

The background of the slide is a complex, abstract network diagram. It consists of numerous nodes, represented by small circles, connected by thin lines. Some nodes are highlighted with larger, thicker circles. The overall structure is dense and interconnected, suggesting a global or multi-scale network. The lines and nodes are rendered in shades of gray and black, creating a technical and digital aesthetic.

# Rackspace project Status and Next Steps

Spyridon Trigazis  
Openlab fellow



# CERN & Rackspace collaboration

## ➤ **Started at 2013 working on Cloud Federation**

- Focused on allowing users to access multiple Cloud Infrastructures using a single set of credentials
- Federation allows users to authenticate through the OpenStack Identity service (keystone) using an identity provider like shibboleth or use keystone as an identity provider

## ➤ **Continued on 2016 working integrating container technologies with OpenStack**

- Providing services like Kubernetes, Docker Swarm and Apache Mesos
- Integrate non-openstack components with OpenStack services like the Block Storage service (cinder), Load Balancing service (neutron/lbaas) and Key Manager service (barbican)

# Upstream collaboration

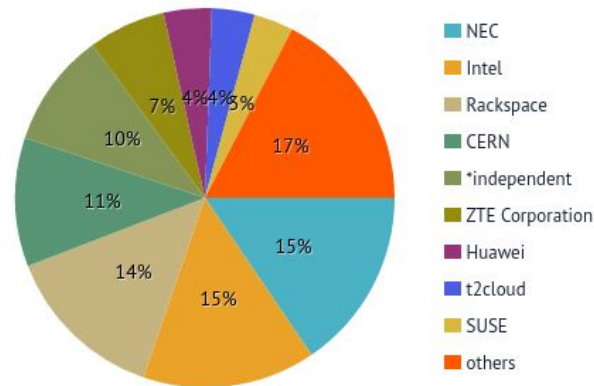
- CERN and Rackspace work closely as members of the openstack-containers team with significant influence and representation in the project
- The Openstack-containers team primarily develops OpenStack/Magnum the Container Infrastructure Management service whose mission is to readily provide all the infrastructure needed to run container clusters on OpenStack
- Secondly, we contribute to related non-openstack projects like Docker, Kubernetes, codedellemc/rexray, gophercloud/gophercloud etc

# Upstream Collaboration

- Apart from CERN and Rackspace, magnum is developed by several companies like Huawei, Intel, IBM, NEC etc
- It's not common for a young project to have such a wide representation

- Development is done openly with the community  
Following the usual 6 month cycle from:
  - Design summit, intermediate milestones, Midcycle design and release

Contribution by companies





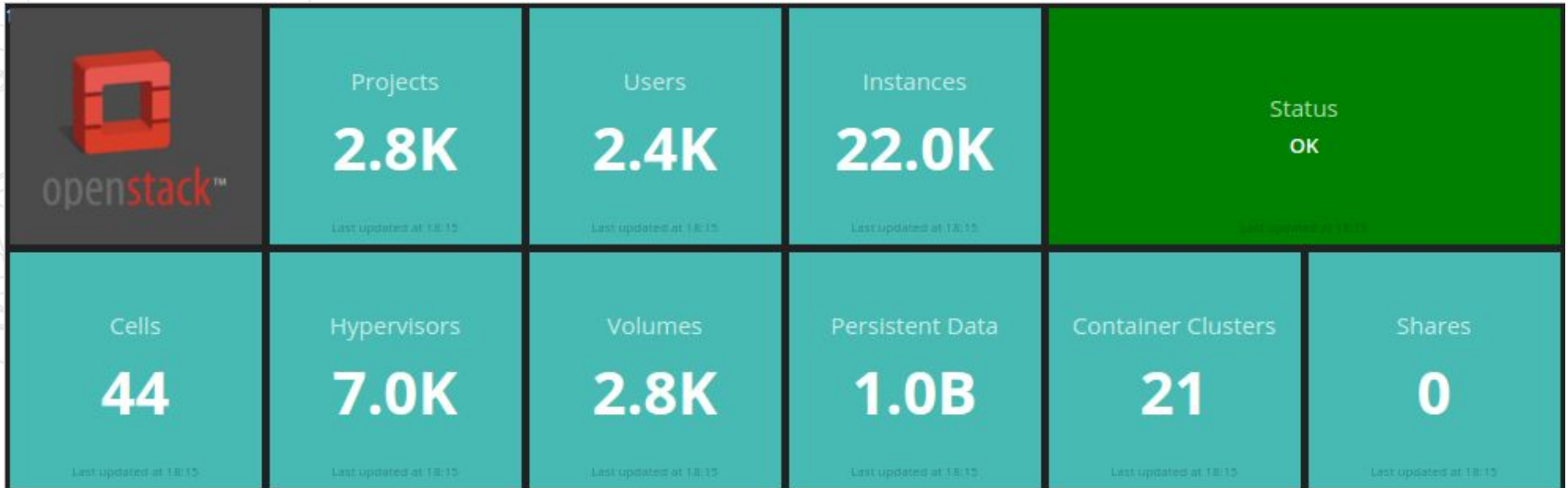
# CERN OpenStack Infrastructure

Production since 2013

~190.000 cores

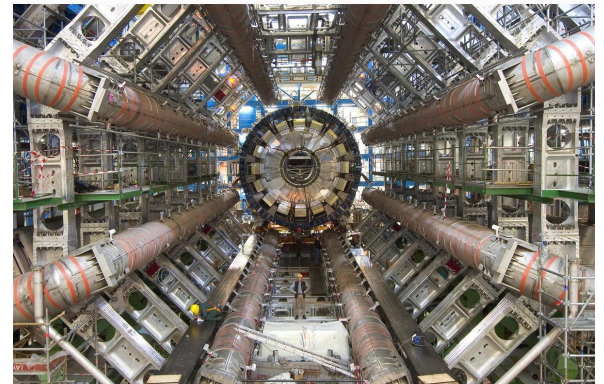
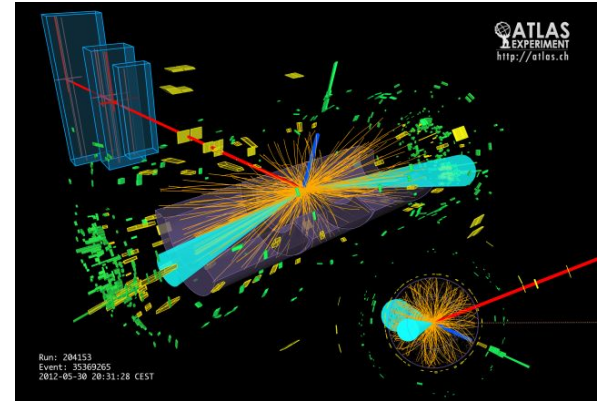
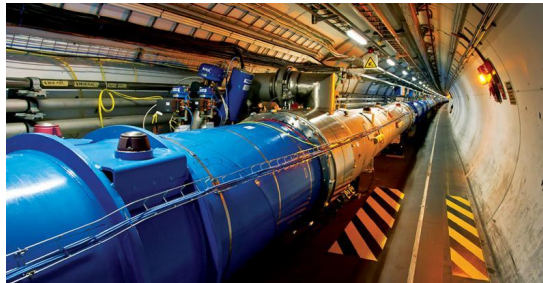
~4million VMs created

~200 VMs created / hour



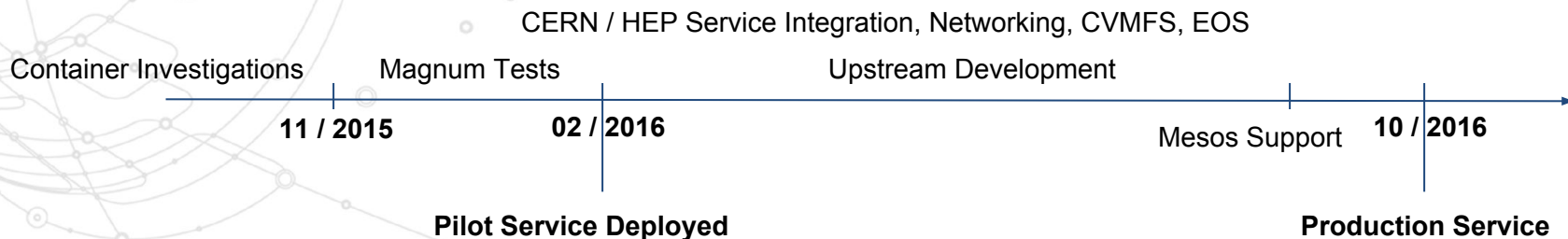
# CERN Container Use Cases

- Batch processing
- End user analysis / Jupyter Notebooks
- Machine Learning / TensorFlow / Keras
- Infrastructure Services
  - Data Movement, Web Servers, PaaS, ...
- Continuous Integration / Deployment
- And many, many others...



# CERN Magnum Deployment

- Integrate containers in the CERN cloud
  - Shared identity, networking integration, storage access, ...
- Agnostic to container orchestration engines
  - Docker Swarm, Kubernetes, Mesos
- Fast, Easy to use



# CERN Magnum Deployment

- Clusters are described by *cluster templates*
- Shared/public templates for most common setups, customizable by users

```
$ magnum cluster-create --name myswarmcluster --cluster-template swarm --node-count 100

$ magnum cluster-list
+-----+-----+-----+-----+-----+
| uuid | name           | node_count | master_count | status           |
+-----+-----+-----+-----+-----+
| .... | myswarmcluster | 100         | 1             | CREATE_COMPLETE |
+-----+-----+-----+-----+-----+

$ $(magnum cluster-config myswarmcluster --dir magnum/myswarmcluster)

$ docker info / ps / ...
$ docker run --volume-driver cvmfs -v atlas.cern.ch:/cvmfs/atlas -it centos /bin/bash
[root@32f4cf39128d /]#
```



# Current Contributions

- Installation documentation and distro package co-ordination
- Support for cluster drivers
- Rally benchmarks and scenarios
- Support for different storage configuration
- Support for kubernetes on baremetal compute instances
- Improvements and maintenance of the upstream CI infrastructure
- Refine CI to build distro custom images with Docker, Kubernetes and per user specific requirements

Contribution summary: *16178* lines of code (across projects, including documentation), *370* reviews since March 2016

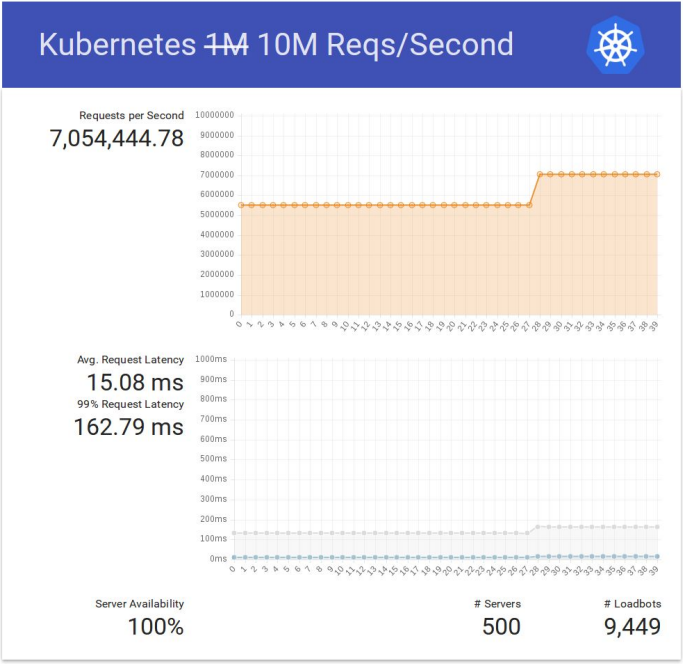
# Scalability Benchmarks

- Is our project production ready? How to identify?
- Benchmark the service itself and benchmark the resources created by the service
  - Scenarios to create and clean-up container clusters (scenarios to create containers are under development) using Openstack/Rally
  - Benchmark the container clusters using a benchmark published by the Kubernetes community
- Repeat both exercises in the CNCF (Cloud Native Computing Foundation) cluster

# CERN Results

- Second go: rally and 7 million requests / sec
  - Kubernetes **7 million requests / sec**
  - 1000 node clusters (4000 cores, 8000 GB / RAM)

Cluster Size (Nodes)	Concurrency	Deployment Time (min)
2	50	2.5
16	10	4
32	10	4
128	5	5.5
512	1	14
1000	1	23

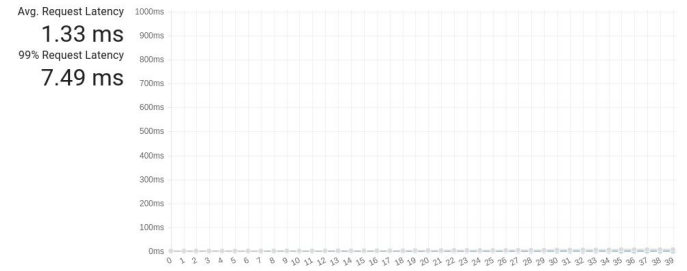
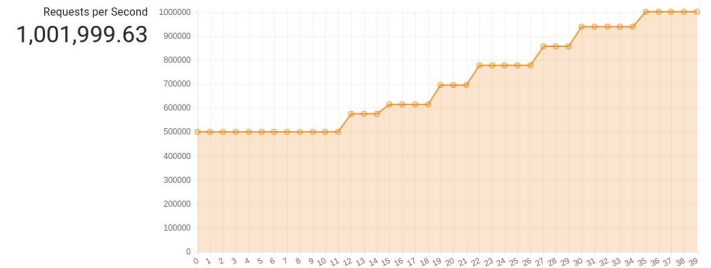


# CNCF Results

COE	Nodes	Concurrency	Containers	Time (sec)
K8S	2	4	8	2.3
Swarm	2	4	8	6.2
Mesos	2	4	8	122.0

Nodes	Concurrency	Clusters	Time (min)
2	10	100	3.02
2	10	1000	Able to create 219 clusters
32	5	100	Able to create 28 clusters
512	1	1	*
4000	1	1	*

## Kubernetes 1M Reqs/Second



Server Availability 100% # Servers 100 # Loadbots 1,000

# Further investigations

- Support for cluster upgrades
  - Upgrade underlying container engine (currently only Docker)
  - Upgrade container orchestration engine (Kubernetes, Docker Swarm, Mesos/Marathon) with zero downtime
- Improve cluster driver management
- Investigate alternative backends instead of Openstack/Heat, like Openstack/Senlin or Openstack/Ansible
- Support for heterogeneous clusters, different hardware configurations (ssd, non-ssd or GPU, non-GPU)

# Questions?