling tales

This talk is about a tale...

OVH
Innovation for Freedom

A true one nevertheless

And as in most tales

It begins with a mission

And a band of heroes

Engulfed into the adventure

They fight against mishaps

OVH
Innovation for Freedom

And all kind of foes

They build a mighty citadel

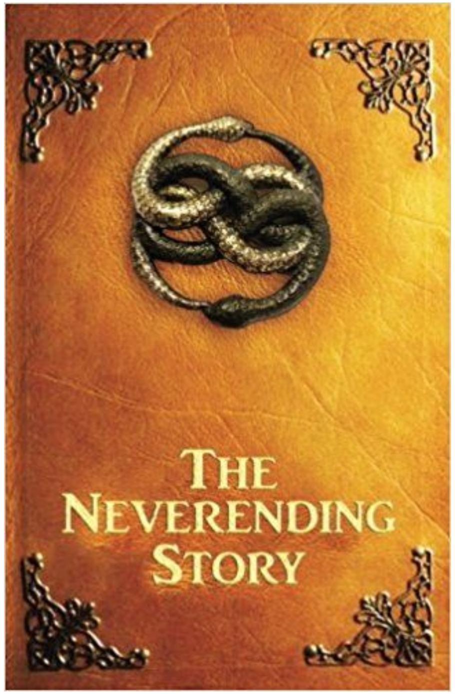
Pushing the limits of Physics

And defend them day after day

www.desktopextreme.com | Lord of the Rings - Return of the King

Against all odds

But we don't know yet the end



Because this tale isn't finished yet

It begins with a mission

Build a metrics platform for OVH

It began with a mission

Build a **metrics** platform for **OVH**

Why do we need metrics?

To make better **decisions**
by using **numbers**

Why do we need metrics?

We need to make better **decisions** about our **code**

Why do we need metrics?

We want our **code** to add **value**

Why do we need metrics?

Innovation for Freedom

Code adds **value** when it **runs**
not when we write it

Why do we need metrics?

We need to know what our code **does** when it **runs**

Why do we need metrics?

We can't do this
unless we **measure** it

Why do we need metrics?

We have a **mental model**
of what our code **does**

Why do we need metrics?

This **representation** can be **wrong**

Why do we need metrics?

We can't **know** until
we **measure** it

Find the bottleneck

“The app is slow.” - User

Find the bottleneck

“The app is slow.” - User

“The page takes 500ms!” - Ops

Find the bottleneck

SQL Query?

Template Rendering?

Session Storage?

Find the bottleneck

We don't know

Find the bottleneck

With observability:

SQL Query	53ms
Template Rendering	1ms
Session Storage	315ms

Find the bottleneck

With observability:

SQL Query	53ms
Template Rendering	1ms
Session Storage	315ms

Why do we need metrics?

We improve our mental model by **measuring** what our code **does**

Why do we need metrics?

We use our **mental model** to **decide** what to do

Why do we need metrics?

A better **mental model** makes us better at **deciding** what to do

Why do we need metrics?

Better **decisions** makes us better at generating **value**

Why do we need metrics?

OVH
Innovation for Freedom

Measuring make your
App **better**

It began with a mission

Build a **metrics** platform for **OVH**

A metrics platform for OVH

Building OVH Metrics

One Platform to unify them all,
One Platform to find them,
One Platform to bring them all
and in the Metrics monitor them

What is OVH Metrics?

Innovation for Freedom

Managed Cloud Platform
for Time Series

OVH monitoring story

We had lots of partial solutions...

OPENTSDB

mongoDB®

graphite

influxdb

OVH monitoring story

One Platform to unify them all

What should we build it on?

OVH monitoring story

First try

OpenTSDB drawbacks

OpenTSDB RowKey Design

metrics timestamp tagk1 tagv1 tagk2 tagv2

!

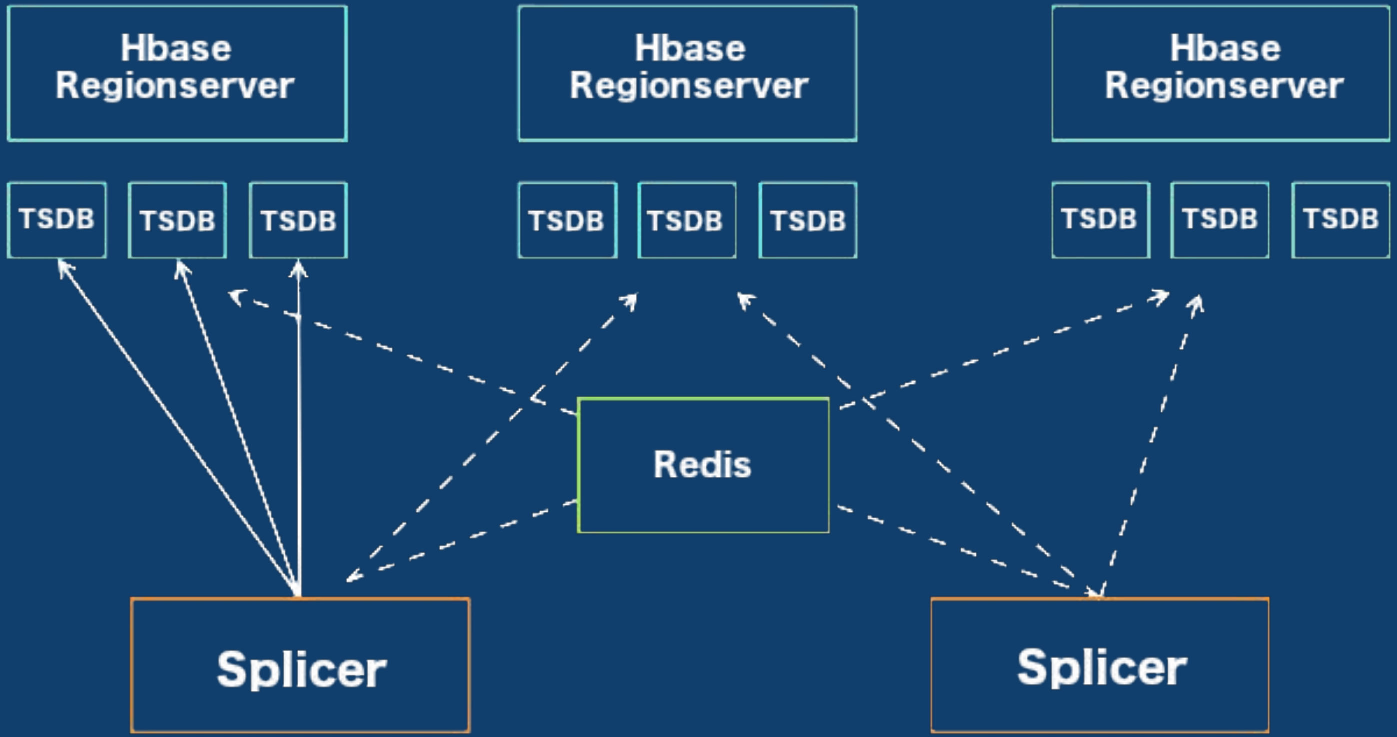
OpenTSDB Rowkey design flaws

- .*regex.* => full table scans
- Cardinality issues (Query latencies)

OpenTSDB other flaws

- Compactions (or append writes)
- /api/query : 1 endpoint per function?
- Asynchronous
- Unauthenticated
- ...

Scaling OpenTSDB



Metrics needs

First **need**:

To be **massively** scalable

Analytics is the key to success

Fetching data is only the tip of the iceberg

Analysing metrics data

To be scalable, analysis must be done in the database,
not in user's computer

Metrics needs

Second **need**:

To have **rich query** capabilities

Enter Warp 10...

WARP10

Open-source
Time series
Database

Cityzen
Data

PRODUIT
EN BRETAGNE

More than a Time Series DB

Warp 10 is a software platform that

- Ingests and stores time series
- Manipulates and analyzes time series

Manipulating Time Series with Warp 10

A true Time Series analysis toolbox

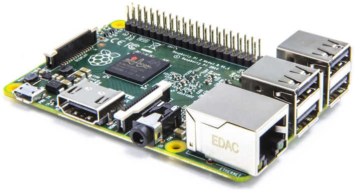
- Hundreds of functions
- Manipulation frameworks
- Analysis workflow

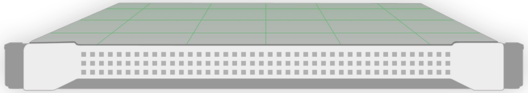
A Time Series manipulation language

WarpScript

Did you say scalability?

Innovation for Freedom



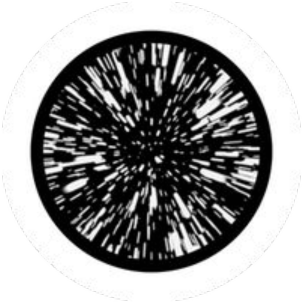


From the smallest to the largest…

Warp 10 goodness

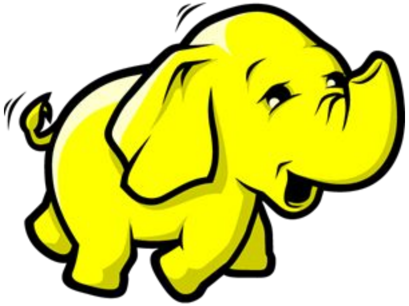
- Secured & multi tenant
- In memory Index
- No cardinality issues
- Lockfree ingestion
- WarpScript Query Language
- Support more data types
- Synchronous (transactions)
- Better Performance
- Better Scalability
- Versatile (standalone, distributed)

Metrics Data Platform

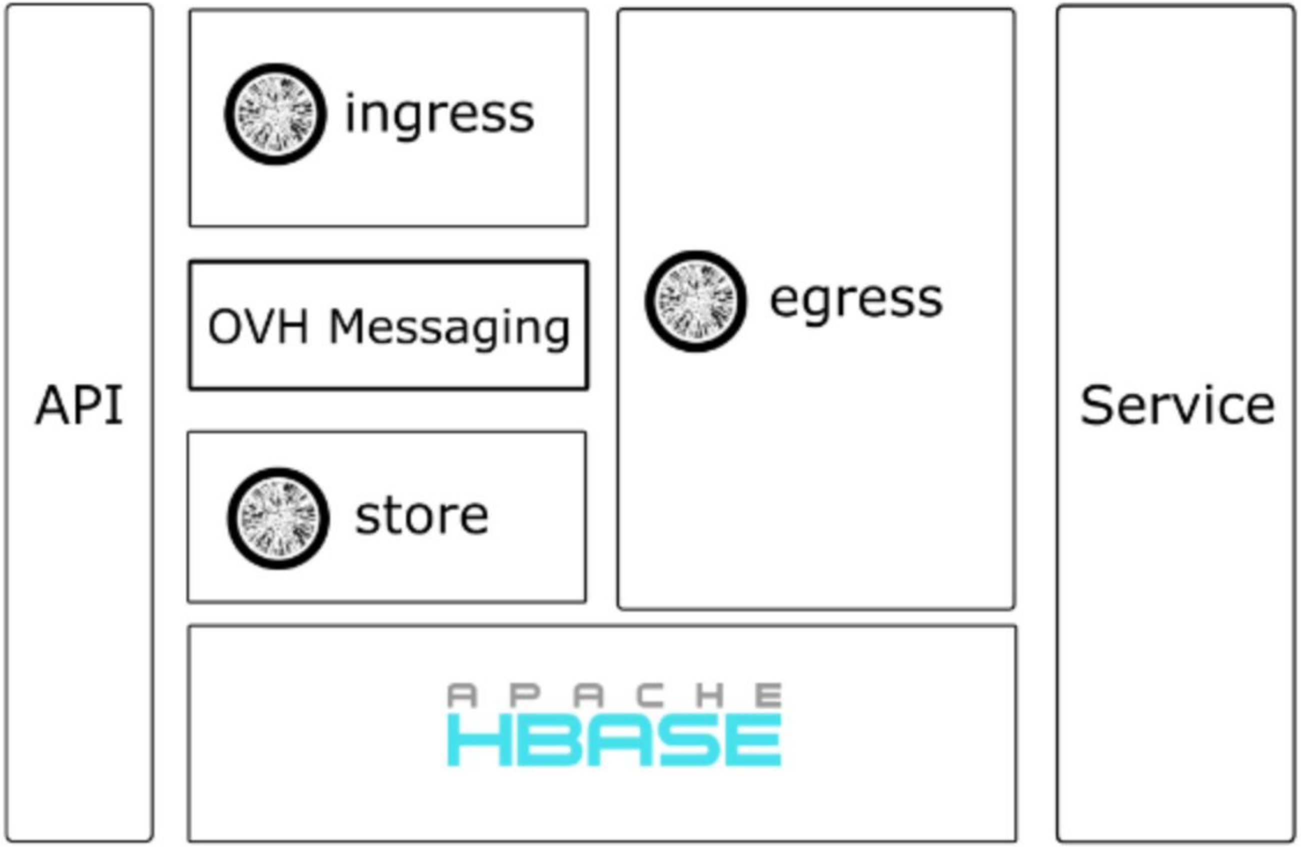


+

+



Metrics Data Platform



Leverage an ecosystem

and choose the right one...

Multi-protocol

Why to choose? We need them all!

Open source monitoring tools

Open source monitoring tools

Open source monitoring tools

Open source monitoring tools

Prometheus

Open source monitoring tools

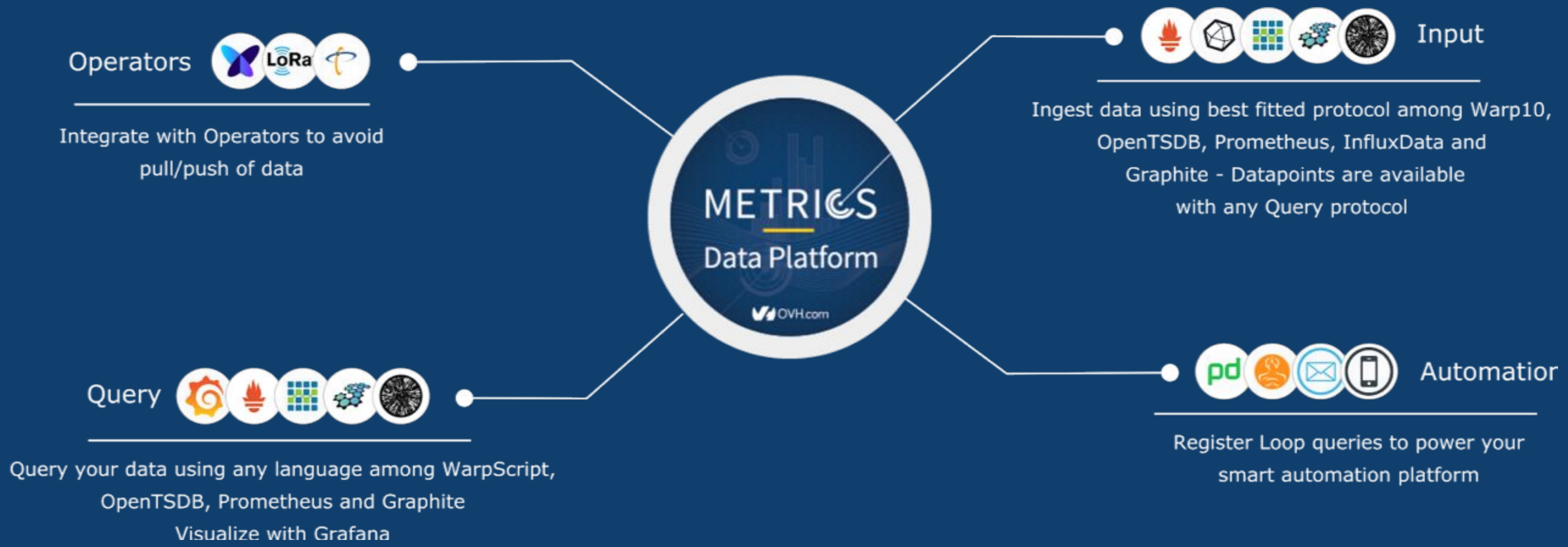
Open source monitoring tools

Open source monitoring tools

Why choose?

Let’s support all of them!

Metrics Platform



Operators

Integrate with Operators to avoid pull/push of data

Input

Ingest data using best fitted protocol among Warp10, OpenTSDB, Prometheus, InfluxData and Graphite - Datapoints are available with any Query protocol

Query

Query your data using any language among WarpScript, OpenTSDB, Prometheus and Graphite
Visualize with Grafana

Automatior

Register Loop queries to power your smart automation platform

Metrics Platform

graphite

influx

https:// **opentsdb** .<region>.metrics.ovh.net

prometheus

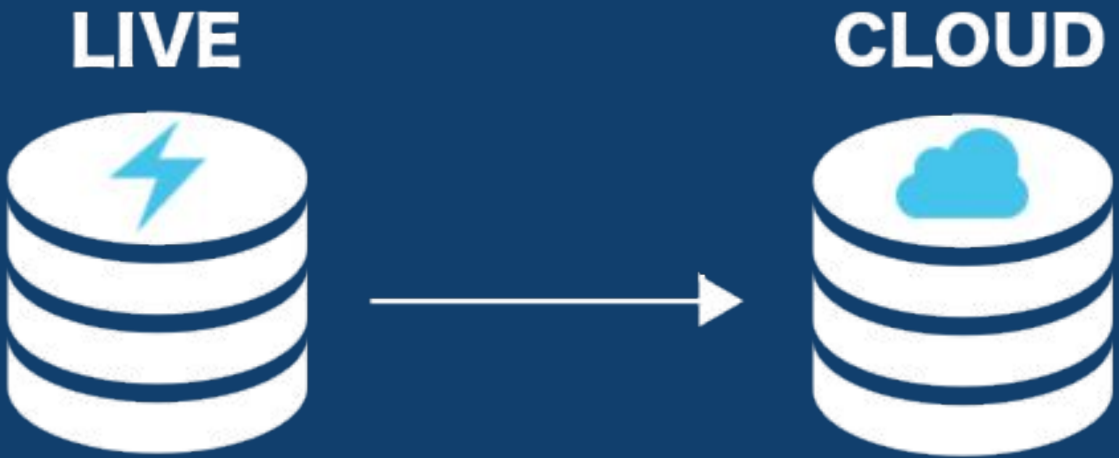
warp10

...

Metrics Live

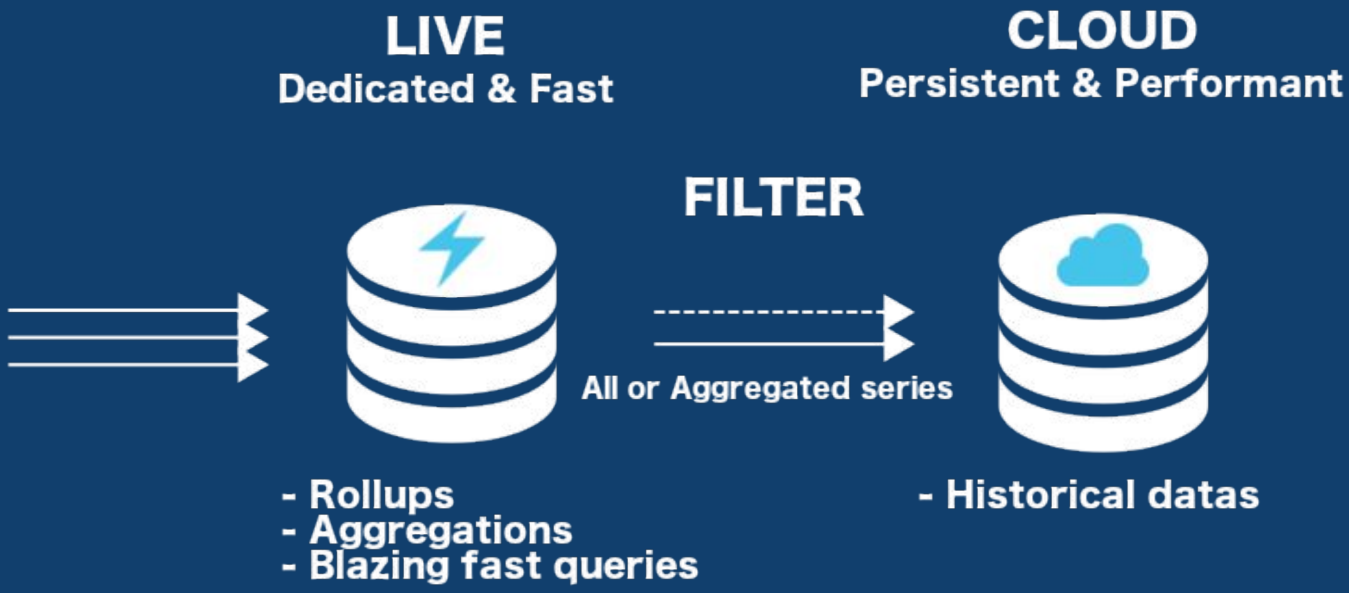
In-memory, high-performance Metrics instances

In-memory: Metrics live



+120 million of writes/s

In-memory: Metrics live



LIVE
Dedicated & Fast

CLOUD
Persistent & Performant

FILTER

All or Aggregated series

- Rollups
- Aggregations
- Blazing fast queries

- Historical datas

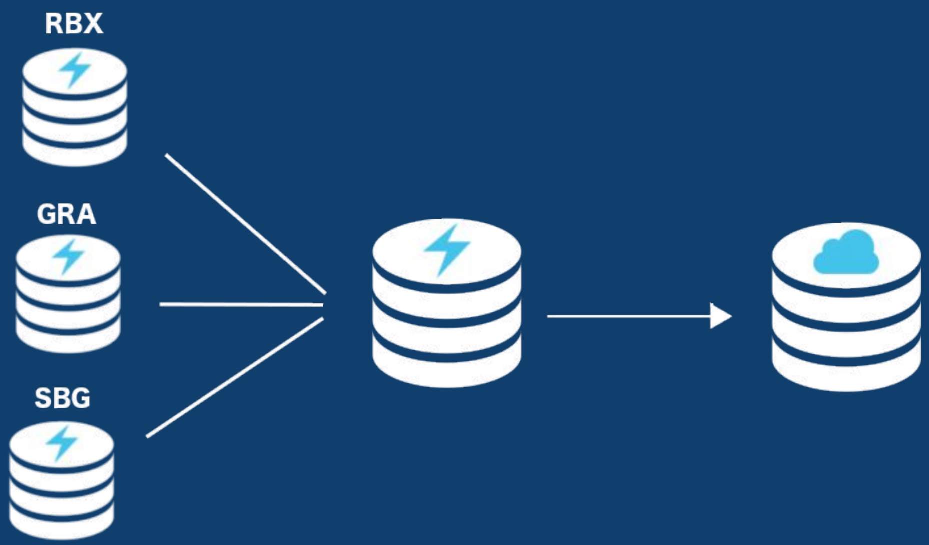
In-memory: Metrics live

OVH
Innovation for Freedom

STAGE 1
- Short retention - hours
- Fine grained monitoring
- Raw data

STAGE 2
- Short retention - days
- Consolidated aggregations
- Global infra monitoring

STAGE 3
- Customer metrics
- Historical datas



Monitoring is only the beginning

OVH Metrics answer to many other use cases

Use cases families

- Billing ....(e.g. bill on monthly max consumption)
- Monitoring ....(APM, infrastructure,appliances,...)
- IoT ....(Manage devices, operator integration, ...)
- Geo Location ....(Manage localized fleets)

Use cases

- DC Temperature/Elec/Cooling map
- Pay as you go billing (PCI/IPLB)
- GSCAN
- Monitoring
- ML Model scoring (Anti-Fraude)
- Pattern Detection for medical applications

Conclusion

That's all folks!

SREing Metrics

With a great power
comes great responsibility

Metrics' own metrics

432 000 000 000 datapoints / day

Metrics' own metrics

10 Tb / day

Metrics' own metrics

5 000 000 dp/s

Metrics' own metrics

500 000 000 series

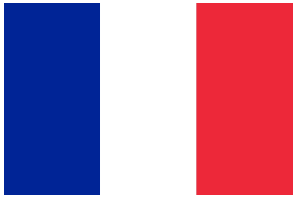
Our clusters size

GRA:

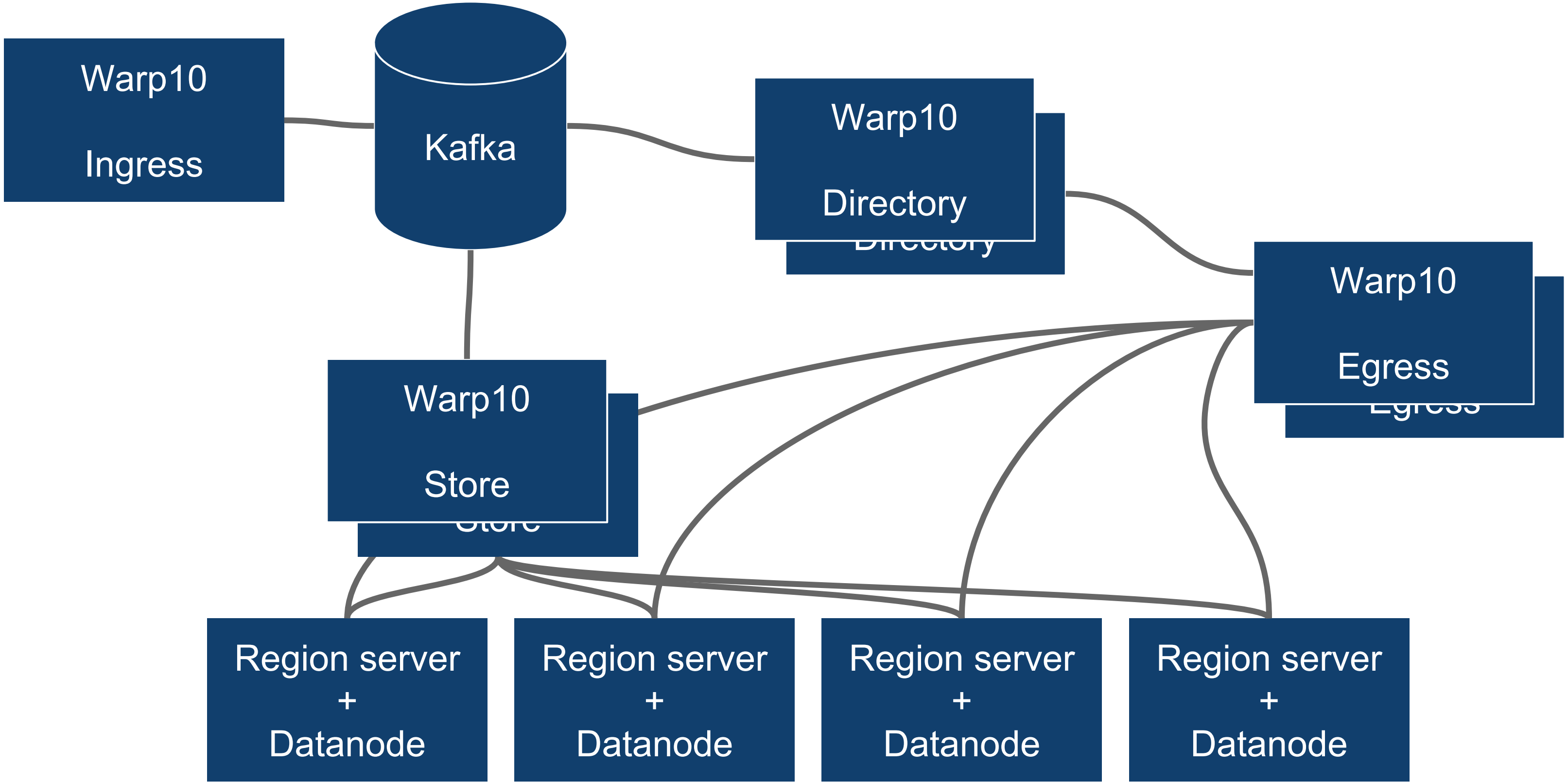
- 150 nodes
- 2 PB
- 1.1 Gbps

BHS:

- 30 nodes
- 400 TB
- 120 Mbps



Our cluster architecture



Detecting errors

Before it's too late

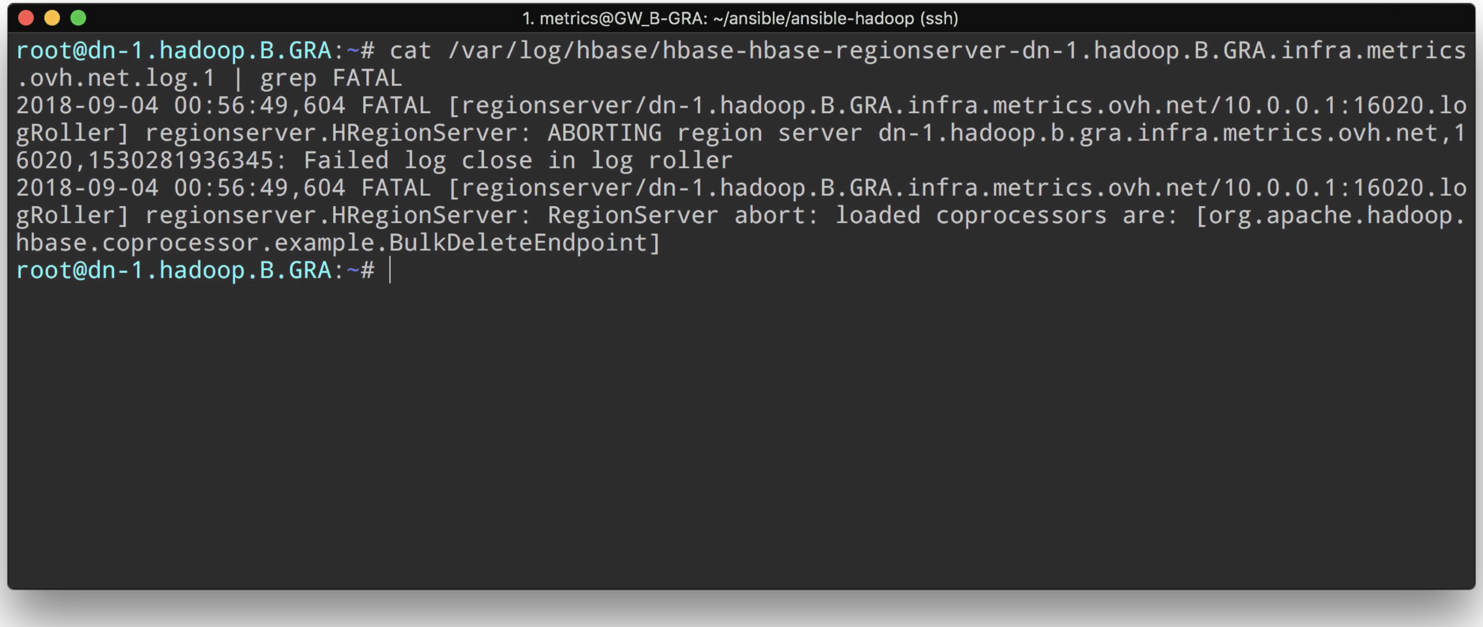
HBASE is designed to fail

It’s really good at it

HBASE fail in infinity ways

NETWORK	STORAGE	COMPUTE
Zookeeper timeout	Slow disk	Java GC
Network latency	Failed disk	Region compaction
Network bandwidth	Corrupted block	Delete handling
Handlers exhaustion		

Extract errors from logs



```
root@dn-1.hadoop.B.GRA:~# cat /var/log/hbase/hbase-hbase-regionserver-dn-1.hadoop.B.GRA.infra.metrics
.ovh.net.log.1 | grep FATAL
2018-09-04 00:56:49,604 FATAL [regionserver/dn-1.hadoop.B.GRA.infra.metrics.ovh.net/10.0.0.1:16020.lo
gRoller] regionserver.HRegionServer: ABORTING region server dn-1.hadoop.b.gra.infra.metrics.ovh.net,1
6020,1530281936345: Failed log close in log roller
2018-09-04 00:56:49,604 FATAL [regionserver/dn-1.hadoop.B.GRA.infra.metrics.ovh.net/10.0.0.1:16020.lo
gRoller] regionserver.HRegionServer: RegionServer abort: loaded coprocessors are: [org.apache.hadoop.
hbase.coprocessor.example.BulkDeleteEndpoint]
root@dn-1.hadoop.B.GRA:~# 
```

Tailor

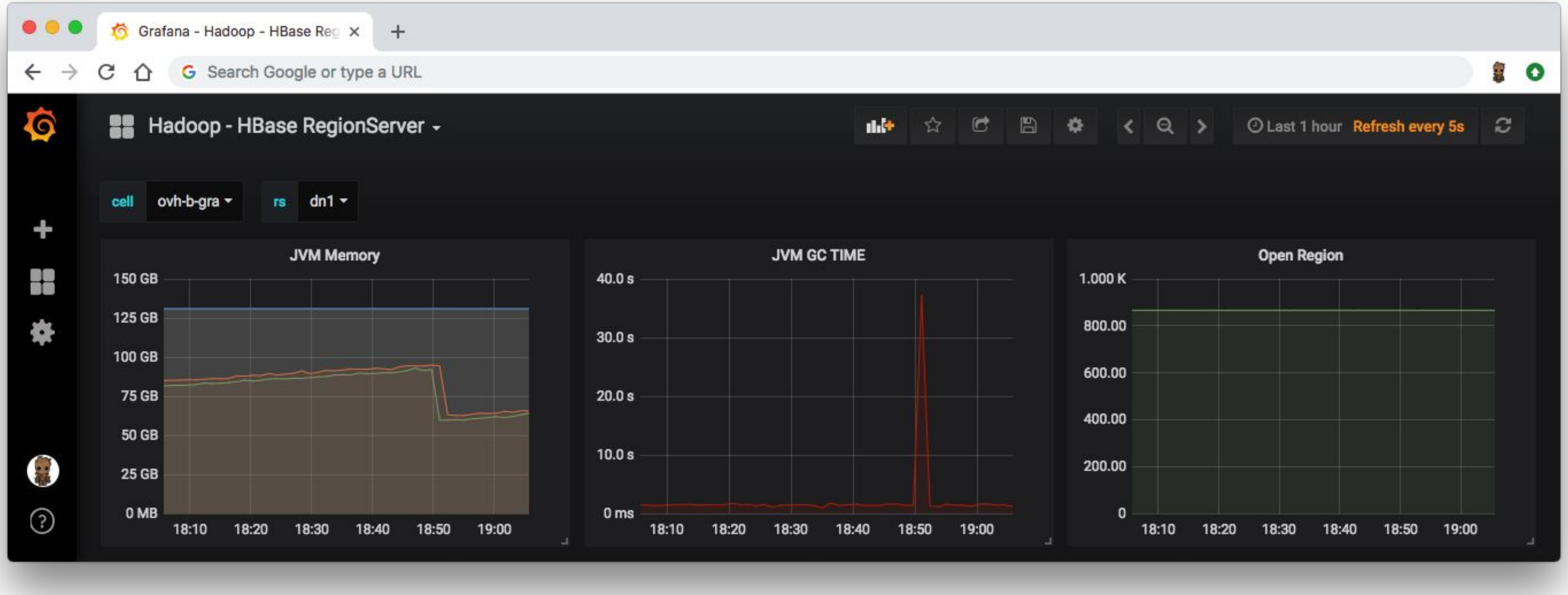
Filter logs

Extract metrics

Detect patterns

Perform correlations

Monitoring the JVM



Documentation

Documentation

The option `-XX:G1SummarizeRSetStatsPeriod` in combination with `gc+remset=trace` level logging shows if this coarsening occurs. If so, then the `X` in the line `Did <X> coarsenings` in the *Before GC Summary* section **shows a high value**. The `-XX:G1RSetRegionEntries` **option could be increased significantly** to decrease the amount of these coarsenings.

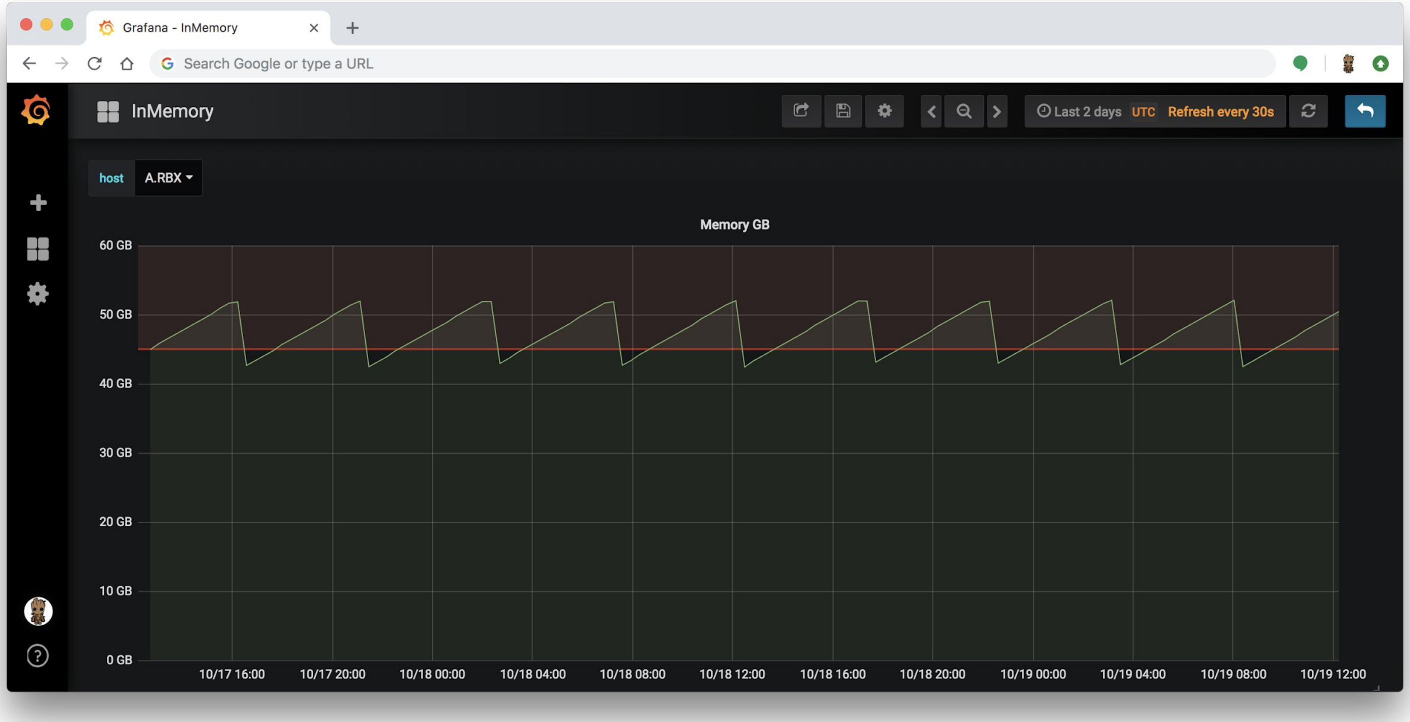
https://docs.oracle.com/javase/10/gctuning/garbage-first-garbage-collector-tuning.htm

Let's observe what is happening

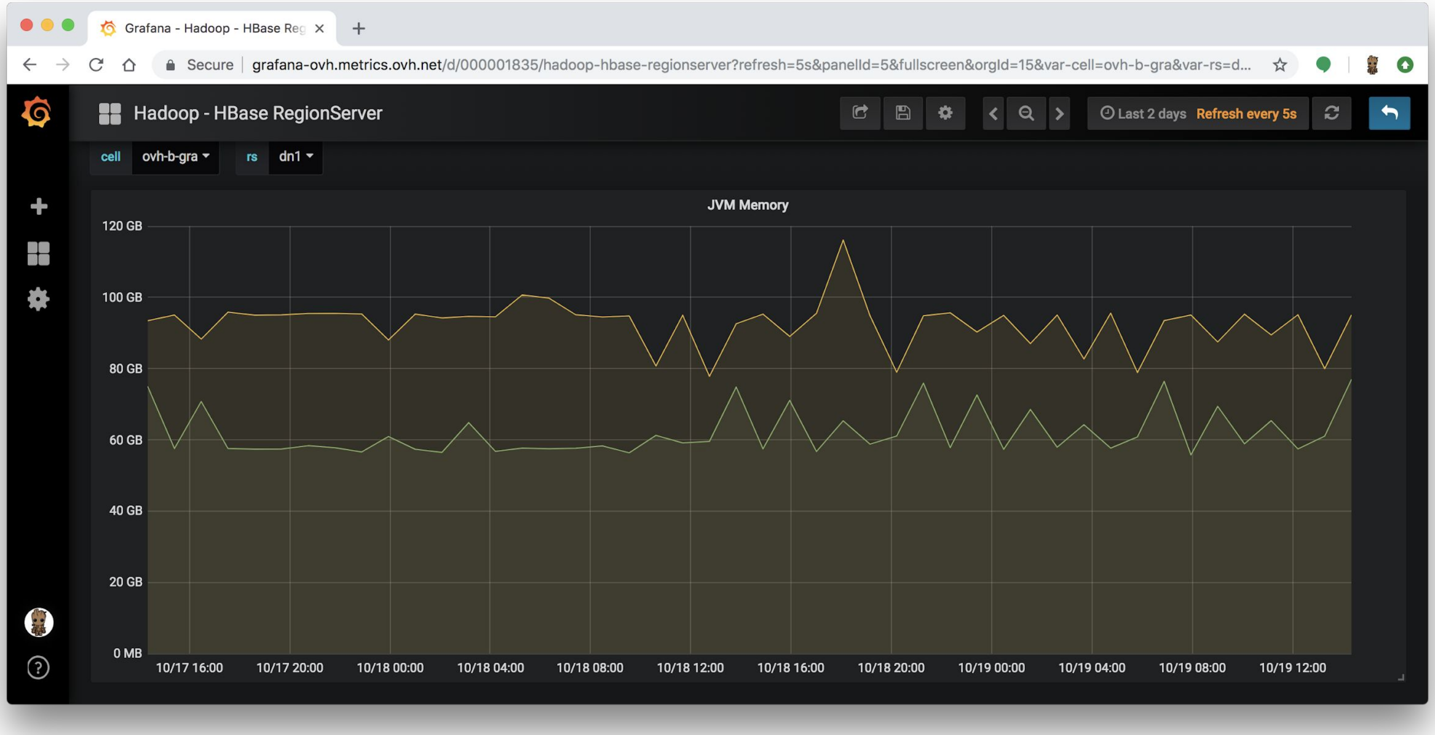
JVM GC

The good, the bad and the ugly

The good

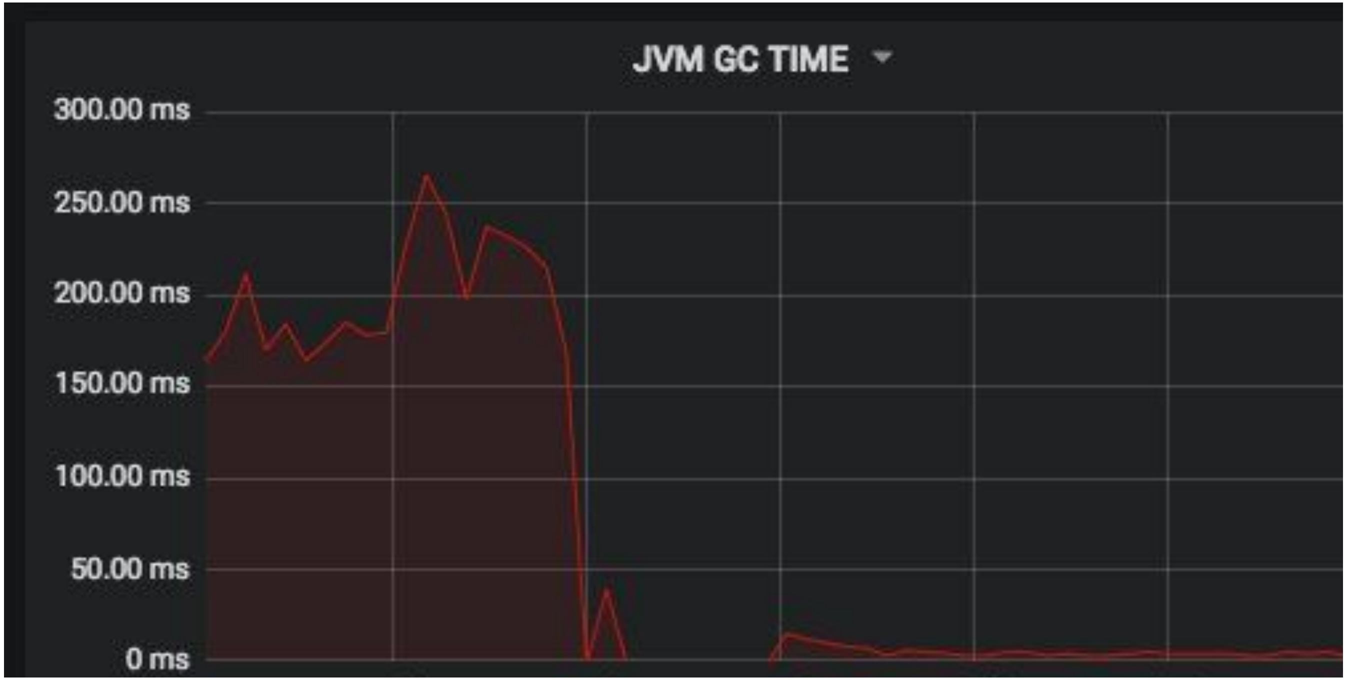

Grafana - InMemory
Search Google or type a URL
InMemory
Last 2 days UTC Refresh every 30s
host
A.RBX
Memory GB
60 GB
50 GB
40 GB
30 GB
20 GB
10 GB
0 GB
10/17 16:00
10/17 20:00
10/18 00:00
10/18 04:00
10/18 08:00
10/18 12:00
10/18 16:00
10/18 20:00
10/19 00:00
10/19 04:00
10/19 08:00
10/19 12:00

The bad



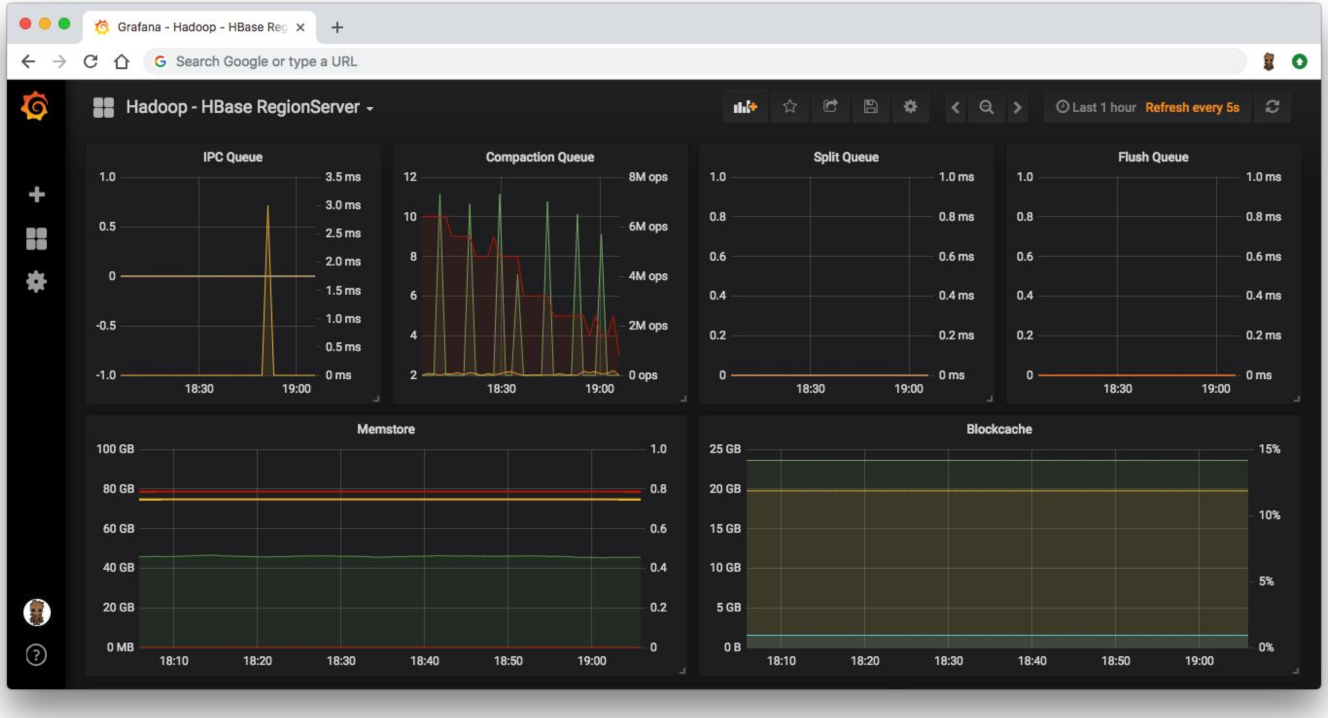
Grafana - Hadoop - HBase Reg
Secure
grafana-ovh.metrics.ovh.net/d/000001835/hadoop-hbase-regionserver?refresh=5s&panelId=5&fullscreen&orgId=15&var-cell=ovh-b-gra&var-rs=d...
Hadoop - HBase RegionServer
Last 2 days
Refresh every 5s
cell
ovh-b-gra
rs
dn1
JVM Memory
120 GB
100 GB
80 GB
60 GB
40 GB
20 GB
0 MB
10/17 16:00
10/17 20:00
10/18 00:00
10/18 04:00
10/18 08:00
10/18 12:00
10/18 16:00
10/18 20:00
10/19 00:00
10/19 04:00
10/19 08:00
10/19 12:00

… and the ugly



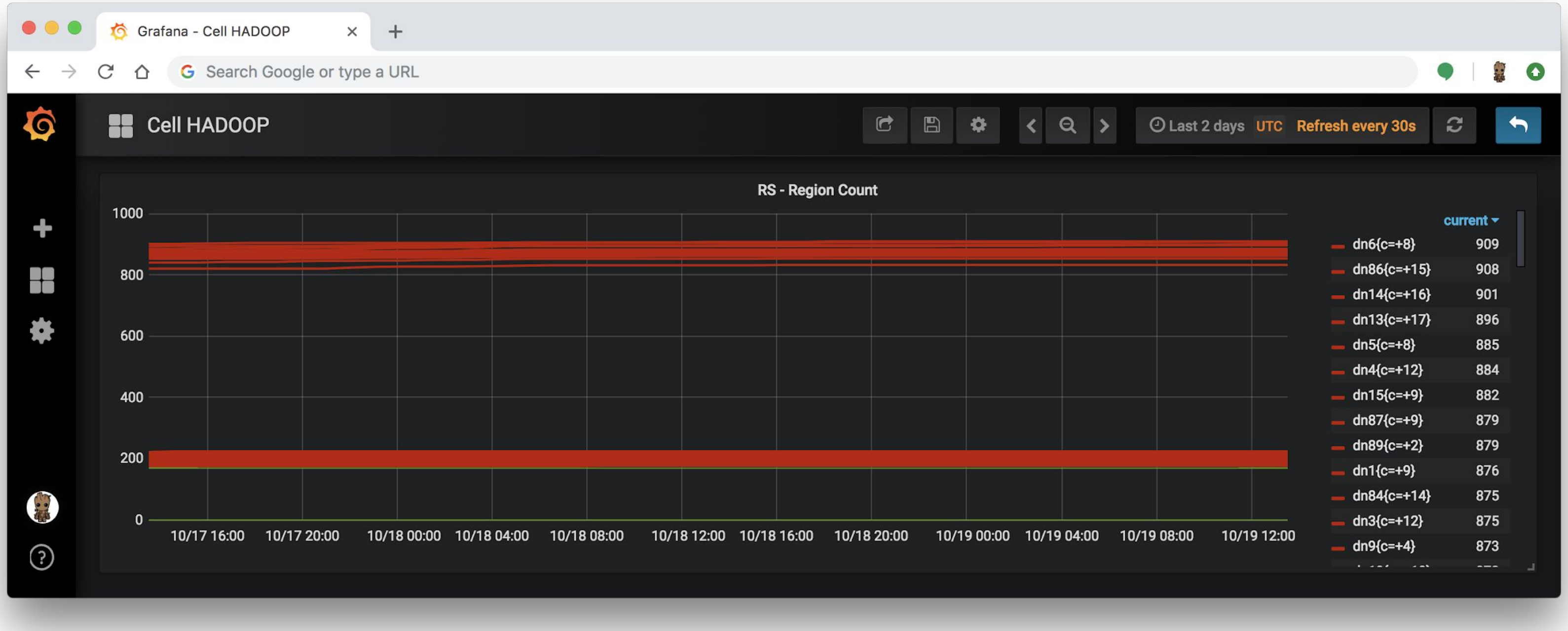
#java #jdk11 #zgc

Monitoring HBase



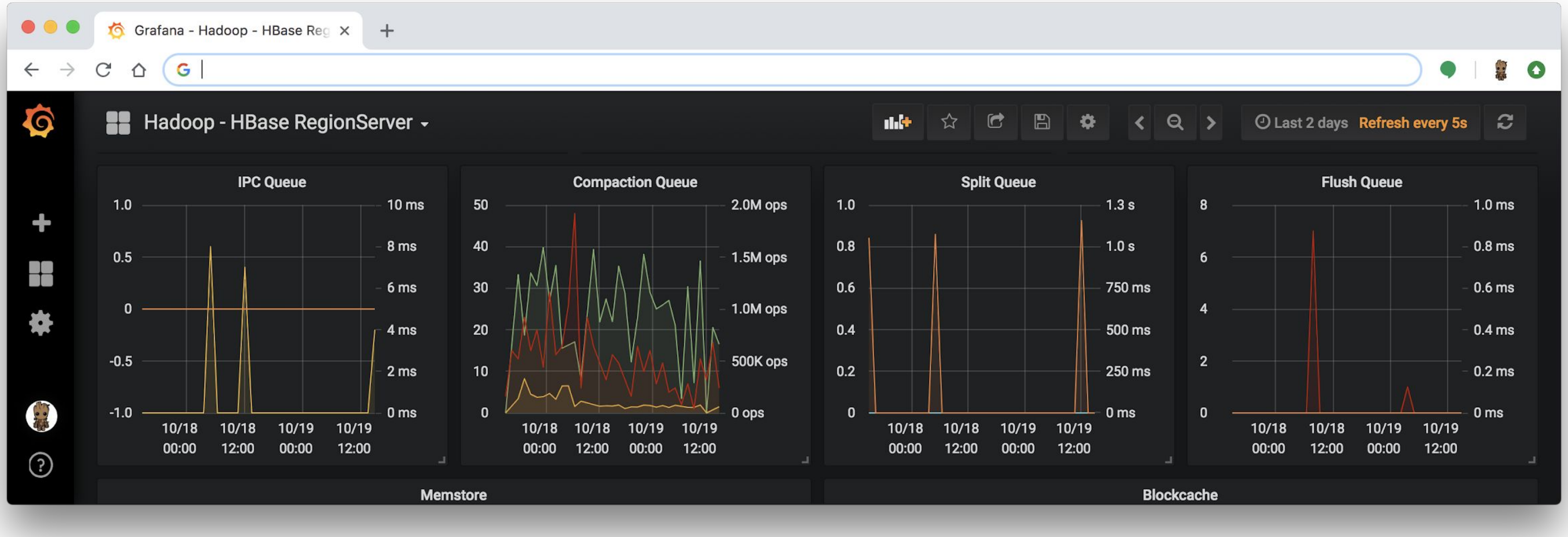
Number of open regions

OVH
Innovation for Freedom

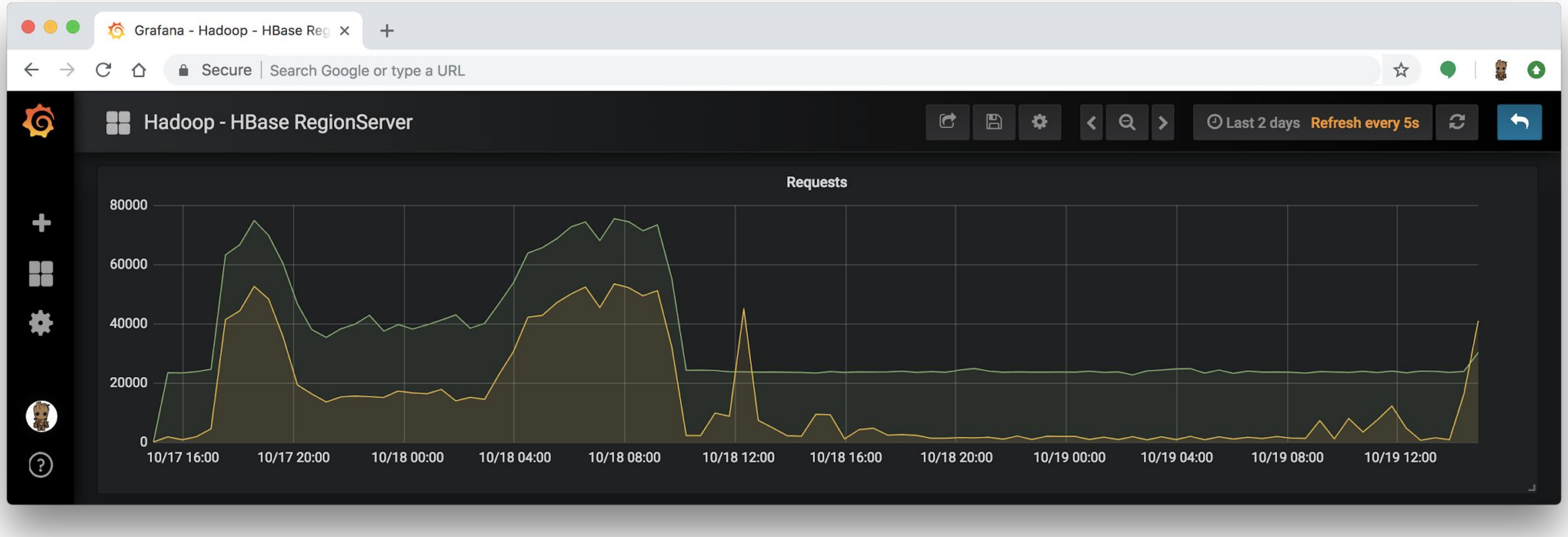


	current
dn6{c=+8}	909
dn86{c=+15}	908
dn14{c=+16}	901
dn13{c=+17}	896
dn5{c=+8}	885
dn4{c=+12}	884
dn15{c=+9}	882
dn87{c=+9}	879
dn89{c=+2}	879
dn1{c=+9}	876
dn84{c=+14}	875
dn3{c=+12}	875
dn9{c=+4}	873

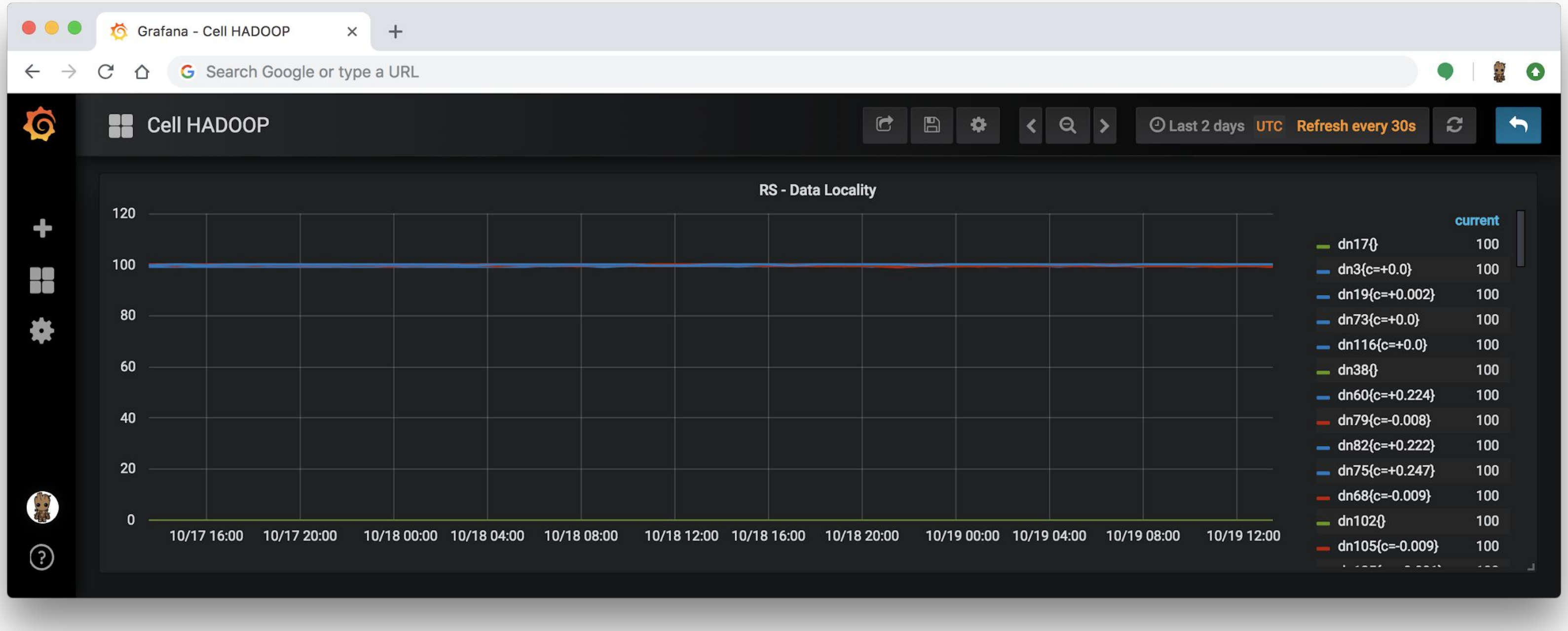
Queues length



Number of read and write requests



Data locality



Grafana - Cell HADOOP

Search Google or type a URL

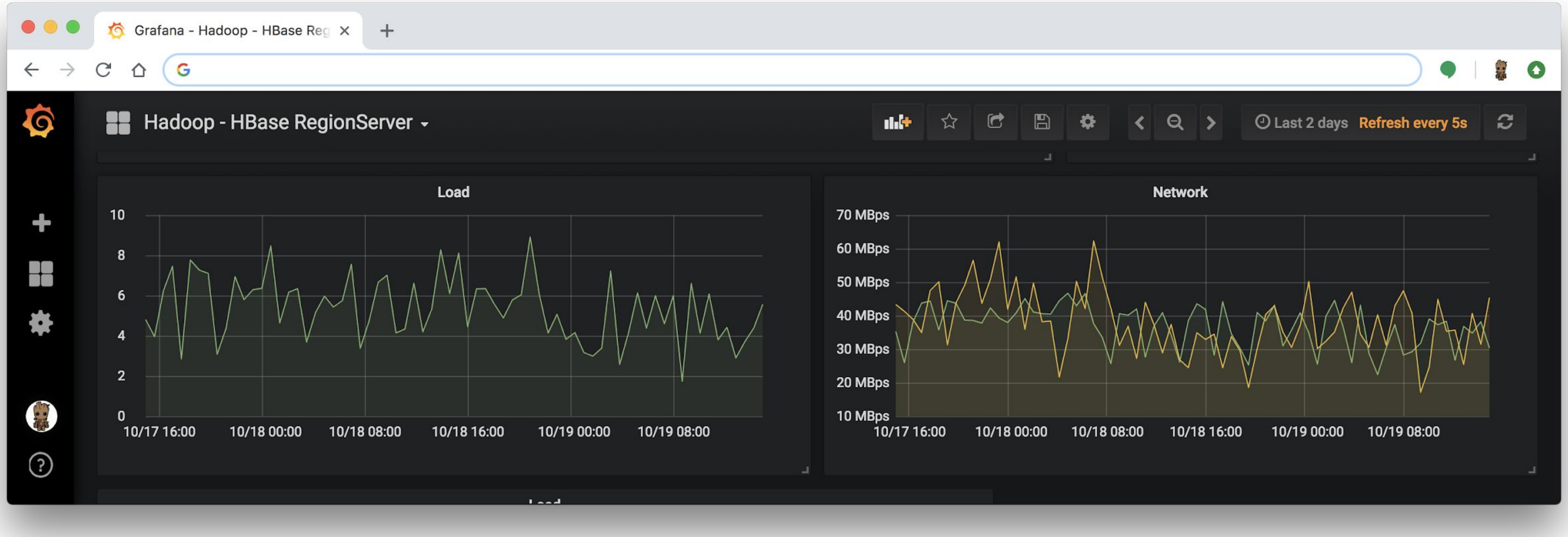
Cell HADOOP

Last 2 days UTC Refresh every 30s

RS - Data Locality

	current
dn17{}	100
dn3{c=+0.0}	100
dn19{c=+0.002}	100
dn73{c=+0.0}	100
dn116{c=+0.0}	100
dn38{}	100
dn60{c=+0.224}	100
dn79{c=-0.008}	100
dn82{c=+0.222}	100
dn75{c=+0.247}	100
dn68{c=-0.009}	100
dn102{}	100
dn105{c=-0.009}	100

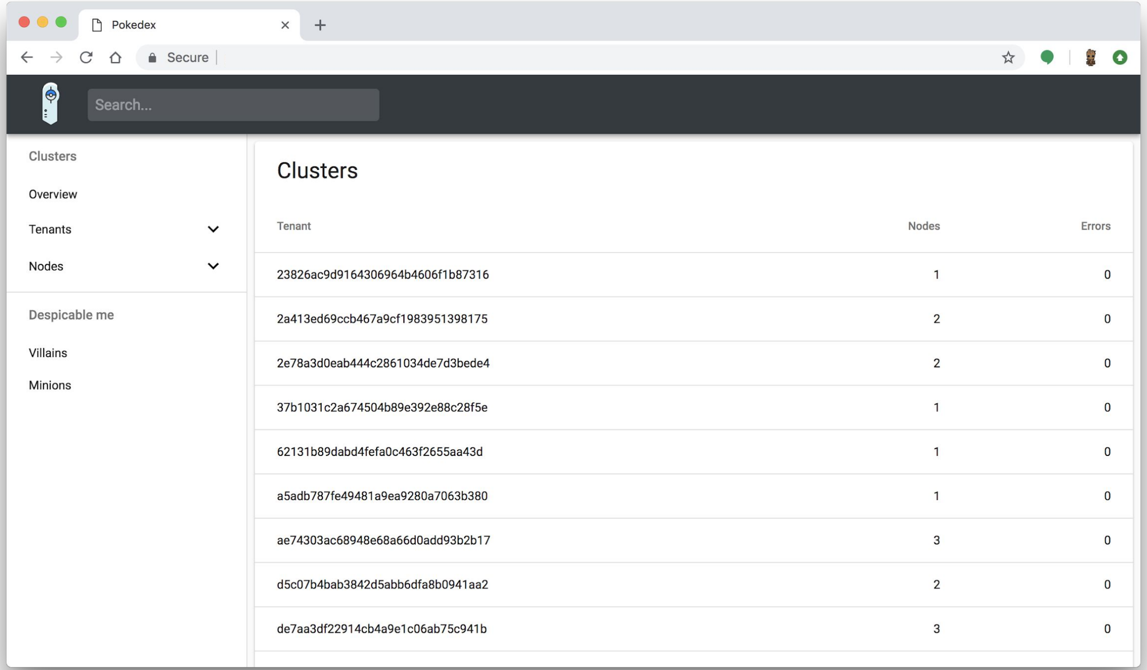
Host health


Grafana - Hadoop - HBase Reg
Hadoop - HBase RegionServer
Last 2 days
Refresh every 5s
Load
10
8
6
4
2
0
10/17 16:00
10/18 00:00
10/18 08:00
10/18 16:00
10/19 00:00
10/19 08:00
Network
70 MBps
60 MBps
50 MBps
40 MBps
30 MBps
20 MBps
10 MBps
10/17 16:00
10/18 00:00
10/18 08:00
10/18 16:00
10/19 00:00
10/19 08:00

Pokédex & Pokeball

Inventory all animals.

Merging all data sources

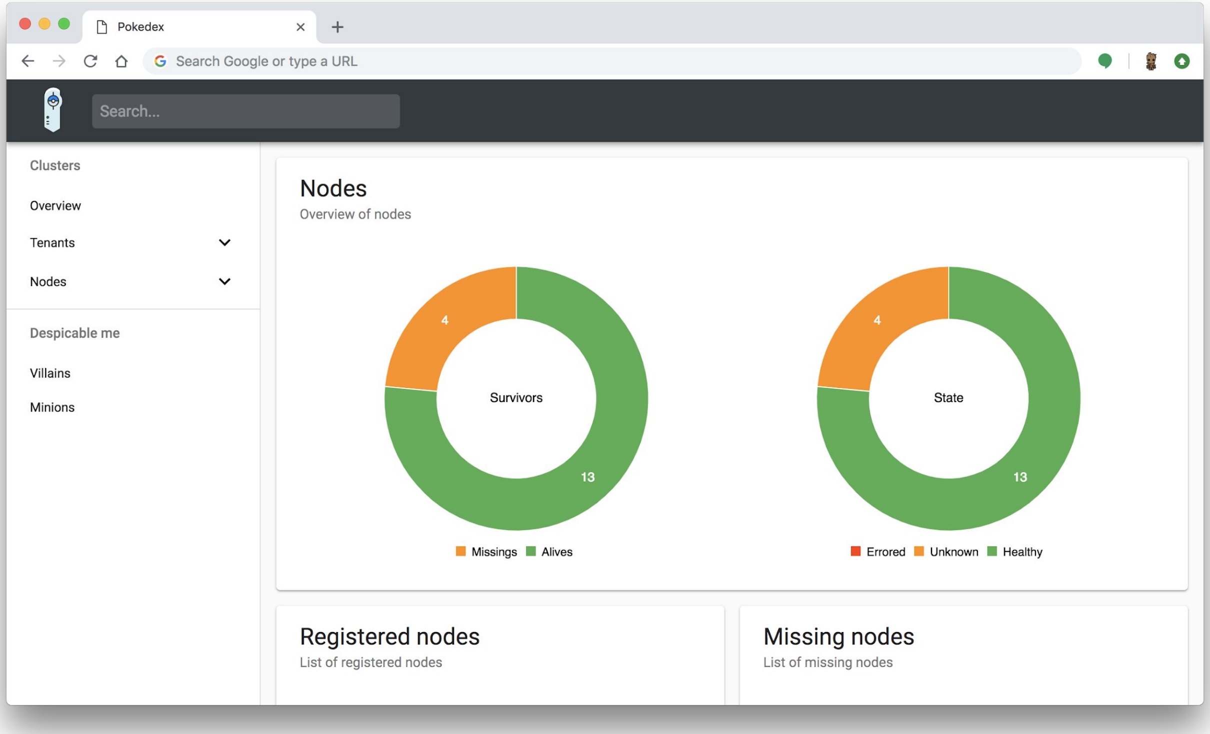


Clusters

Tenant	Nodes	Errors
23826ac9d9164306964b4606f1b87316	1	0
2a413ed69ccb467a9cf1983951398175	2	0
2e78a3d0eab444c2861034de7d3bede4	2	0
37b1031c2a674504b89e392e88c28f5e	1	0
62131b89dabd4fefa0c463f2655aa43d	1	0
a5adb787fe49481a9ea9280a7063b380	1	0
ae74303ac68948e68a66d0add93b2b17	3	0
d5c07b4bab3842d5abb6dfa8b0941aa2	2	0
de7aa3df22914cb4a9e1c06ab75c941b	3	0

Global visualization

OVH
Innovation for Freedom



Nodes

Overview of nodes

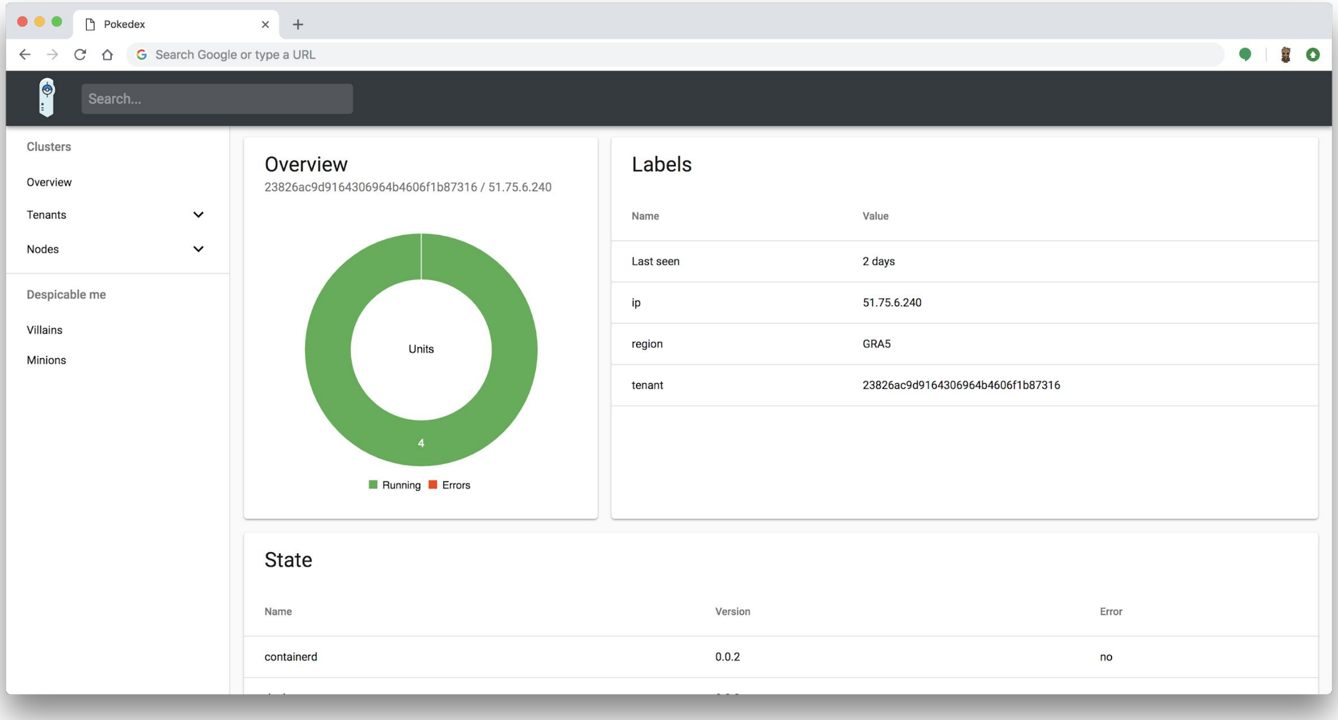
Registered nodes

List of registered nodes

Missing nodes

List of missing nodes

Correlate information



Overview

23826ac9d9164306964b4606f1b87316 / 51.75.6.240

Labels

Name	Value
Last seen	2 days
ip	51.75.6.240
region	GRA5
tenant	23826ac9d9164306964b4606f1b87316

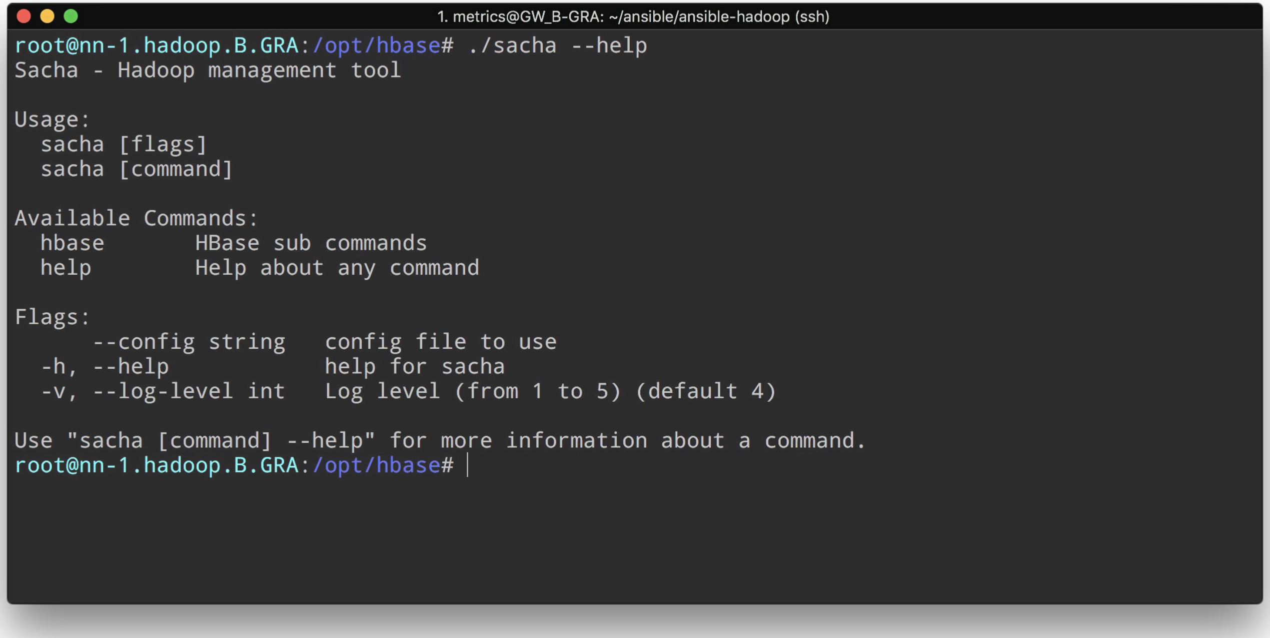
State

Name	Version	Error
containerd	0.0.2	no

Sacha

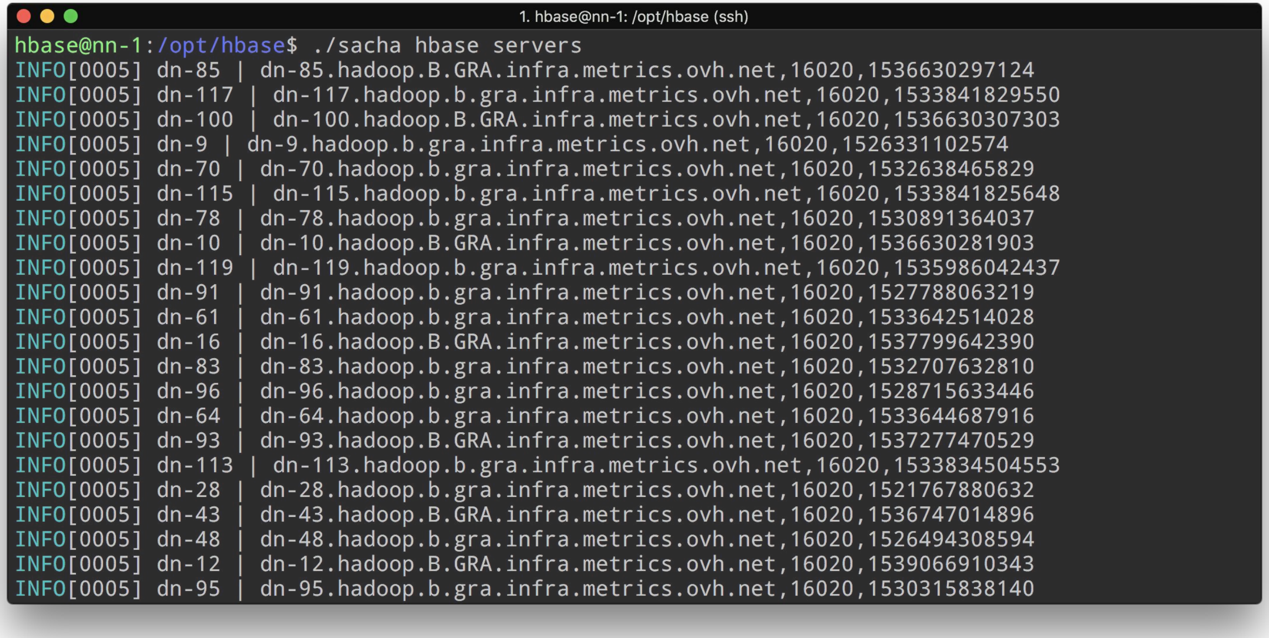
The best tamer

An awesome CLI



```
root@nn-1.hadoop.B.GRA:/opt/hbase# ./sacha --help
Sacha - Hadoop management tool

Usage:
  sacha [flags]
  sacha [command]

Available Commands:
  hbase       HBase sub commands
  help        Help about any command

Flags:
      --config string   config file to use
  -h, --help            help for sacha
  -v, --log-level int   Log level (from 1 to 5) (default 4)

Use "sacha [command] --help" for more information about a command.
root@nn-1.hadoop.B.GRA:/opt/hbase# 
```

Retrieving bare informations

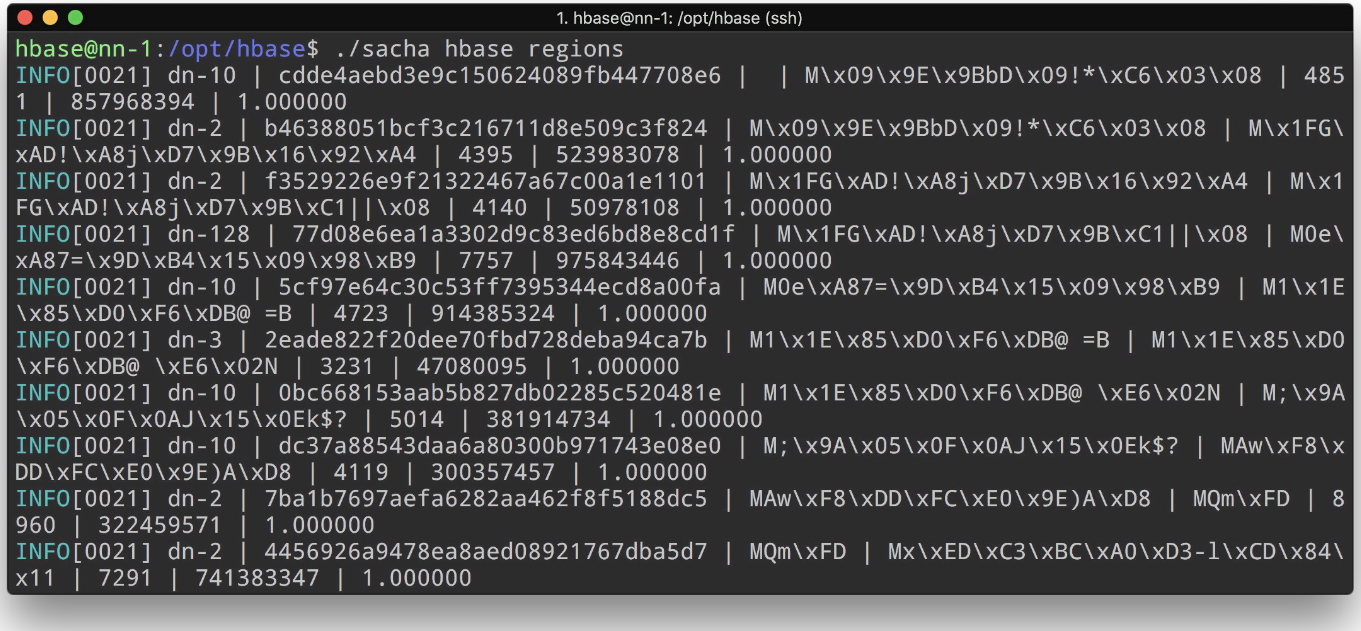


1. hbase@nn-1: /opt/hbase (ssh)

```
hbase@nn-1:/opt/hbase$ ./sacha hbase servers
INFO[0005] dn-85 | dn-85.hadoop.B.GRA.infra.metrics.ovh.net,16020,1536630297124
INFO[0005] dn-117 | dn-117.hadoop.b.gra.infra.metrics.ovh.net,16020,1533841829550
INFO[0005] dn-100 | dn-100.hadoop.B.GRA.infra.metrics.ovh.net,16020,1536630307303
INFO[0005] dn-9 | dn-9.hadoop.b.gra.infra.metrics.ovh.net,16020,1526331102574
INFO[0005] dn-70 | dn-70.hadoop.b.gra.infra.metrics.ovh.net,16020,1532638465829
INFO[0005] dn-115 | dn-115.hadoop.b.gra.infra.metrics.ovh.net,16020,1533841825648
INFO[0005] dn-78 | dn-78.hadoop.b.gra.infra.metrics.ovh.net,16020,1530891364037
INFO[0005] dn-10 | dn-10.hadoop.B.GRA.infra.metrics.ovh.net,16020,1536630281903
INFO[0005] dn-119 | dn-119.hadoop.b.gra.infra.metrics.ovh.net,16020,1535986042437
INFO[0005] dn-91 | dn-91.hadoop.b.gra.infra.metrics.ovh.net,16020,1527788063219
INFO[0005] dn-61 | dn-61.hadoop.b.gra.infra.metrics.ovh.net,16020,1533642514028
INFO[0005] dn-16 | dn-16.hadoop.B.GRA.infra.metrics.ovh.net,16020,1537799642390
INFO[0005] dn-83 | dn-83.hadoop.b.gra.infra.metrics.ovh.net,16020,1532707632810
INFO[0005] dn-96 | dn-96.hadoop.b.gra.infra.metrics.ovh.net,16020,1528715633446
INFO[0005] dn-64 | dn-64.hadoop.b.gra.infra.metrics.ovh.net,16020,1533644687916
INFO[0005] dn-93 | dn-93.hadoop.B.GRA.infra.metrics.ovh.net,16020,1537277470529
INFO[0005] dn-113 | dn-113.hadoop.b.gra.infra.metrics.ovh.net,16020,1533834504553
INFO[0005] dn-28 | dn-28.hadoop.b.gra.infra.metrics.ovh.net,16020,1521767880632
INFO[0005] dn-43 | dn-43.hadoop.B.GRA.infra.metrics.ovh.net,16020,1536747014896
INFO[0005] dn-48 | dn-48.hadoop.b.gra.infra.metrics.ovh.net,16020,1526494308594
INFO[0005] dn-12 | dn-12.hadoop.B.GRA.infra.metrics.ovh.net,16020,1539066910343
INFO[0005] dn-95 | dn-95.hadoop.b.gra.infra.metrics.ovh.net,16020,1530315838140
```

Create region map

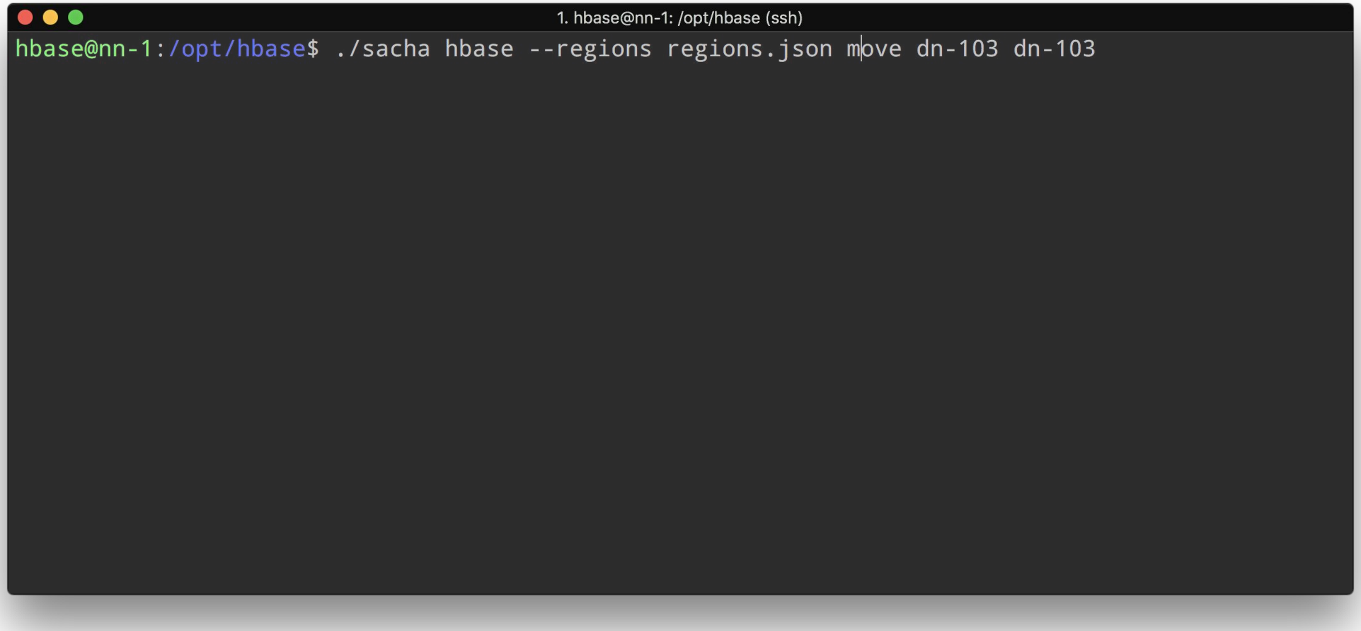
Innovation for Freedom



1. hbase@nn-1: /opt/hbase (ssh)

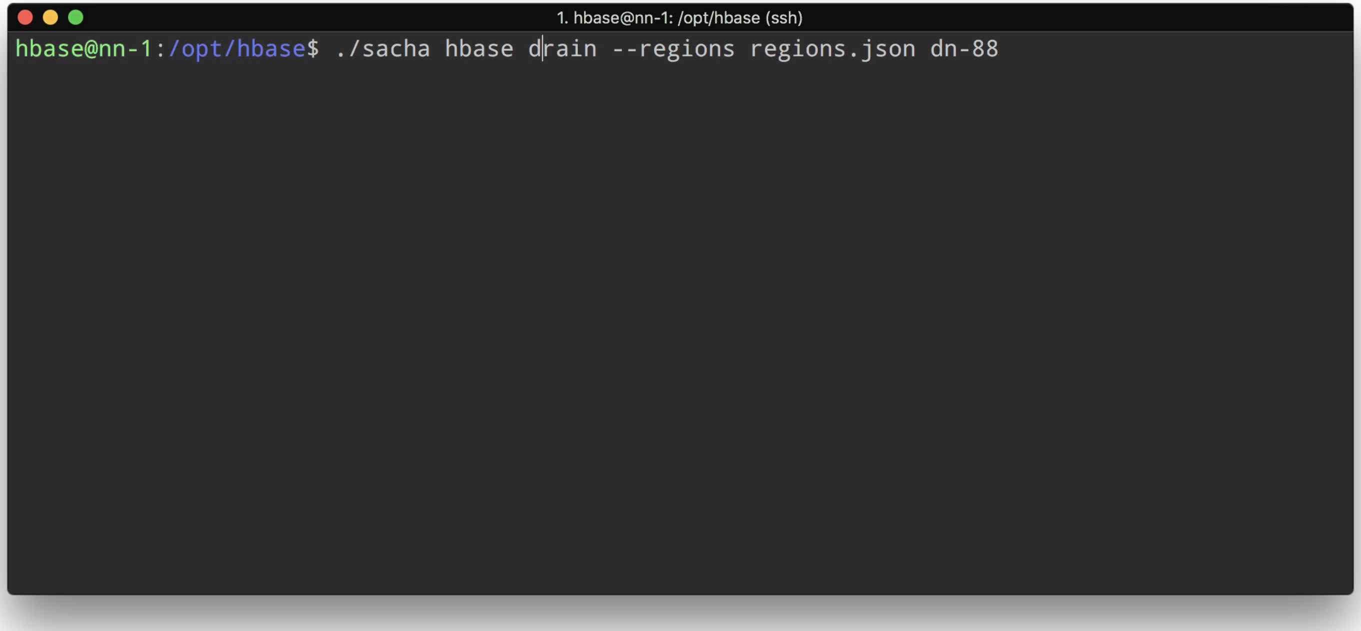
```
hbase@nn-1:/opt/hbase$ ./sacha hbase regions
INFO[0021] dn-10 | cdde4aebd3e9c150624089fb447708e6 |  | M\x09\x9E\x9BbD\x09!*\xC6\x03\x08 | 485
1 | 857968394 | 1.000000
INFO[0021] dn-2 | b46388051bcf3c216711d8e509c3f824 | M\x09\x9E\x9BbD\x09!*\xC6\x03\x08 | M\x1FG\
xAD!\xA8j\xD7\x9B\x16\x92\xA4 | 4395 | 523983078 | 1.000000
INFO[0021] dn-2 | f3529226e9f21322467a67c00a1e1101 | M\x1FG\xAD!\xA8j\xD7\x9B\x16\x92\xA4 | M\x1
FG\xAD!\xA8j\xD7\x9B\xC1||\x08 | 4140 | 50978108 | 1.000000
INFO[0021] dn-128 | 77d08e6ea1a3302d9c83ed6bd8e8cd1f | M\x1FG\xAD!\xA8j\xD7\x9B\xC1||\x08 | M0e\
xA87=\x9D\xB4\x15\x09\x98\xB9 | 7757 | 975843446 | 1.000000
INFO[0021] dn-10 | 5cf97e64c30c53ff7395344ecd8a00fa | M0e\xA87=\x9D\xB4\x15\x09\x98\xB9 | M1\x1E
\x85\xD0\xF6\xDB@ =B | 4723 | 914385324 | 1.000000
INFO[0021] dn-3 | 2eade822f20dee70fbd728deba94ca7b | M1\x1E\x85\xD0\xF6\xDB@ =B | M1\x1E\x85\xD0
\xF6\xDB@ \xE6\x02N | 3231 | 47080095 | 1.000000
INFO[0021] dn-10 | 0bc668153aab5b827db02285c520481e | M1\x1E\x85\xD0\xF6\xDB@ \xE6\x02N | M;\x9A
\x05\x0F\x0AJ\x15\x0Ek$? | 5014 | 381914734 | 1.000000
INFO[0021] dn-10 | dc37a88543daa6a80300b971743e08e0 | M;\x9A\x05\x0F\x0AJ\x15\x0Ek$? | MAw\xF8\x
DD\xFC\xE0\x9E)A\xD8 | 4119 | 300357457 | 1.000000
INFO[0021] dn-2 | 7ba1b7697aefa6282aa462f8f5188dc5 | MAw\xF8\xDD\xFC\xE0\x9E)A\xD8 | MQm\xFD | 8
960 | 322459571 | 1.000000
INFO[0021] dn-2 | 4456926a9478ea8aed08921767dba5d7 | MQm\xFD | Mx\xED\xC3\xBC\xA0\xD3-l\xCD\x84\
x11 | 7291 | 741383347 | 1.000000
```

Move region to another region server



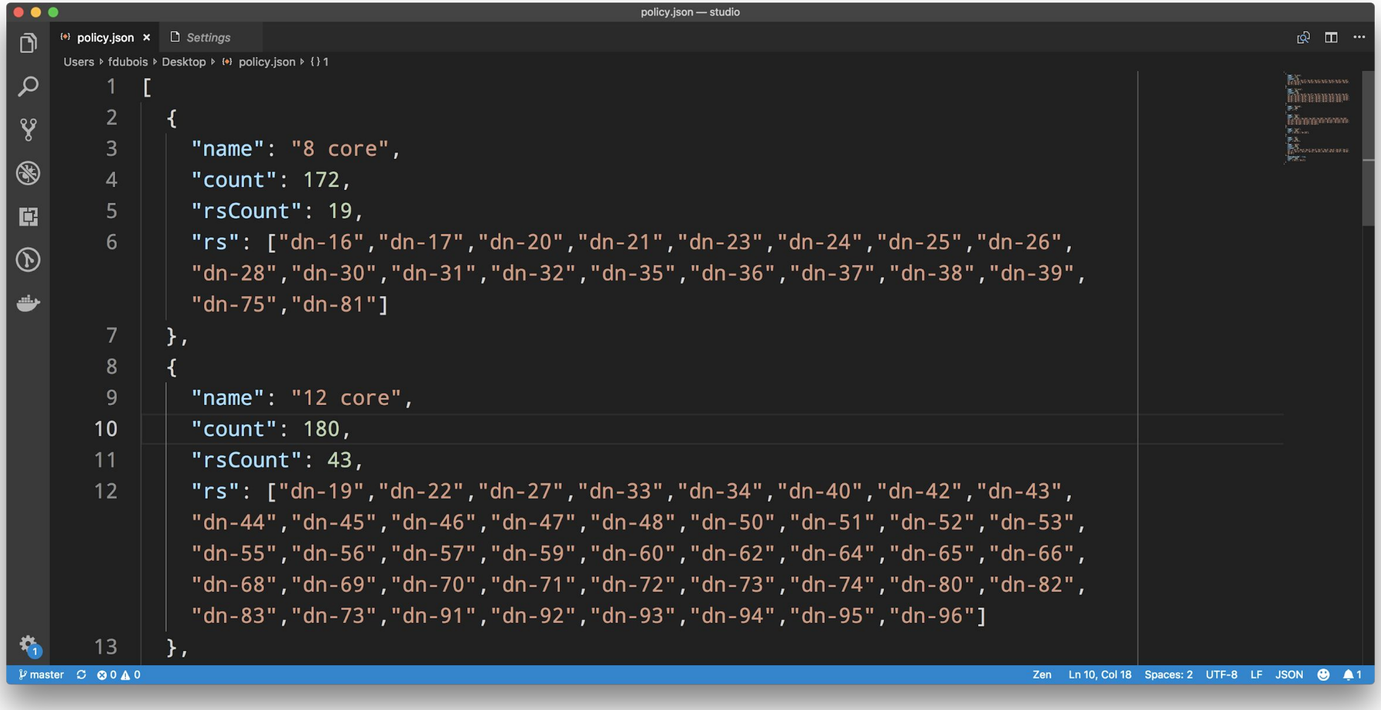
```
hbase@nn-1:/opt/hbase$ ./sacha hbase --regions regions.json move dn-103 dn-103
```

Drain regions of the region server



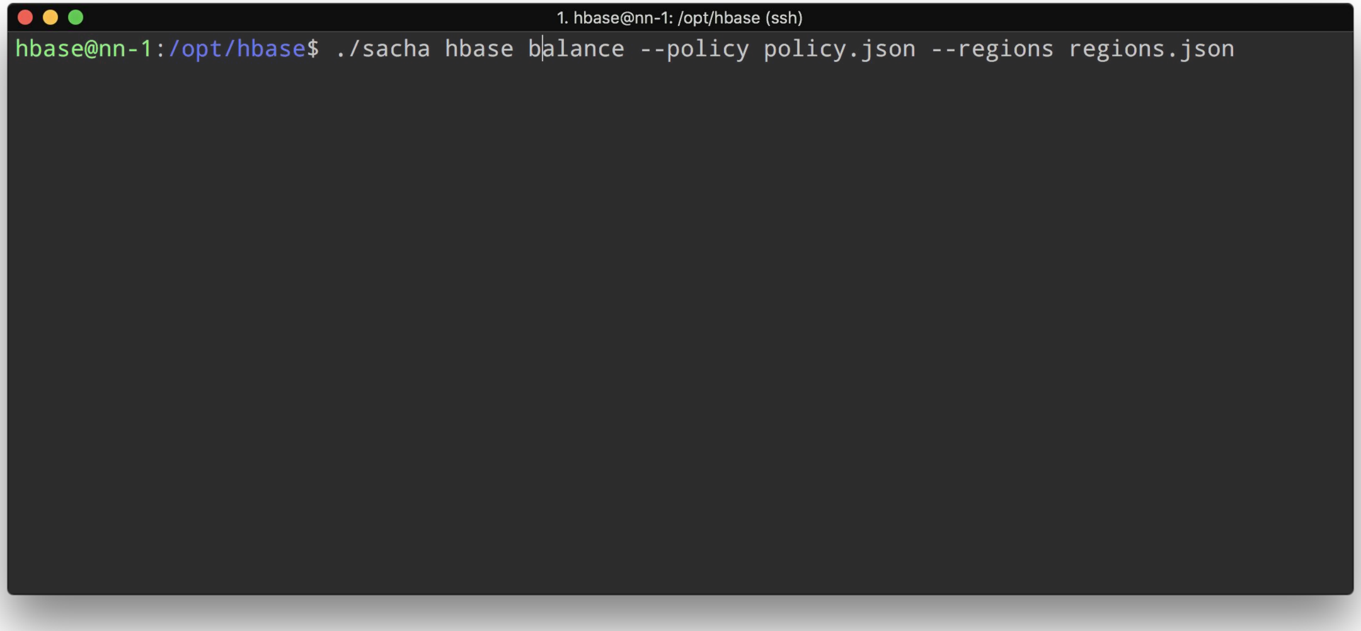
```
hbase@nn-1:/opt/hbase$ ./sacha hbase drain --regions regions.json dn-88
```

Managing multiple hardware profiles



```
[
  {
    "name": "8 core",
    "count": 172,
    "rsCount": 19,
    "rs": ["dn-16","dn-17","dn-20","dn-21","dn-23","dn-24","dn-25","dn-26",
    "dn-28","dn-30","dn-31","dn-32","dn-35","dn-36","dn-37","dn-38","dn-39",
    "dn-75","dn-81"]
  },
  {
    "name": "12 core",
    "count": 180,
    "rsCount": 43,
    "rs": ["dn-19","dn-22","dn-27","dn-33","dn-34","dn-40","dn-42","dn-43",
    "dn-44","dn-45","dn-46","dn-47","dn-48","dn-50","dn-51","dn-52","dn-53",
    "dn-55","dn-56","dn-57","dn-59","dn-60","dn-62","dn-64","dn-65","dn-66",
    "dn-68","dn-69","dn-70","dn-71","dn-72","dn-73","dn-74","dn-80","dn-82",
    "dn-83","dn-73","dn-91","dn-92","dn-93","dn-94","dn-95","dn-96"]
  },
```

Balance the cluster



```
hbase@nn-1:/opt/hbase$ ./sacha hbase balance --policy policy.json --regions regions.json
```

Conclusion

That's all folks!