CERN

CERN Cloud Infrastructure report

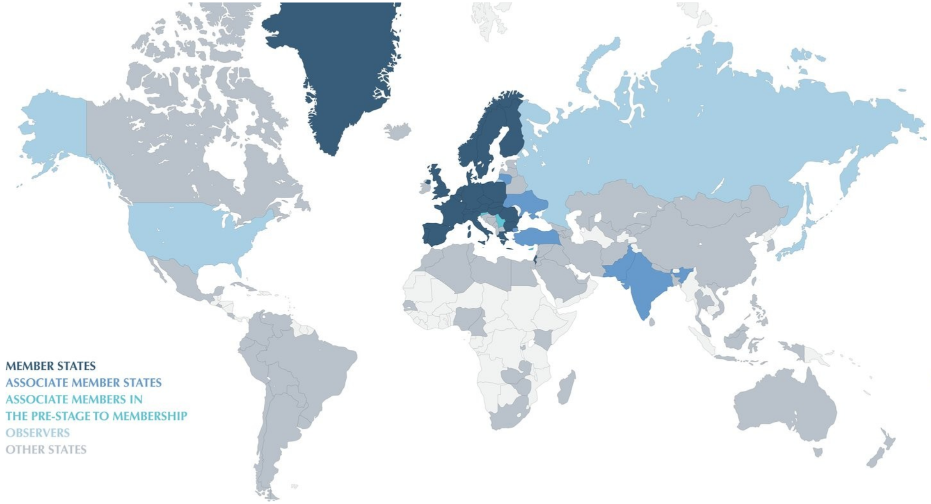
José Castro León
CERN Cloud Infrastructure

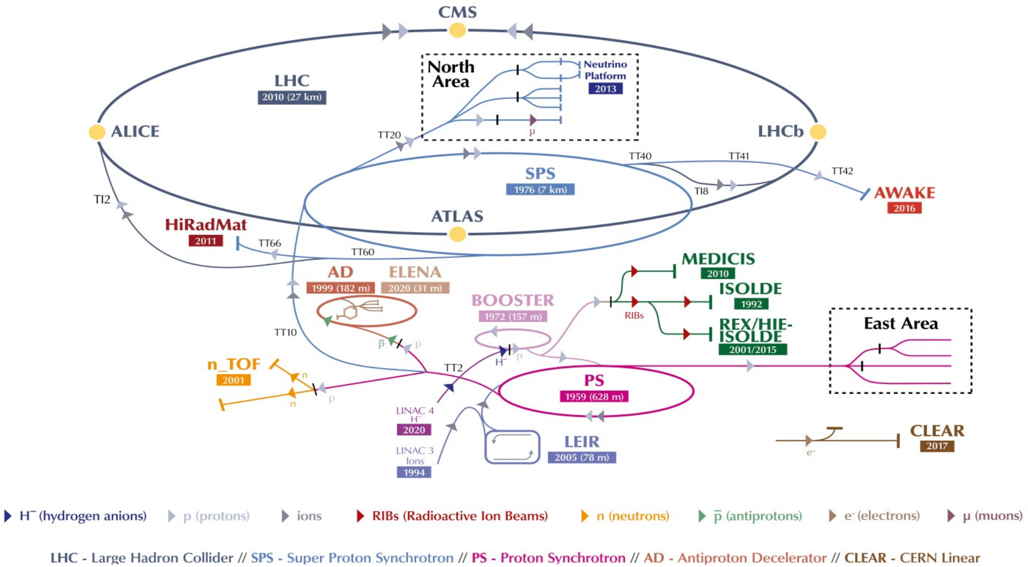
Outline

- Introduction
- CERN Cloud service
  - Deployment
  - Monitoring
  - Accounting
  - Identity
  - Probe and Debug

European Organization for Nuclear Research

- **World largest particle physics laboratory**
- **Founded in 1954**
- **23 member states**
- **Fundamental research in physics**

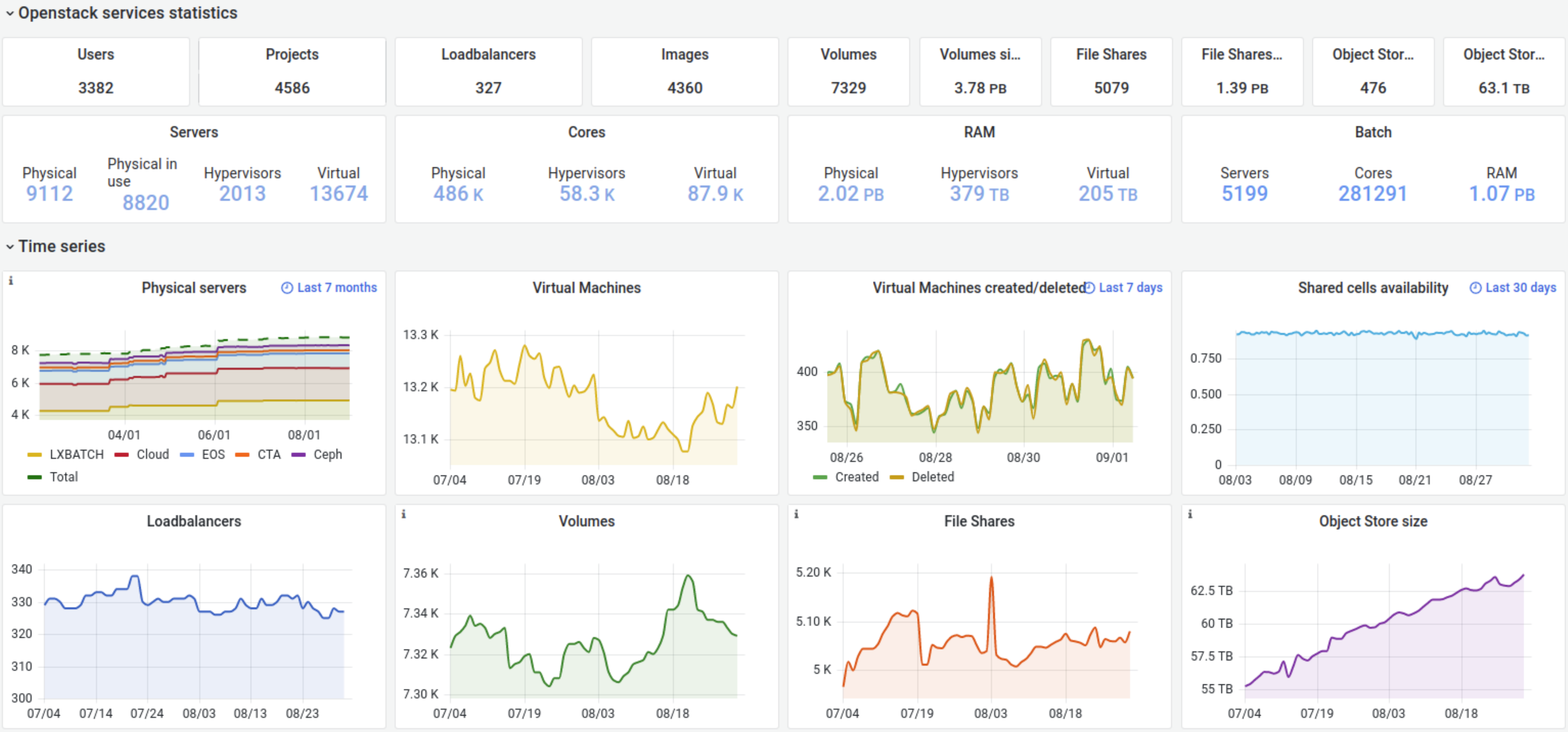




and RUN3 has just started …

CERN Cloud Infrastructure

- Infrastructure as a Service
- Production since **July 2013**
- **CentOS 7** based (adding CentOS Stream 8 soon)
  - Based on RDO
- Geneva Computer centre (adding a new DC)
- Highly **scalable** architecture
  - 48 cells on 5 regions
- Currently running **Stein*** release
  - Some services already in Xena release



Openstack services statistics

Users	Projects	Loadbalancers	Images	Volumes	Volumes si...	File Shares	File Shares...	Object Stor...	Object Stor...
3382	4586	327	4360	7329	3.78 PB	5079	1.39 PB	476	63.1 TB

Servers

Physical	Physical in use	Hypervisors	Virtual
9112	8820	2013	13674

Cores

Physical	Hypervisors	Virtual
486 K	58.3 K	87.9 K

RAM

Physical	Hypervisors	Virtual
2.02 PB	379 TB	205 TB

Batch

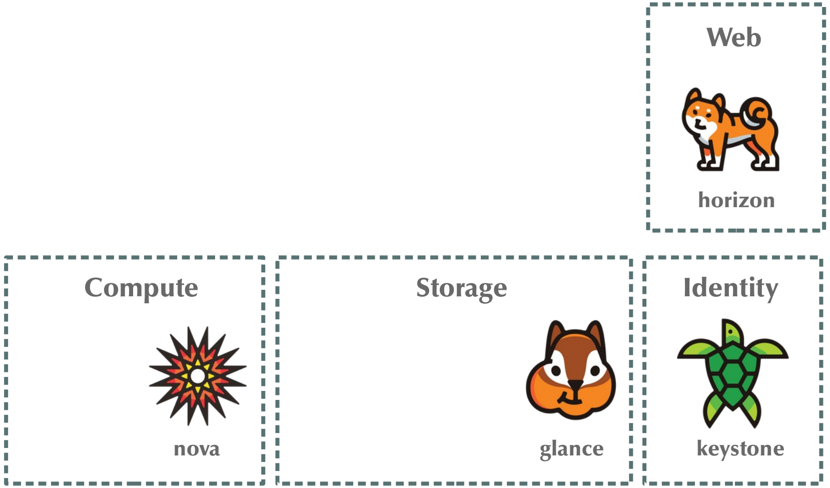
Servers	Cores	RAM
5199	281291	1.07 PB

Time series

Initial offering

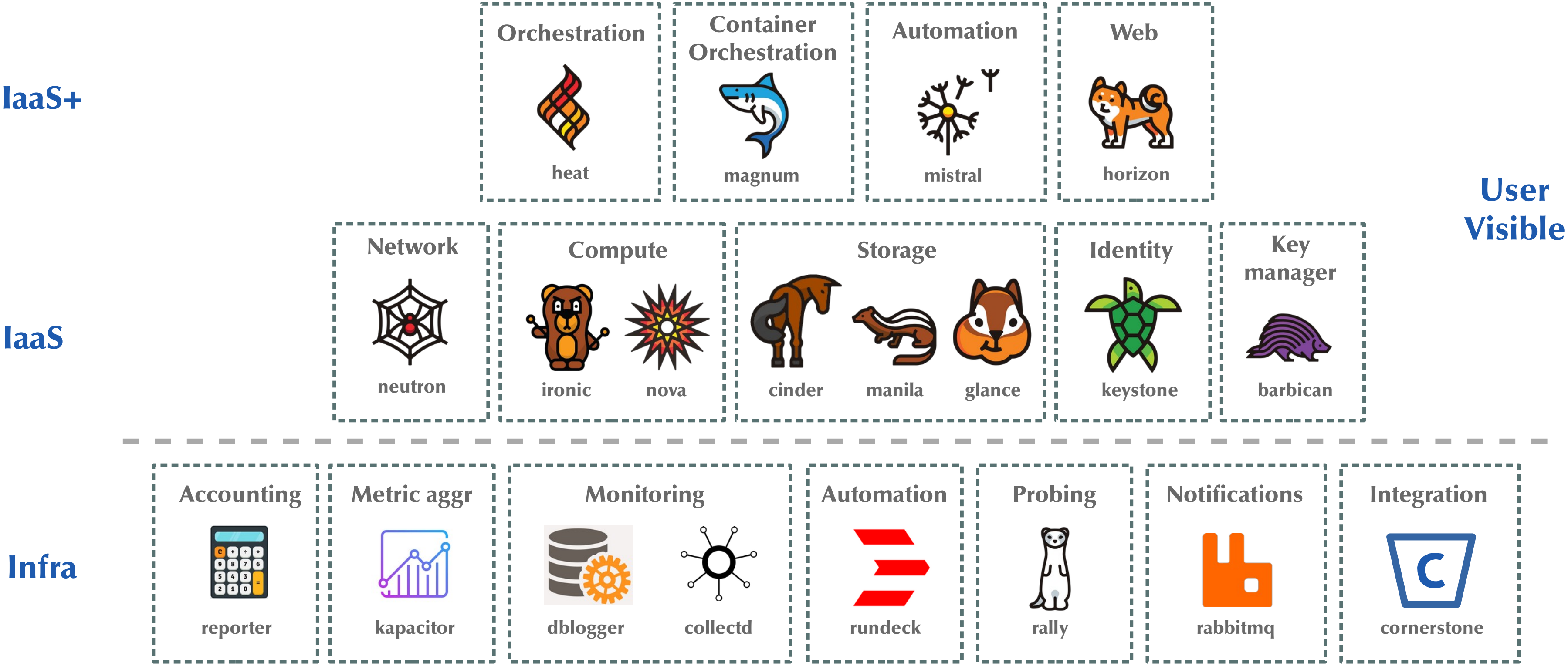
IaaS+

IaaS



User Visible

CERN Cloud Infrastructure - now



IaaS+

Orchestration
heat

Container Orchestration
magnum

Automation
mistral

Web
horizon

User Visible

IaaS

Network
neutron

Compute
ironic
nova

Storage
cinder
manila
glance

Identity
keystone

Key manager
barbican

Infra

Accounting
reporter

Metric aggr
kapacitor

Monitoring
dblogger
collectd

Automation
rundeck

Probing
rally

Notifications
rabbitmq

Integration
cornerstone

Service deployment

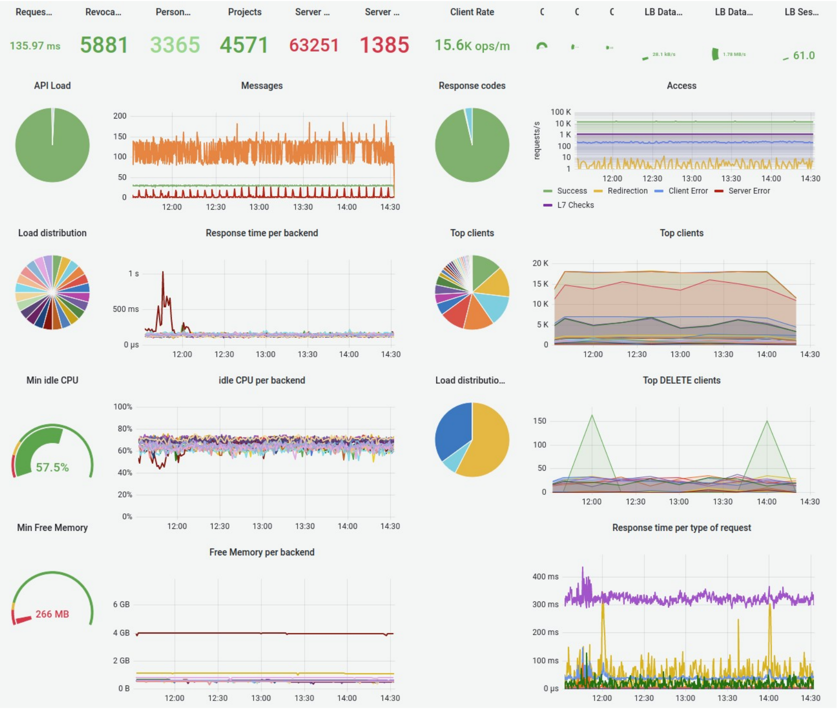
- From shared to "per service" architecture
  - Break dependencies between services
  - Some shared components (rabbitmq, loadbalancers, caches)
  - Freedom to update components under the same API/RPC version
- All deployed in VMs on our own infrastructure: *"eat our own dogfood"*
  - Bootstrap procedure and recovery methods
- Puppet managed running on CentOS 7 (hypervisors) and CentOS Stream 8 (services)

Service operations

- Deployment upgraded since **July 2013**
- Per-service upgrade model (A/B or in place)
- Compute + Storage availability zones (3 zones each)
- Huge investment on **automation**:
  - Delegate as much as possible administrative tasks (repair team, quota mgmt, end-user)
  - Detect and fix known issues
  - User communication
- Quite some big campaigns:
  - KVM consolidation, Spectre/Meltdown and L1TF, Cold Migration

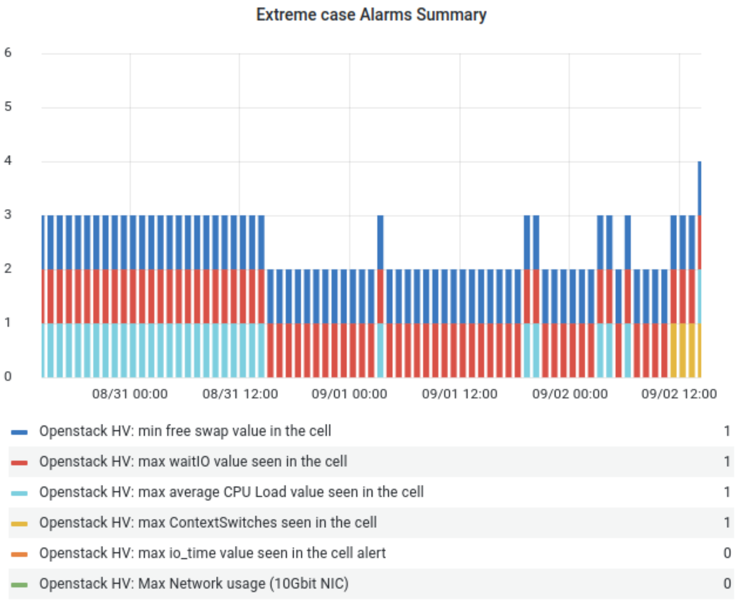
Cloud Monitoring

- Use same monitoring pipeline as any other IT service
  - Metrics (Collectd => InfluxDB)
  - Logs (Flume => Kafka => ES, HDFS)
- Custom sensors for VM monitoring, service metrics
- Threshold based alarming on individual nodes
- Per-service grafana dashboards

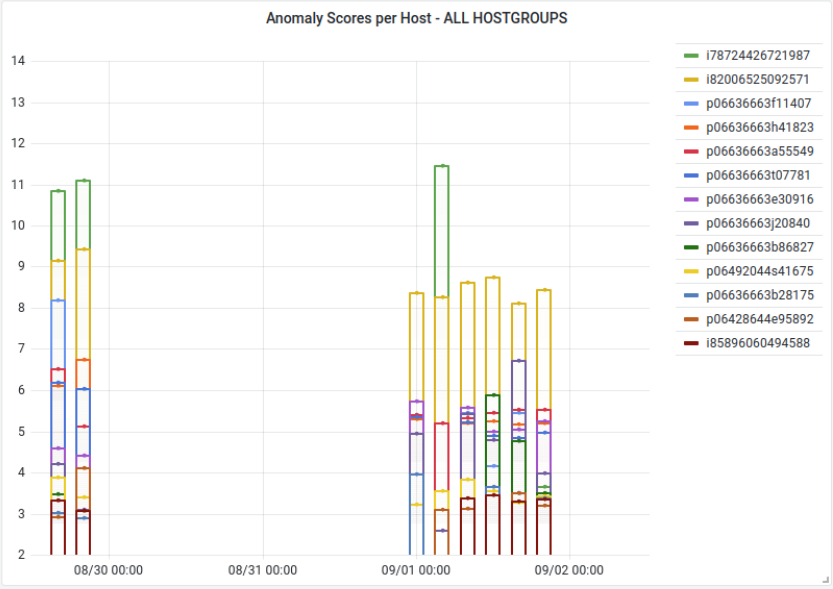


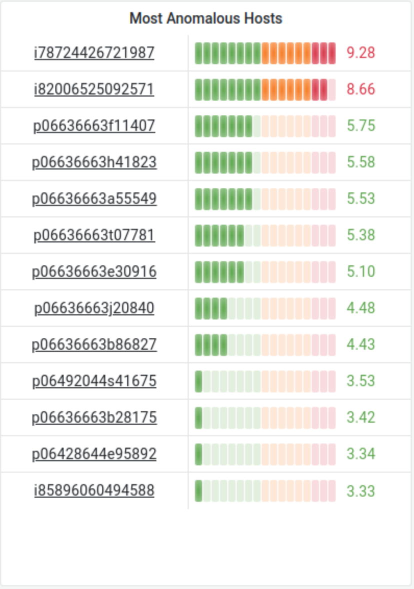
Find the needle in the haystack

- Threshold based alarming on extreme cases
- Anomaly detection to find misbehaving nodes



Openstack HV: min free swap value in the cell	1
Openstack HV: max waitIO value seen in the cell	1
Openstack HV: max average CPU Load value seen in the cell	1
Openstack HV: max ContextSwitches seen in the cell	1
Openstack HV: max io_time value seen in the cell alert	0
Openstack HV: Max Network usage (10Gbit NIC)	0

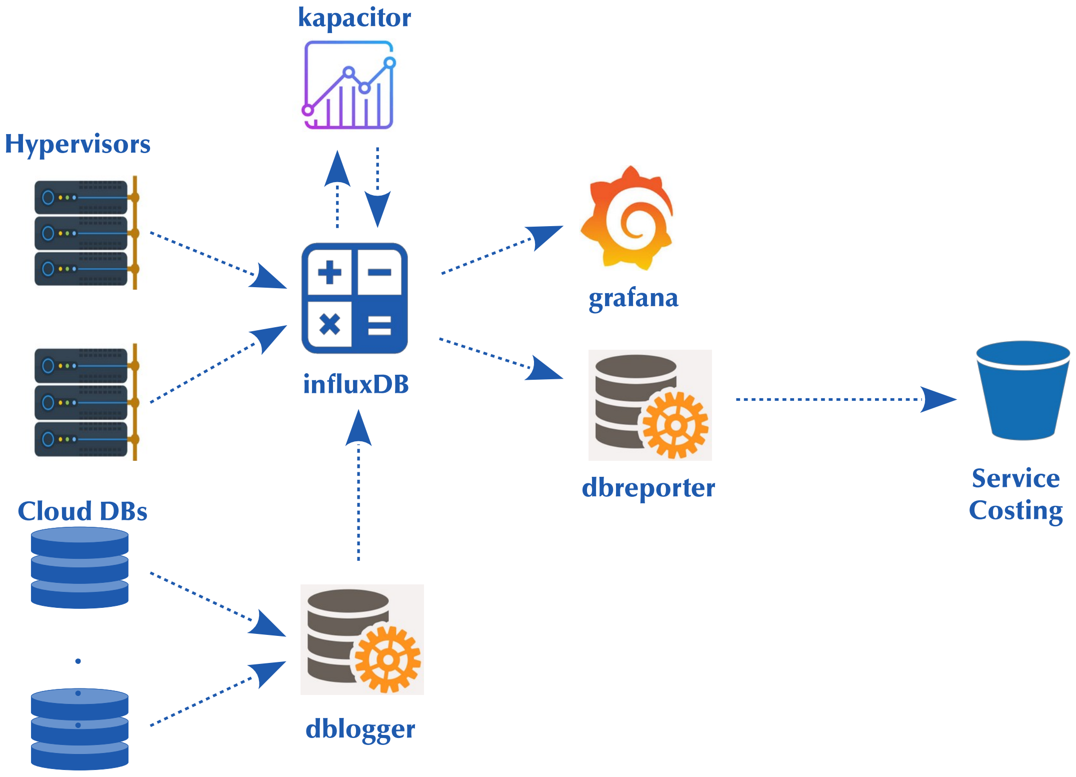



Most Anomalous Hosts

i78724426721987	9.28
i82006525092571	8.66
p06636663f11407	5.75
p06636663h41823	5.58
p06636663a55549	5.53
p06636663t07781	5.38
p06636663e30916	5.10
p06636663j20840	4.48
p06636663b86827	4.43
p06492044s41675	3.53
p06636663b28175	3.42
p06428644e95892	3.34
i85896060494588	3.33

Cloud Accounting

- All resources grouped by project
  - chargegroup & chargerole
- Producers data
  - VM stats sensor in all HVs
  - Metrics from OpenStack DBs
- Stored in InfluxDB
- Aggregated by Kapacitor
- Exported to Service Costing



Identity management

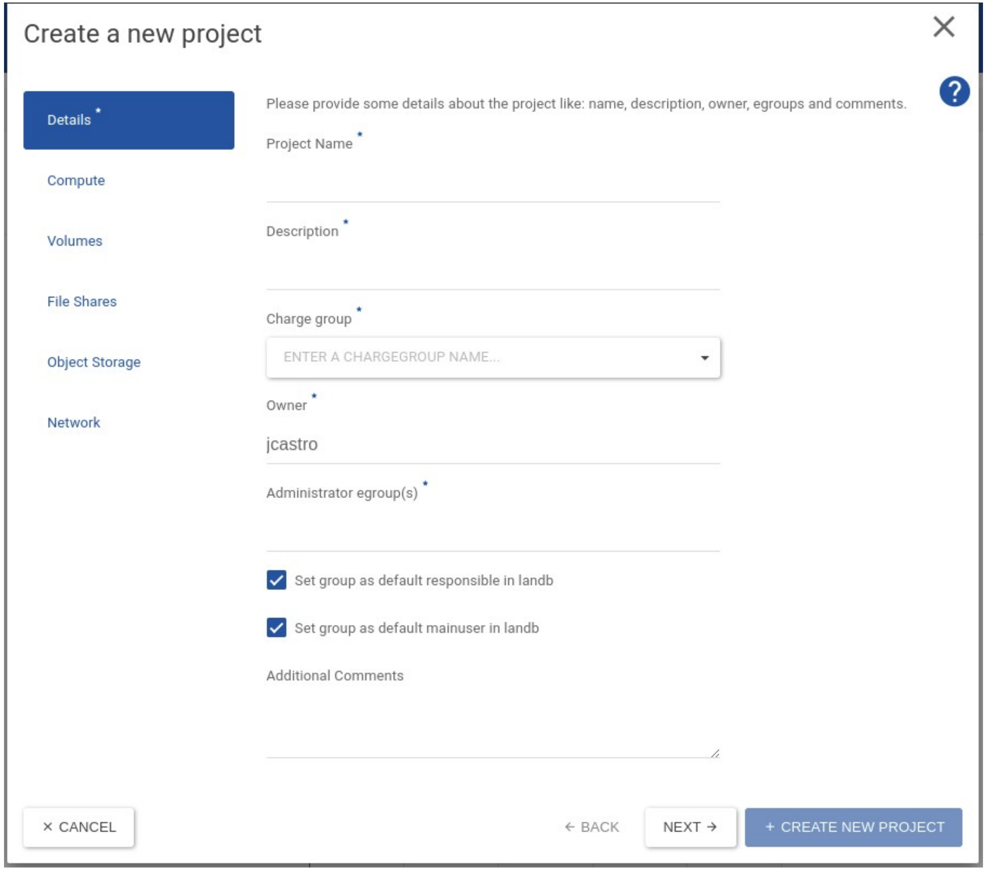
- Available to all CERN Users
  - On-demand provision of resources to federation users (based on group membership)
- Types of projects (owned by a CERN primary account)

	Affiliation Expired	User Disabled	User Deletion
Shared	**Promote**	-	-
Personal	-	**Stop**	**Delete**

- Provisioning and cleanup in Mistral workflows (inter-dependency handling)

Resource management for end user

REQUEST NEW PROJECT



Create a new project

- Details *
- Compute
- Volumes
- File Shares
- Object Storage
- Network

Please provide some details about the project like: name, description, owner, egroups and comments.

Project Name *

Description *

Charge group *

ENTER A CHARGEGROUP NAME...

Owner *

jcastro

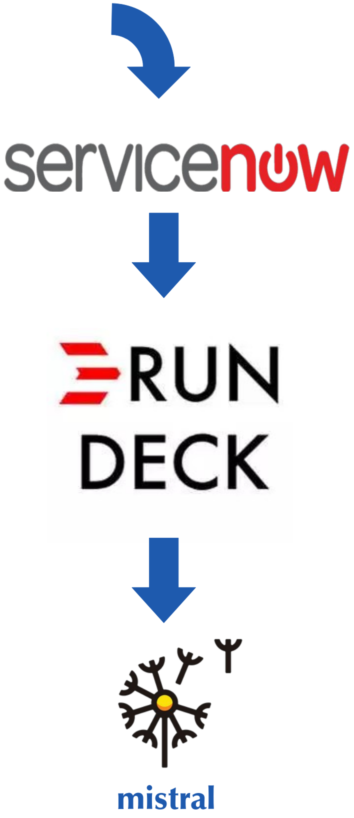
Administrator egroup(s) *

☑ Set group as default responsible in landb

☑ Set group as default mainuser in landb

Additional Comments

✕ CANCEL ← BACK NEXT → + CREATE NEW PROJECT

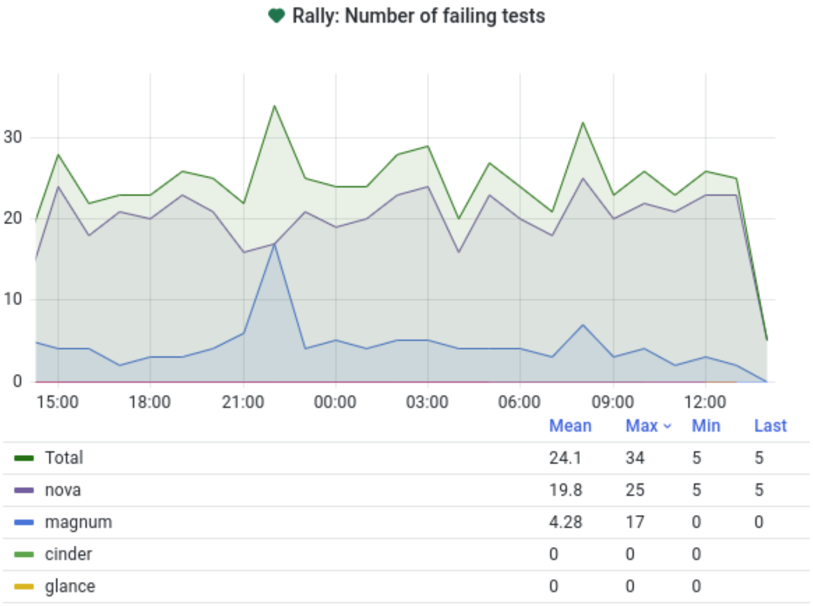


Security approach

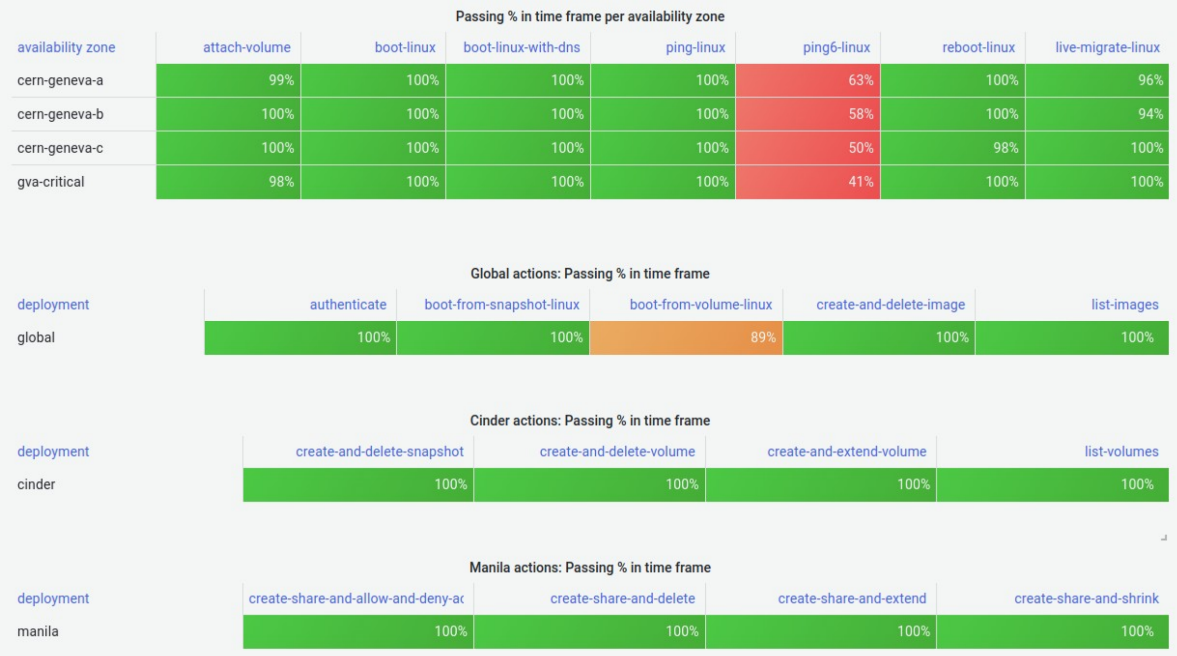
- TLS everywhere (Regular check on TLS security level on endpoints)
- DoS protection on Load Balancers
- 2FA for Administrative operations
- Follow up CVEs on openstack/virtualisation packages with local backports
- Standard vs Audit notifications
- CERN Security team analyses network traffic and controls external firewall
  - Granted additional permissions to stop/lock user VMs
  - Network Isolate any VM/physical node

Cloud Probing

- Use Rally as automated probe system
- Focus on infrastructure wide issues



	Mean	Max	Min	Last
▬ Total	24.1	34	5	5
▬ nova	19.8	25	5	5
▬ magnum	4.28	17	0	0
▬ cinder	0	0	0	
▬ glance	0	0	0	



Passing % in time frame per availability zone

availability zone	attach-volume	boot-linux	boot-linux-with-dns	ping-linux	ping6-linux	reboot-linux	live-migrate-linux
cern-geneva-a	99%	100%	100%	100%	63%	100%	96%
cern-geneva-b	100%	100%	100%	100%	58%	100%	94%
cern-geneva-c	100%	100%	100%	100%	50%	98%	100%
gva-critical	98%	100%	100%	100%	41%	100%	100%

Global actions: Passing % in time frame

deployment	authenticate	boot-from-snapshot-linux	boot-from-volume-linux	create-and-delete-image	list-images
global	100%	100%	89%	100%	100%

Cinder actions: Passing % in time frame

deployment	create-and-delete-snapshot	create-and-delete-volume	create-and-extend-volume	list-volumes
cinder	100%	100%	100%	100%

Manila actions: Passing % in time frame

deployment	create-share-and-allow-and-deny-ac	create-share-and-delete	create-share-and-extend	create-share-and-shrink
manila	100%	100%	100%	100%

Debugging

- Stateless vs Stateful services
  - Focus on reproducing issue
- Use of dedicated “testing” regions
  - Route user requests to extremely verbose setup
  - Connected to other production services
- Probing on Testing Regions
  - Validate minor/major upgrades
  - Introduce feedback with more user scenarios

Thank you

More info:

https://computing-blog.web.cern.ch/

All our **open source** code is available on:

https://gitlab.cern.ch/cloud-infrastructure

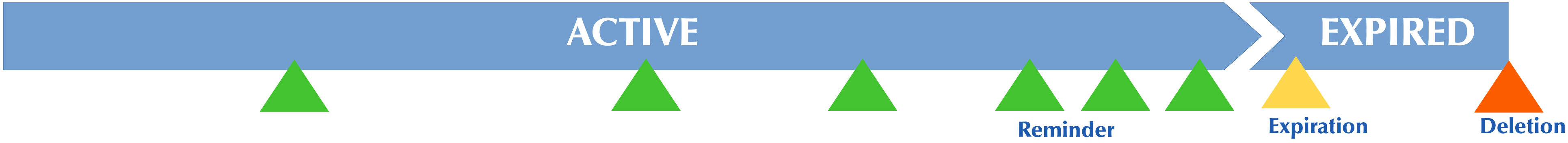
Thank to the work of my team colleagues

CERN

BACKUP SLIDES

Optimize resource availability - Expiration

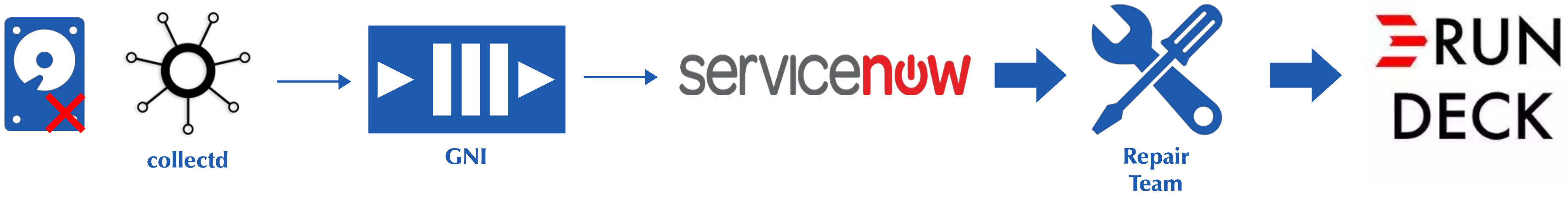
- Each VM in a personal project has an expiration date
- Set shortly after creation and evaluated daily
- Configured to 180 days and renewable
- Reminder mails starting 30 days before expiration
- Implemented on a Workbook in Mistral



Task delegation

- Rely on Rundeck for offloading tasks to different teams
  - Repair Team
  - Resource coordinator
  - Cloud operations

- Example: disk replacement



Why baremetal provisioning?

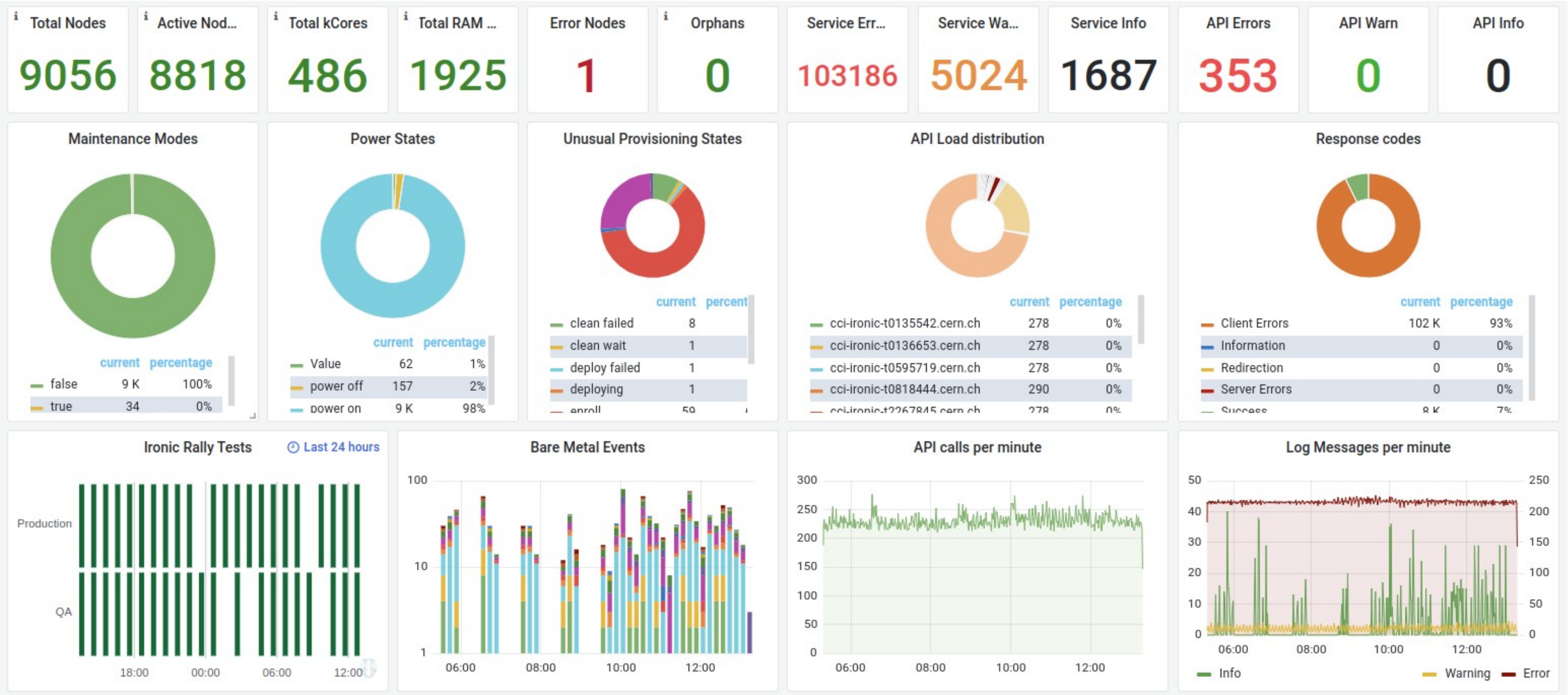
ironic

- VMs not sensible/suitable for all of our use cases
  - Storage nodes, HPC clusters, Batch nodes
- Complete our service offering
  - Physical nodes (in addition to VMs and containers)
  - OpenStack as single pane of glass
- Simplify hardware provisioning workflows
- Consolidate accounting & bookkeeping
  - Machine re-assignments will be easier to track

HW lifecycle at CERN

ironic

Ironic Service setup and status



Total Nodes	Active Nod...	Total kCores	Total RAM ...	Error Nodes	Orphans	Service Err...	Service Wa...	Service Info	API Errors	API Warn	API Info
9056	8818	486	1925	1	0	103186	5024	1687	353	0	0

Maintenance Modes

	current	percentage
false	9 K	100%
true	34	0%

Power States

	current	percentage
Value	62	1%
power off	157	2%
power on	9 K	98%

Unusual Provisioning States

	current	percent
clean failed	8	
clean wait	1	
deploy failed	1	
deploying	1	
enroll	59	

API Load distribution

	current	percentage
cci-ironic-t0135542.cern.ch	278	0%
cci-ironic-t0136653.cern.ch	278	0%
cci-ironic-t0595719.cern.ch	278	0%
cci-ironic-t0818444.cern.ch	290	0%
cci-ironic-t2267845.cern.ch	278	0%

Response codes

	current	percentage
Client Errors	102 K	93%
Information	0	0%
Redirection	0	0%
Server Errors	0	0%
Success	8 K	7%

Ironic Rally Tests

Last 24 hours

Bare Metal Events

API calls per minute

Log Messages per minute

ironic

Users:

- Cloud
- Batch
- HPC
- Windows
- DB
- ...