

openstack®

FROM THE CLOUDS TO THE EDGE

RETOUR D'EXPÉRIENCE SUR LES
CAMPAGNES D'EXPÉRIMENTATIONS
CONDUITES DANS LE CADRE DE IPL
DISCOVERY.

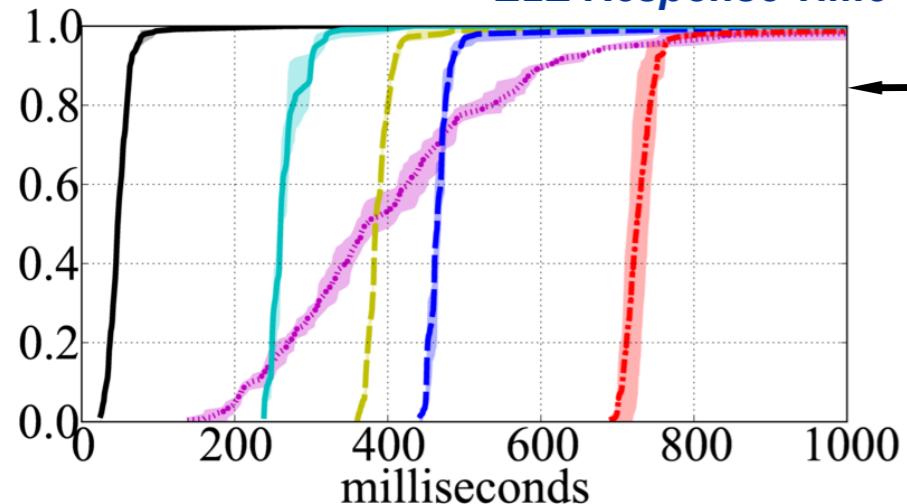
Orange Labs



ADRIEN LEBRE, TILECS, JULY 2019

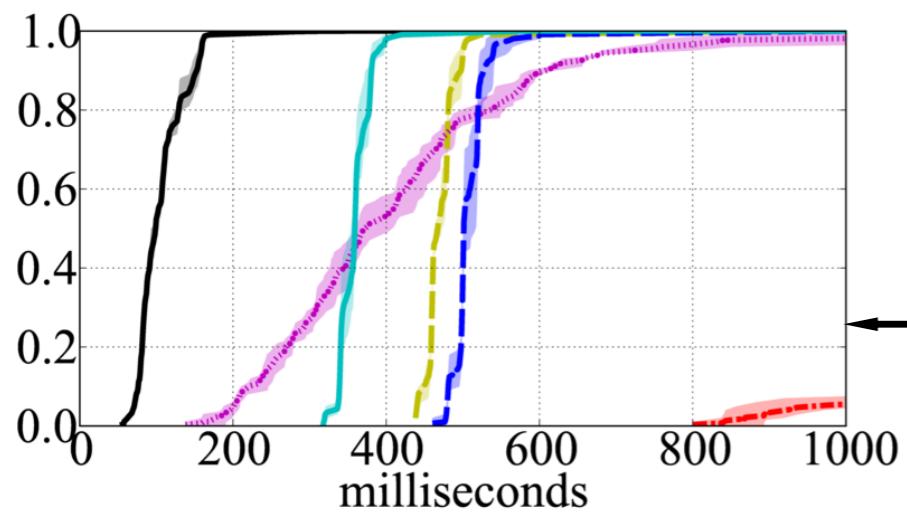
Augmented Reality

E2E Response Time CDF



Wi-Fi
802.11n

- Mobile-only
- Amazon East
- - - Amazon West
- - - Amazon EU
- - - Amazon Asia
- Cloudlet



4G LTE
T-Mobile for Cloud
In-lab Nokia eNodeB for Clouddlet



"The Impact of Mobile Multimedia Applications on Data Center Consolidation"

Ha, K., Pillai, P., Lewis, G., Simanta, S., Clinch, S., Davies, N., Satyanarayanan, M.
Proceedings of IEEE International Conference on Cloud Engineering (IC2E), San Francisco, CA, March 2013

"Quantifying the Impact of Edge Computing on Mobile Applications"

Hu, W., Gao, Y., Ha, K., Wang, J., Amos, B., Pillai, P., Satyanarayanan, M.
Proceedings of ACM APSys 2016, Hong Kong, China, August 2016

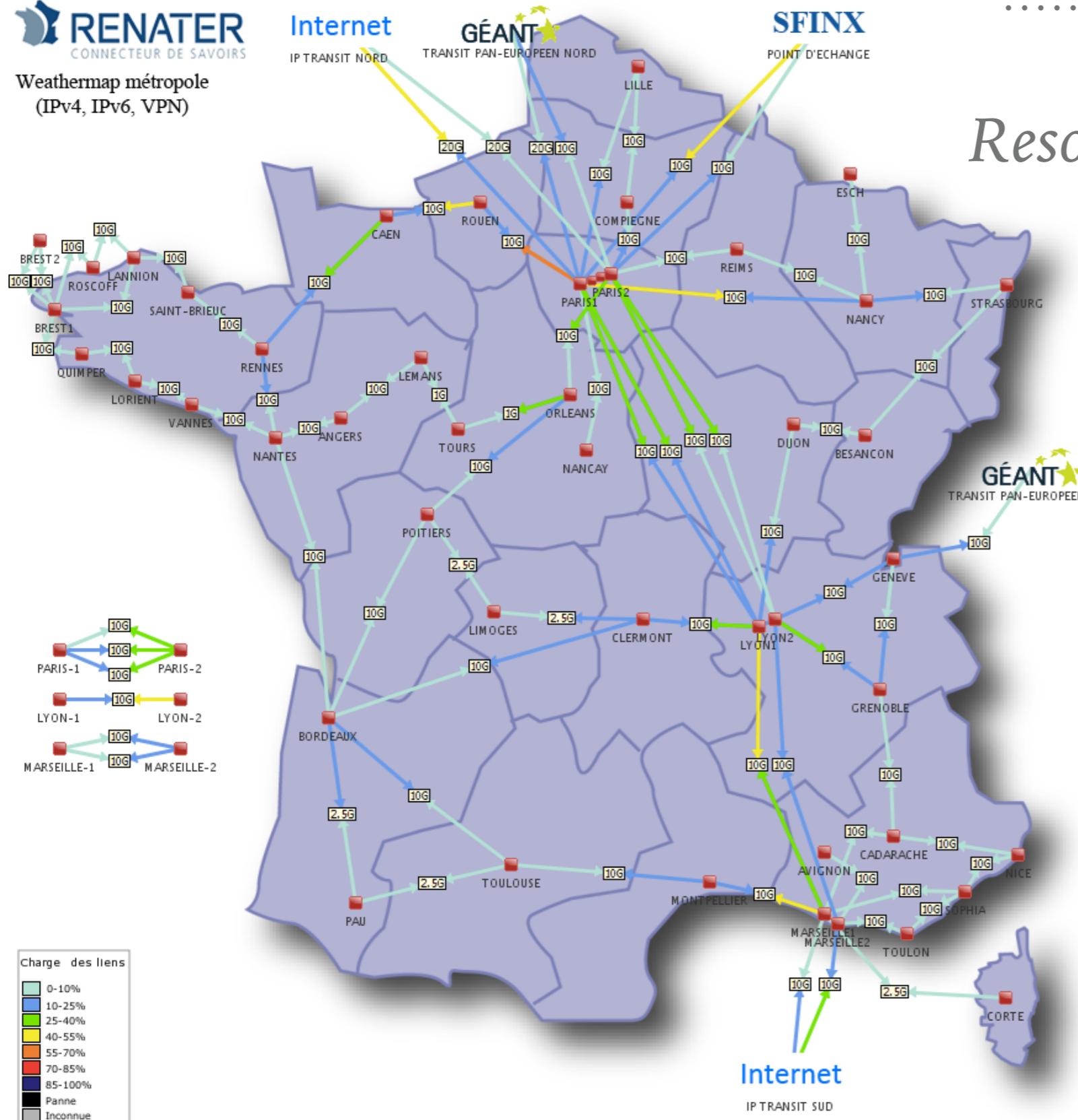
Why the edge?
Mahadev Satyanarayanan

Many papers/blogs that point out the relevance of Edge Computing (latency, data privacy, energy, ...)

DISCOVERY



Weathermap métropole
(IPv4, IPv6, VPN)



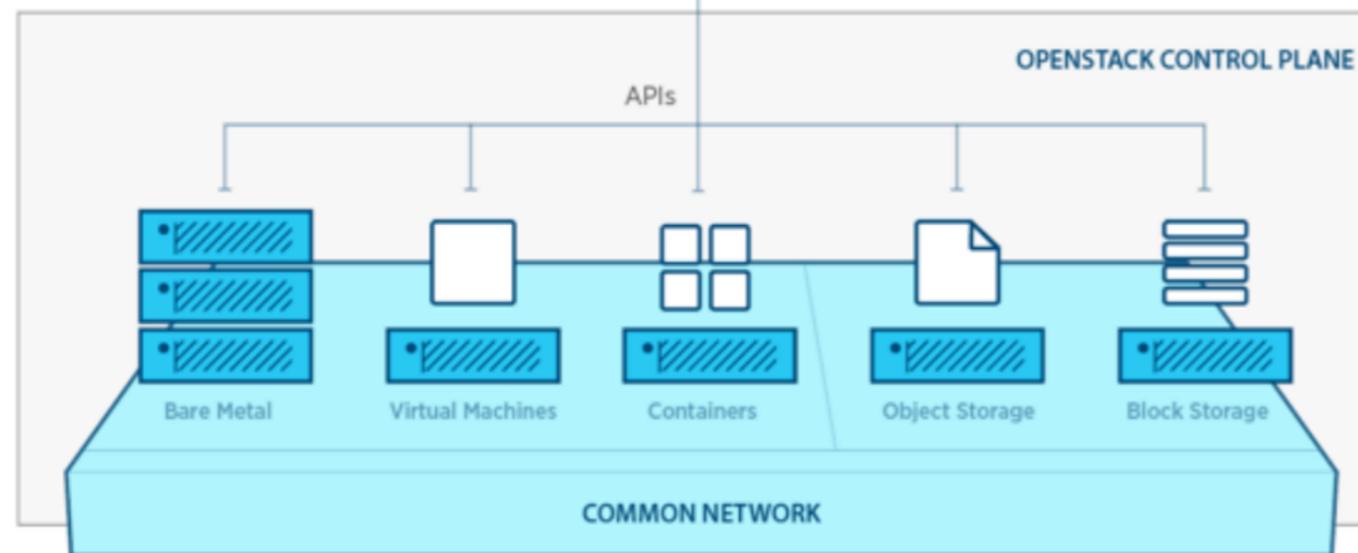
*Edge Computing
Resource Management System:
a Critical Building Block!
HotEdge2018*

*Designing/implementing a RMS
for such an infrastructure?*

INITIATING THE DEBATE WITH OPENSTACK

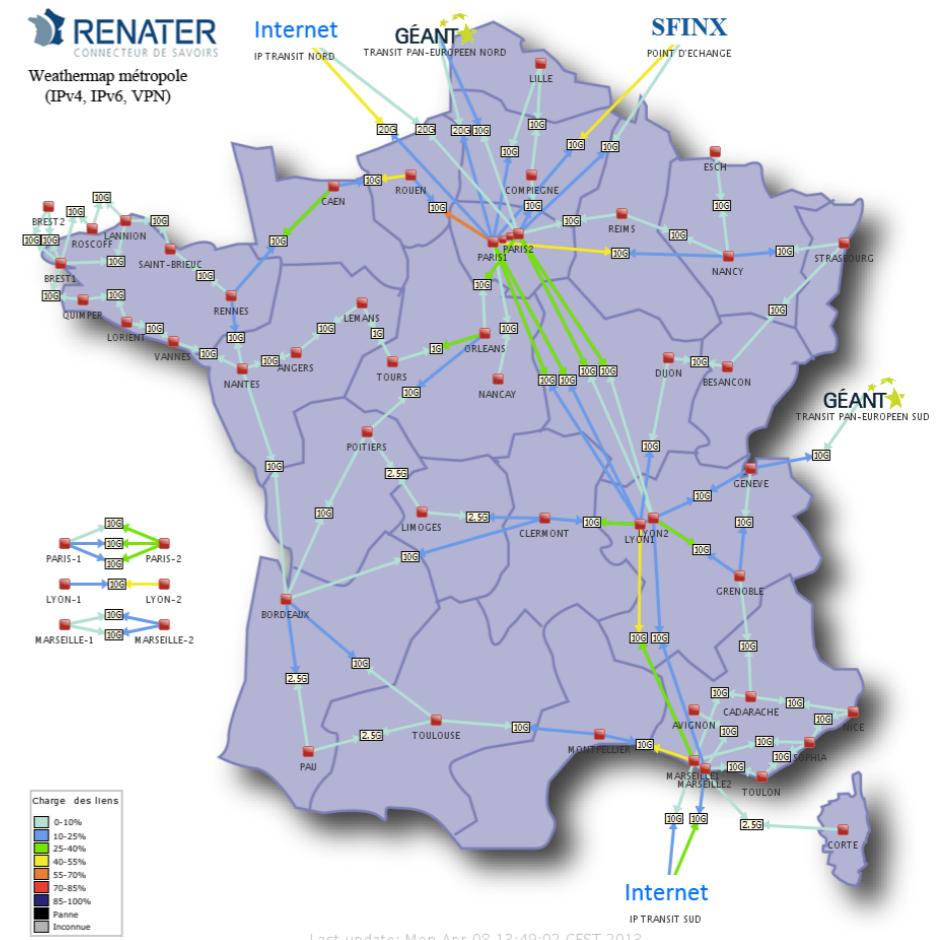


YOUR APPLICATIONS



*Scalability?
Latency/throughput impact?
Network partitioning issues?*

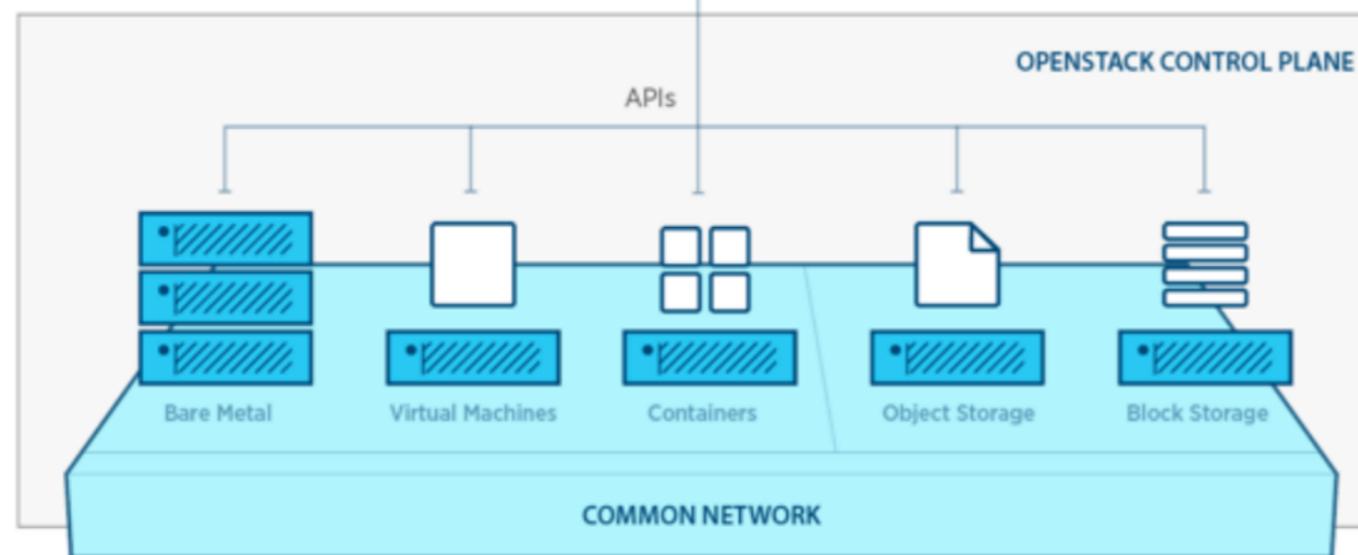
...



INITIATING THE DEBATE WITH OPENSTACK

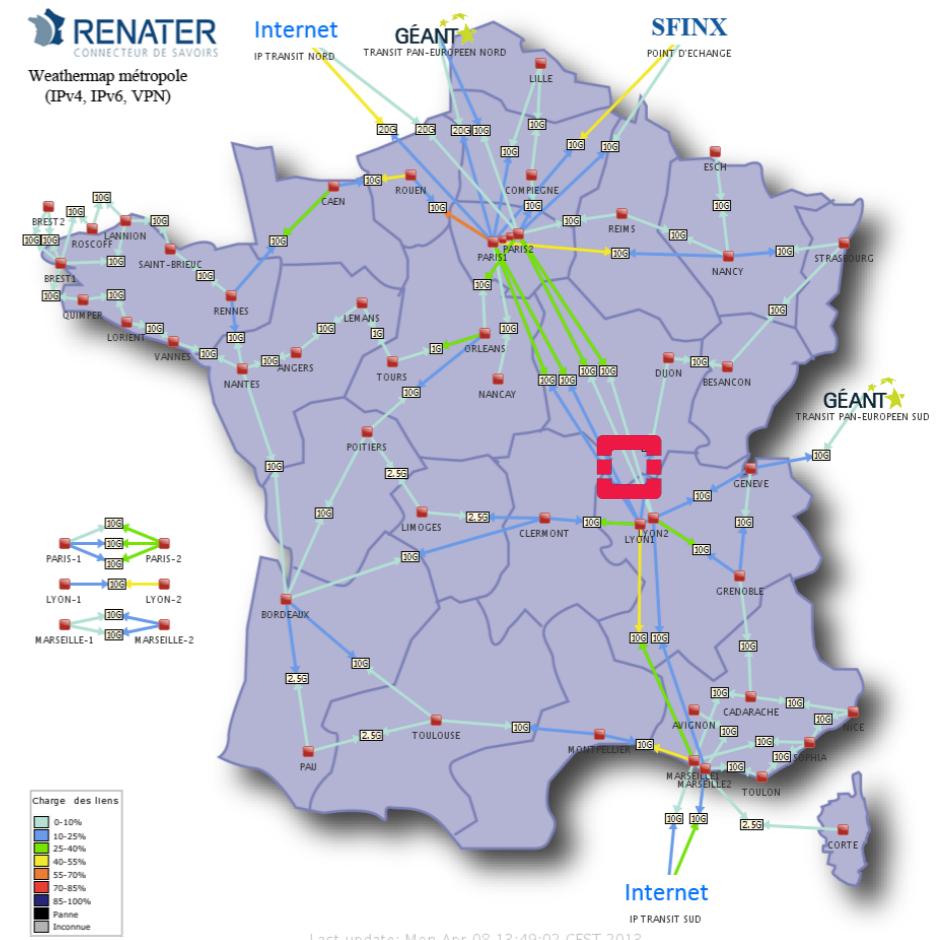


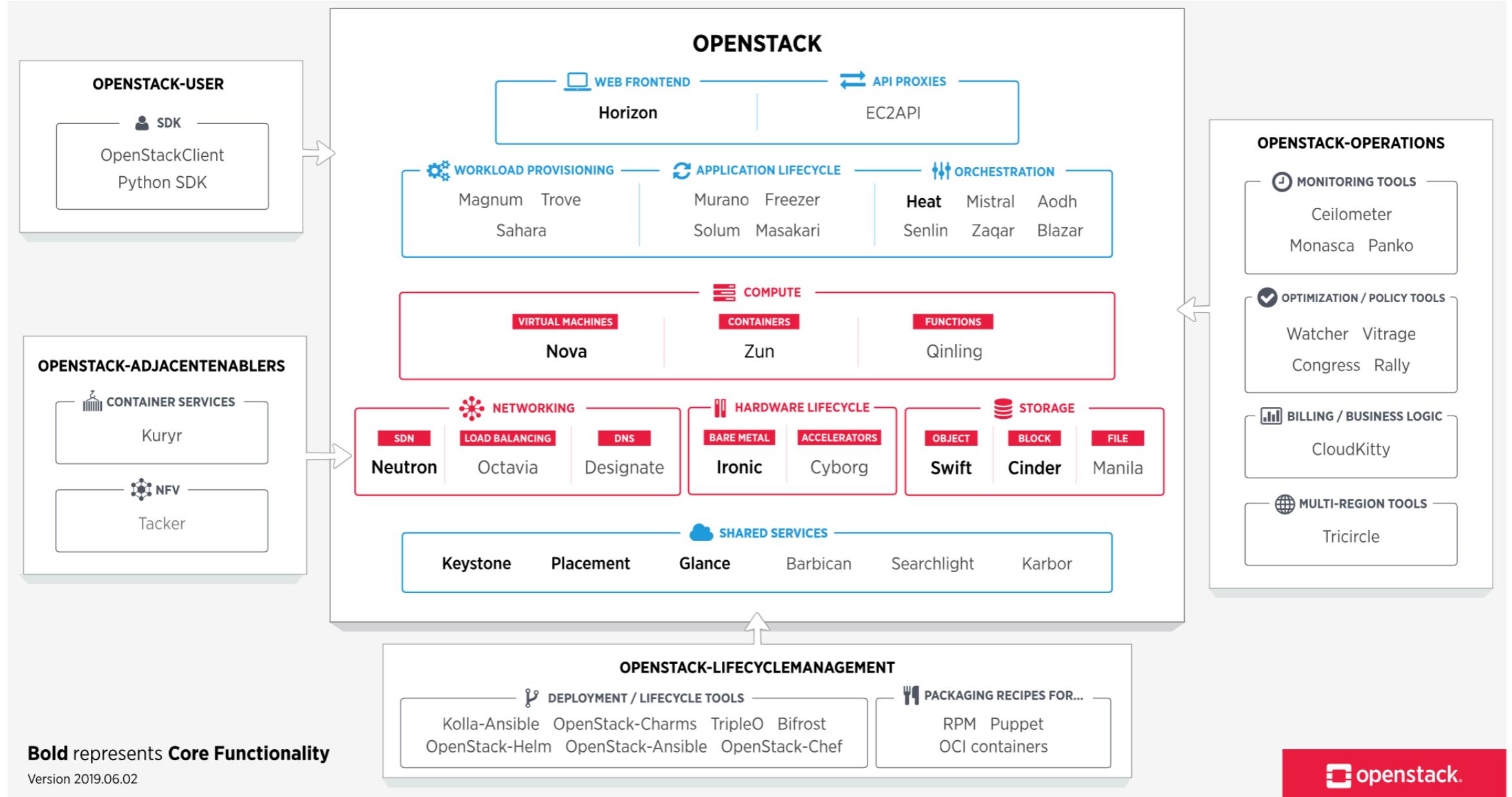
YOUR APPLICATIONS



Scalability?
Latency/throughput impact?
Network partitioning issues?

...





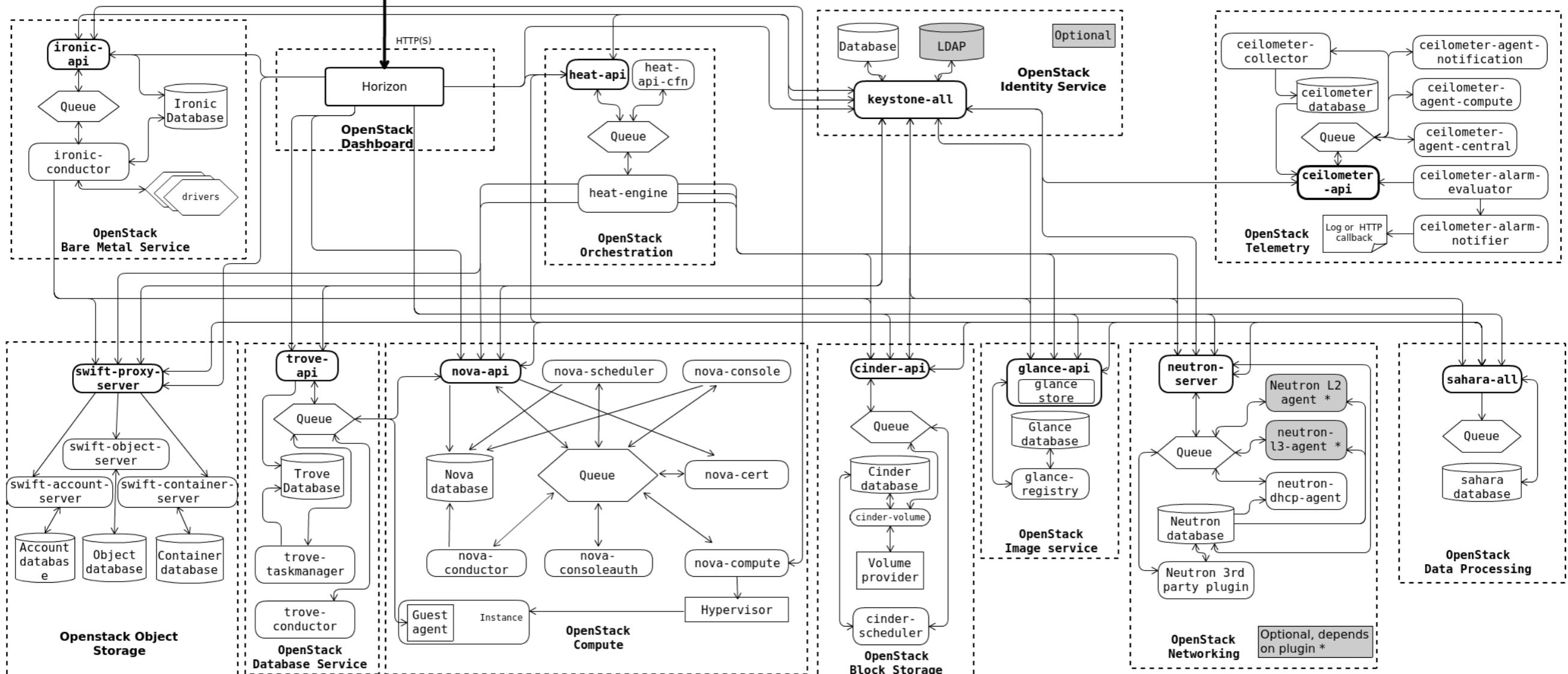
OPENSTACK (THE DEVIL IN DETAILS)

13 Millions of LOCs
186 subservices

Internet

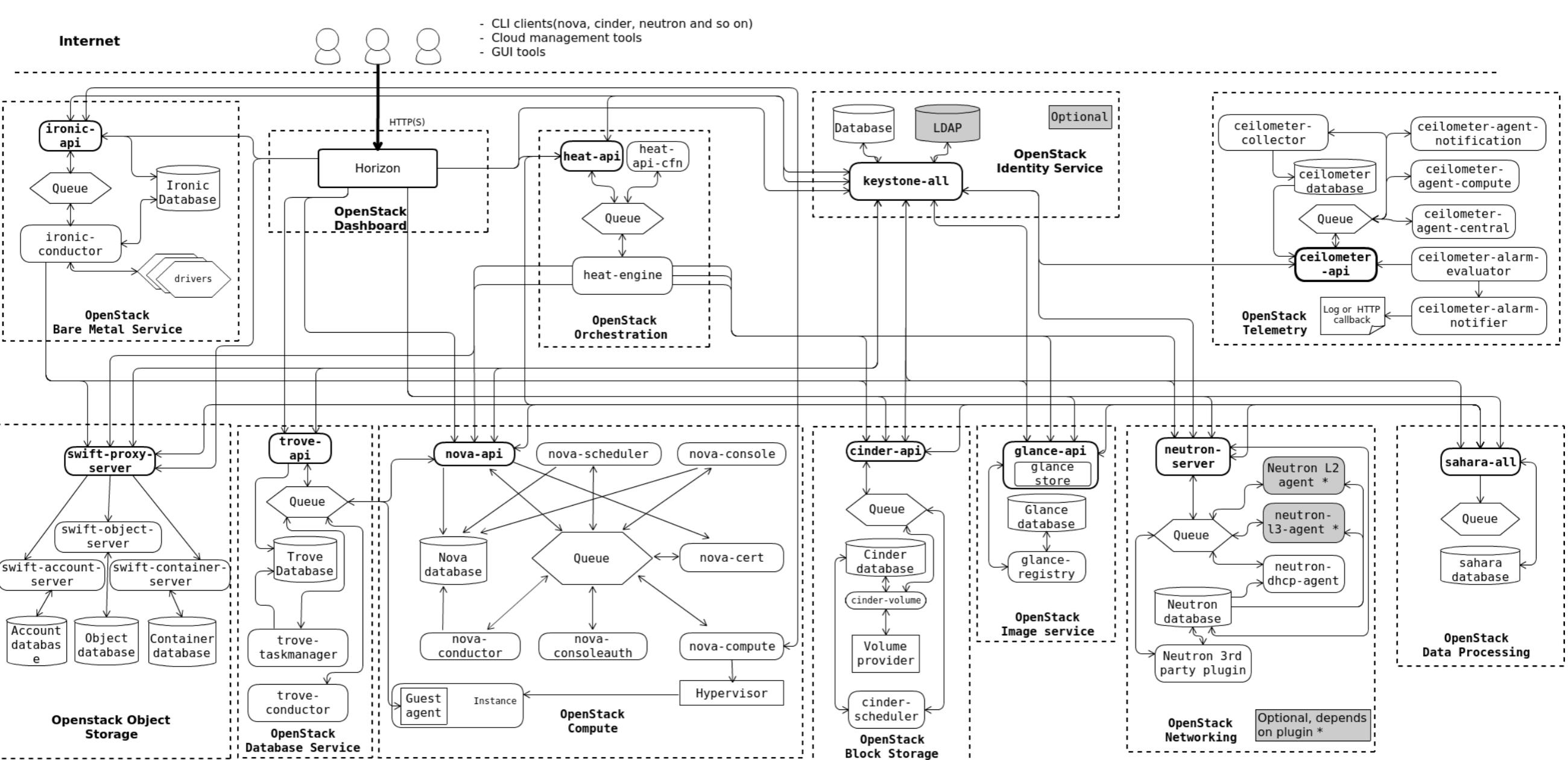


- CLI clients(nova, cinder, neutron and so on)
- Cloud management tools
- GUI tools



OPENSTACK (THE DEVIL IN DETAILS)

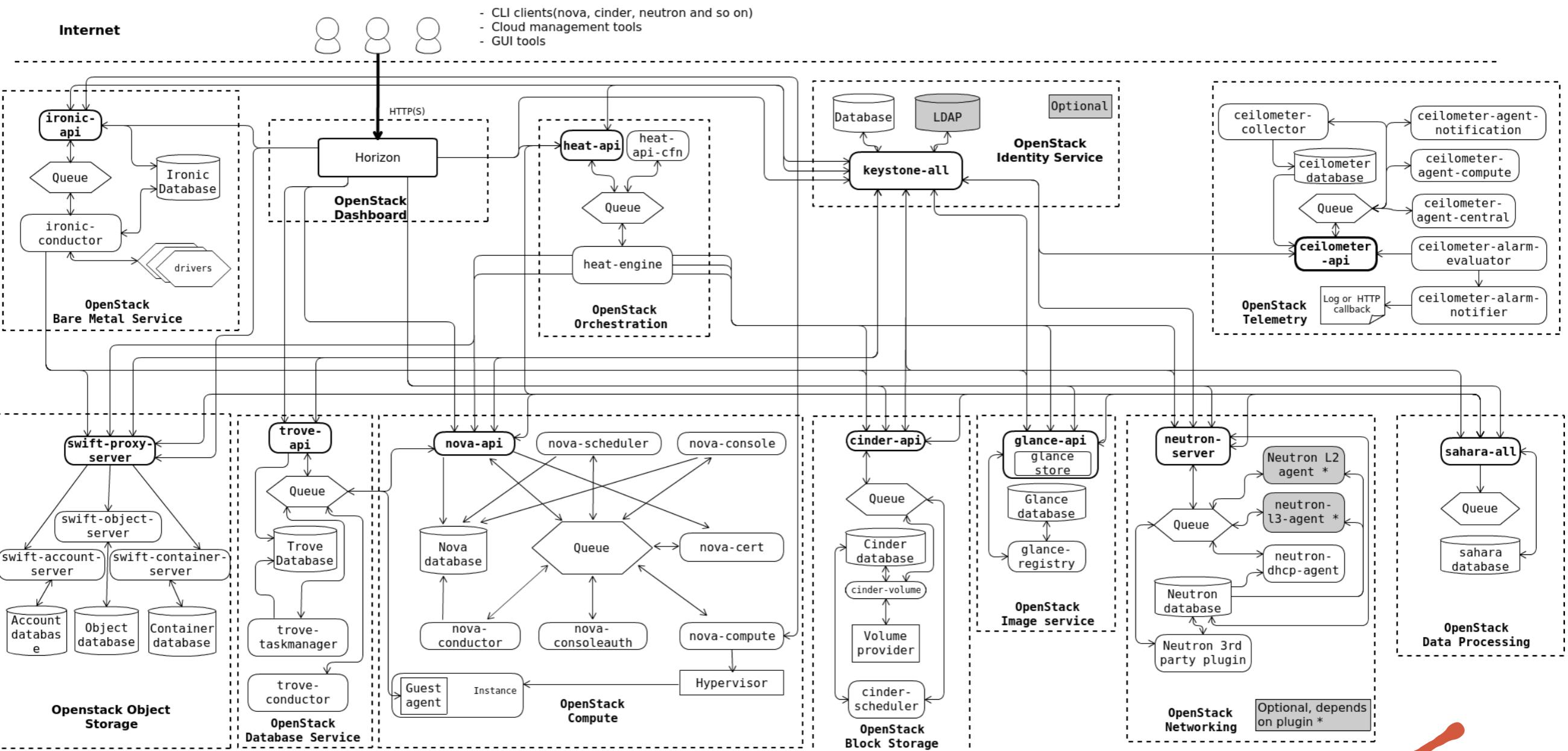
13 Millions of LOCs
186 subservices



OPENSTACK (THE DEVIL IN DETAILS)

13 Millions of LOCs
186 subservices

Simulation (a model of OpenStack)
ANR SONGS



OPENSTACK (THE DEVIL IN DETAILS)

13 Millions of LOCs
186 subservices

Simulation (a model of OpenStack)
ANR SONGS

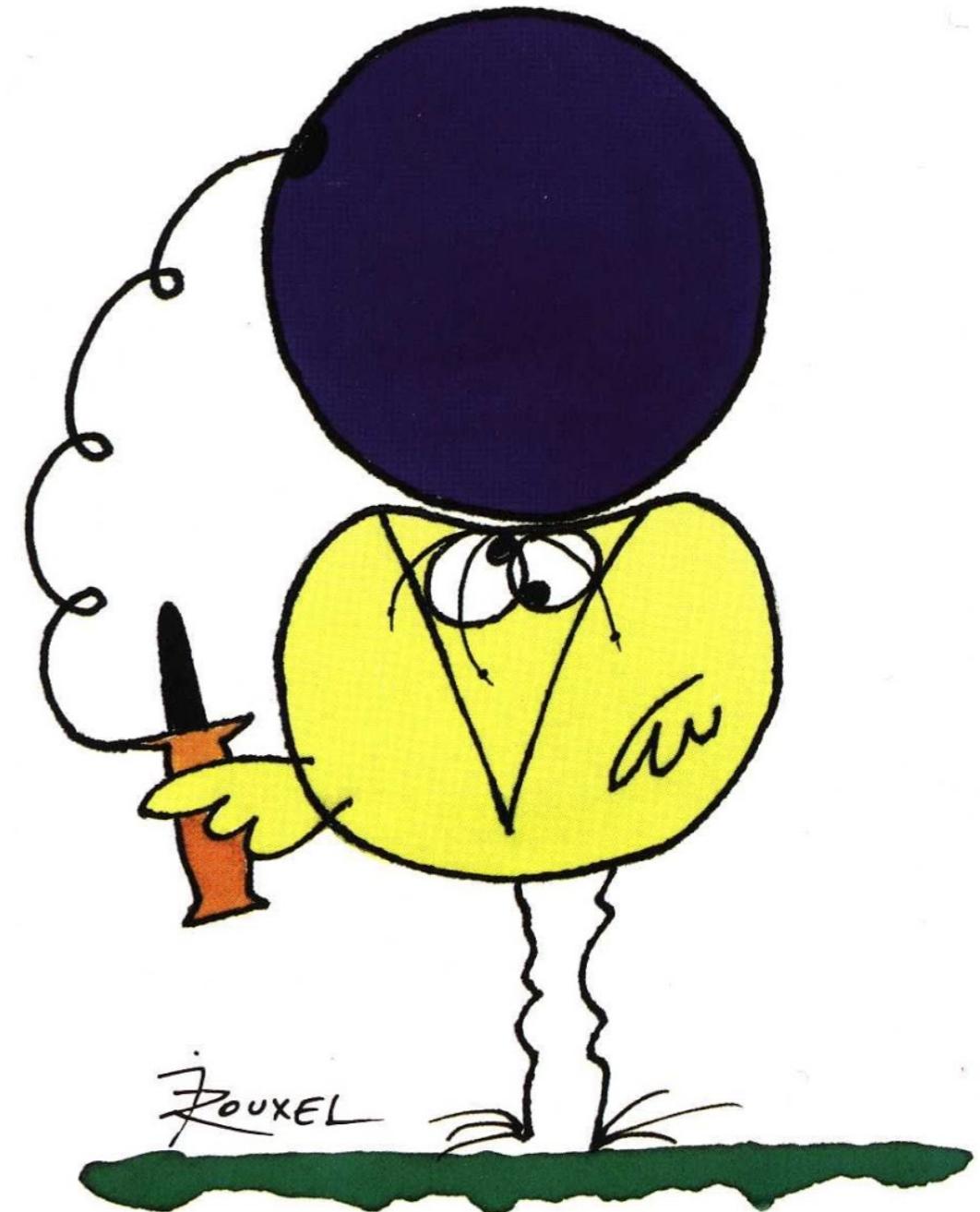
Grid'5000

OPENSTACK AND GRID'5000

.....

- Understand how does OpenStack run and identify possible improvements
- Make new proposals and evaluate them under multiple setups/topologies
- Conducting performance analyses in a scientific and reproducible manner
 - Automation
 - At small and large-scale
Under different network topologies (traffic shaping)
 - Between different releases
With any kind of benchmarks
- Deploying cloud stacks on top of G5K
 - A (long) run started since 2013
 - Generally to evaluate Applications (not the stack itself)

Les devises Shadok



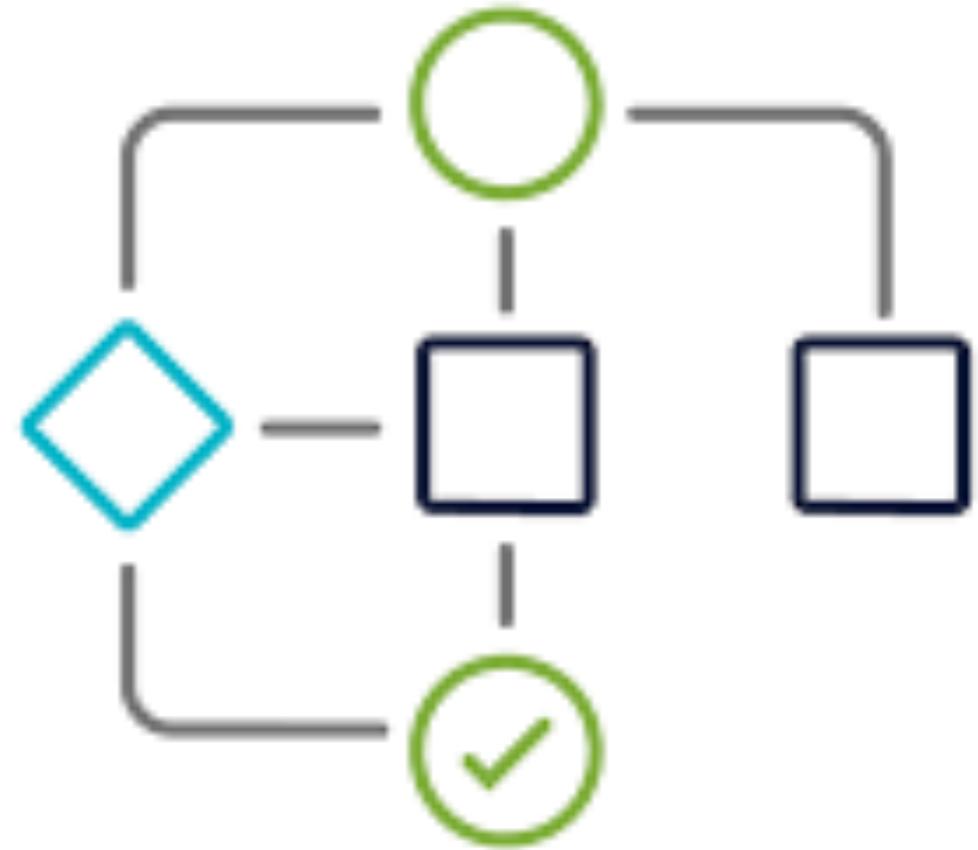
EN ESSAYANT CONTINUELLEMENT
ON FINIT PAR RÉUSSIR. DONC:
PLUS ÇA RATE, PLUS ON A
DE CHANCES QUE ÇA MARCHE.

ENOS

EXPERIMENTAL ENVIRONMENT FOR OPENSTACK

► *Workflow*

1. *Book and provision servers*
2. *Deploy OpenStack*
3. *Perform benchmarks*
4. *Collect and save metrics*
5. *Visualize & share*



```
$ enos deploy  
$ enos bench  
$ enos backup
```

ENOS DEPLOY

RESOURCE/TOPOLOGY DESCRIPTION

```
$ cat ./basic.yml  
resources:  
  clusterA:  
    control: 1  
    network: 1  
  clusterB:  
    compute: 50  
  
$ enos deploy -f basic.yml
```

```
$ cat ./advanced.yml  
resources:  
  clusterA:  
    control: 1  
    network: 1  
    nova-conductor: 5  
  clusterB:  
    compute: 50  
  
$ enos deploy -f advanced.yml
```

```
$ cat ./network-topo.yml  
resources:  
  grp1:  
    clusterA:  
      control: 1  
      network: 1  
      nova-conductor: 5  
  grp2:  
    clusterB:  
      compute: 50  
network_constraints:  
  src: grp1  
  dst: grp2  
  delay: 100ms  
  rate: 10Gbit  
  loss: 0%  
  symmetric: true  
  
$ enos deploy -f network-topo.yml
```

ENOS DEPLOY UNDER THE HOOD

.....

```
resources:
```

```
grp1:  
  clusterA:  
    control: 1  
    network: 1
```

```
grp2:  
  clusterB:
```

```
compute: 50
```

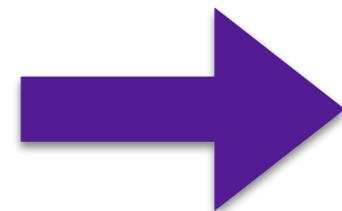
```
network_constraints:
```

```
delay: 100ms
```

```
rate: 10Gbit
```

```
loss: 0%
```

```
$ enos deploy
```



1. Provider gets 2 nodes on clusterA, 50 nodes on clusterB and returns node's IP addresses
2. EnOS provisions nodes with Docker daemons
3. EnOS installs OpenStack using Kolla
4. EnOS sets up bare necessities (flavors, cirros image, router, ...)
5. EnOS applies network constraints between grp1 and grp2 using tc

-
- From preliminary scripts to a engine for replicable experiments
 - Provider to get testbed resources
 - Resources: anything running a Docker daemon and EnOS can SSH to + some IPs
 - Existing Provider: Vagrant (VBox/Libvirt), Grid'5000, Chameleon, OpenStack (~500 LoC each)
 - Kolla to deploy OpenStack over testbed resources (containerised based deployment)
 - TC to apply network constraints (Linux Netem)

► *Benchmarks description*

```
$ cat ./run.yml
rally:
  args:
    concurrency: 5
    times: 100
  scenarios:
    - name: boot and list servers
      file: nova-boot-list-cc.yml
      osprofiler: true
    - ...
shaker: ...
$ enos bench --workload=run.yml
```

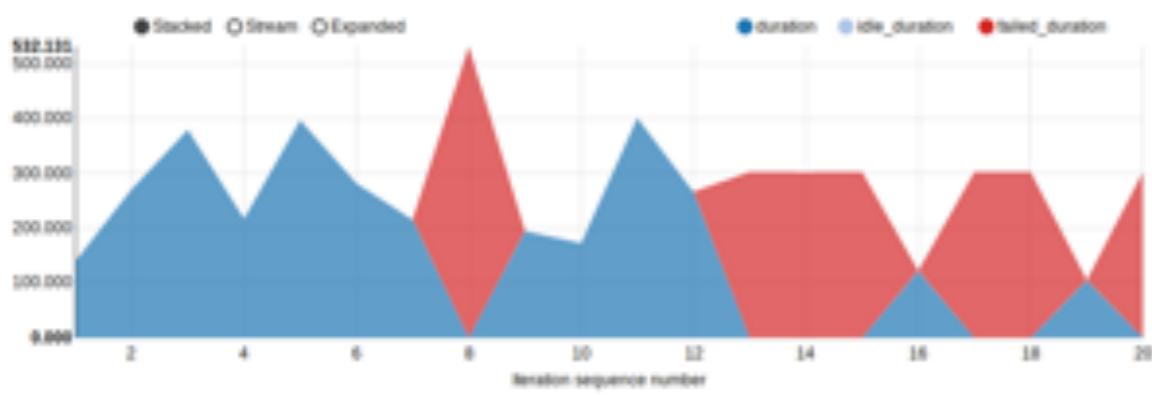
► *Under the hood*

- *Rally: control plane benchmark*
- *Shaker: data plane benchmark*
- *OSProfiler: code profiling*
- *Monitoring stack: cAdvisor/Collectd to collect CPU/RAM/Network consumption per service/node/cluster*

ENOS BACKUP

-
 - *EnOS backup produces a tarball with*
 - *Rally/Shaker reports*
 - *InfluxDB database with cAdvisor/collectd measures*
 - *OpenStack logs*
 - *OS Profiler traces*

Action	Min (sec)	Median (sec)	90%ile (sec)	95%ile (sec)	Max (sec)	Avg (sec)	Success	Count
nova.boot_server	55.826	205.736	263.908	271.632	279.61	181.196	65.0%	20
nova.delete_server	10.278	40.468	135.245	146.816	162.718	63.236	92.9%	14
total	104.094	217.546	294.025	399.115	401.927	249.932	65.0%	20



ENOS BACKUP

-
 - *EnOS backup produces a tarball with*
 - *Rally/Shaker reports*
 - *InfluxDB database with cAdvisor/collectd measures*
 - *OpenStack logs*
 - *OS Profiler traces*



<https://enos.readthedocs.io/>

EnOS BACKUP

-
- *EnOS backup produces a tarball with*
 - *Rally/Shaker reports*
 - *InfluxDB database with cAdvisor/collectd measures*
 - *OpenStack logs*
 - *OS Profiler traces*





Chasing 1000 nodes scale (Inria/Mirantis)

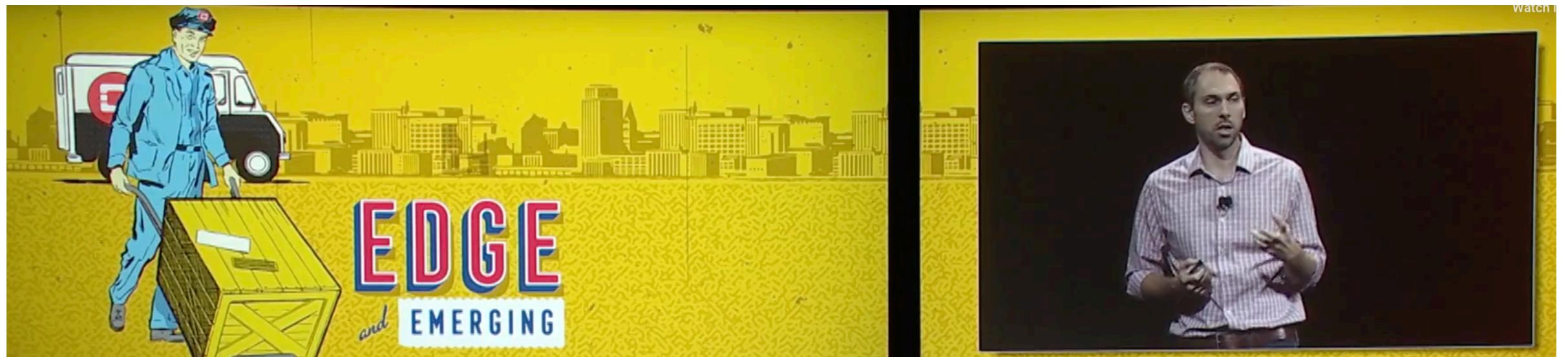
The slide features a blue header with the text "Toward Fog, Edge, and NFV Deployments- Evaluating OpenStack WANwide" and a red section title "TestBed – Chameleon". Below the title is a bulleted list of features:

- NSF-funded testbed for computer science experimentation
- Built with OpenStack software (and some from Grid'5000)
- **Reconfigurable:** node reservation (Blazar), bare-metal deployment (Ironic)
- **Large-scale:** 504 compute nodes & 48 storage nodes, 3.6 PB global storage
- **Heterogeneous hardware:** Infiniband, NVMe, SSDs, GPUs, FPGAs, ARM & Atom
- Hardware distributed over two sites: TACC (Austin, TX) and UC (Chicago, IL)
- Serving ~1,400 users in 200+ projects

At the bottom of the slide, a red button contains the text: "Just after this talk in MR 208: ‘We Need Clouds to Build Clouds: Developing an Open Cloud Testbed Using OpenStack’".

OPENSTACK
SUMMIT → BOSTON

OpenStack WANWide (Inria, Orange, Univ Chicago)



OPENSTACK COLLABORATES WITH OTHER EDGE GROUPS



Fog Edge Massively Distributed Clouds

The goal of the Fog/Edge/Massively Distributed Clouds SIG is to guide the OpenStack community to best address fog/edge as the supervision and use of a large number of remote mini/micro/nano data centers through a collaborative OpenStack advances the topic through debate and investigation of requirements for various implementation options.

Status: active

Contact: Adrien Lebre <adrien.lebre@inria.fr> Paul-André Raymond <paul-andre.raymond@b-yond.com> Gergely Csatai

Contents

- [hide]
- [1 Meetings](#)
- [2 Problem description](#)
- [3 Mission](#)
- [4 Interactions with other Groups](#)
- [5 How to participate](#)
- [6 Planned Actions for Rocky cycle](#)
- [7 Achieved Actions](#)
- [8 cross-cycle actions](#)
- [9 Previous documents](#)

Projects User Stories Community Blog Wiki **Documentation**

Performance Documentation

Abstract

This documentation section aims to introduce ideas and best practices on how OpenStack cloud may be exercised to evaluate its performance, what can be called performance at all and how it can be measured. Here you will find both testing/probing methodologies and test plans, proposed by OpenStack community members (as well as results of their execution against various cloud topologies).

Contents

- [1. Introduction](#)
 - [1.1. What is the reason for writing this?](#)
 - [1.2. Who will be interested in this?](#)
 - [1.3. Who is contributing to this guide?](#)
- [2. OpenStack summits news](#)
 - [2.1. Ocata OpenStack summit recap](#)
 - [2.2. Newton OpenStack summit recap](#)
- [3. Methodologies](#)
 - [3.1. Tools](#)
 - [3.2. Methodology for testing Hyper-Scale](#)
 - [3.3. Methodology for Containerized Openstack Monitoring](#)
 - [3.4. HA InfluxDB as an external storage for Prometheus](#)
- [4. Labs](#)
 - [4.1. Intel-Mirantis Performance-Team Lab #1](#)
 - [4.2. Intel Mirantis Performance-Team Lab #2](#)
 - [4.3. Grid'5000](#)

Inria has a key actor for edge related challenges within the OpenStack framework

Multi-Level Elasticity for Data Stream Processing

Vania Marangozova-Martin, Noël de Palma and Ahmed El Rhedane
 Univ. Grenoble Alpes, CNRS, LIG, F-38000 Grenoble France
 E-mail: firstName.secondName@imag.fr

Abstract—This paper investigates reactive elasticity in stream processing environments where the performance goal is to analyze large amounts of data with low latency and minimum resources. Working in the context of Apache Storm, we propose an elastic management strategy which modulates the parallelism degree of applications' components while explicitly addressing the hierarchy of execution containers (virtual machines, processes and threads). We show that provisioning the wrong kind of container may lead to performance degradation and propose a solution that provisions the least expensive container (with minimum resources) to increase performance. We describe our monitoring metrics and show how we take into account the specifics of an execution environment. We provide an experimental evaluation with real-world applications which validates the applicability of our approach.

Index Terms—stream processing, multi-level elasticity, Apache Storm

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2019.2907950, IEEE

Transactions on Parallel and Distributed Systems

SUBMISSION TO TPDS

12

would need to be deployed in containers with different capacities which in turn call for multi-dimensional-bin-packing-oriented scheduling [45].

ACKNOWLEDGEMENTS

The experimental work presented in this paper would not have been possible without the existence of the Grid'5000 platform and the help of the supporting teams. The authors would also like to thank the *enos* team who made the Openstack deployment process a child's play.

- [18] “CoMD,” <https://gpuopen.com/compute-product/comd/>.
- [19] Y. Wu and K. L. Tan, “ChronoStream: Elastic Stateful Stream Computation in the Cloud,” in *2015 IEEE 31st International Conference on Data Engineering*, April 2015, pp. 723–734.
- [20] V. Gulisano, R. Jimenez-Peris, M. Patino-Martinez, C. Soriente, and P. Valduriez, “StreamCloud: An Elastic and Scalable Data Streaming System,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 12, pp. 2351–2365, Dec. 2012.
- [21] L. Neumeyer, B. Robbins, A. Nair, and A. Kesari, “S4: Distributed stream computing platform,” in *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*. IEEE, 2010, pp. 170–177.
- [22] OpenStack. <https://www.openstack.org/>.
- [23] “Grid’5000,” <http://www.grid5000.fr/>.
- [24] R. Cherrueau, D. Pertin, A. Simonet, A. Lebre, and M. Simonin, “Toward a Holistic Framework for Conducting Scientific Evaluations of OpenStack,” in *2017 17th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid)*, May 2017, pp. 1–8.

Not really rocket science but ... it is used !

Multi-Level Elasticity for Data Stream Processing

Vania Marangozova-Martin, Noël de Palma and Ahmed El Rhedane
 Univ. Grenoble Alpes, CNRS, LIG, F-38000 Grenoble France
 E-mail: firstName.secondName@imag.fr

Abstract—This paper investigates reactive elasticity in stream processing environments where the performance goal is to analyze large amounts of data with low latency and minimum resources. Working in the context of Apache Storm, we propose an elastic management strategy which modulates the parallelism degree of applications' components while explicitly addressing the hierarchy of execution containers (virtual machines, processes and threads). We show that provisioning the wrong kind of container may lead to performance degradation and propose a solution that provisions the least expensive container (with minimum resources) to increase performance. We describe our monitoring metrics and show how we take into account the specifics of an execution environment. We provi

Inde

ACKNOWLEDGEMENTS

The experimental work presented in this paper would not have been possible without the existence of the Grid'5000 platform and the help of the supporting teams. The authors would also like to thank the *enos* team who made the Openstack deployment process a child's play.

n: DOI 10.1109/TPDS.2019.2907950, IEEE

12
 e-product/comd/.
 tic Stateful Stream Com-
 t International Conference
 34.

ACKNOWLEDGEMENTS

The experimental work presented in this paper would not have been possible without the existence of the Grid'5000 platform and the help of the supporting teams. The authors would also like to thank the *enos* team who made the Openstack deployment process a child's play.

- [20] V. Gómez, R. Jiménez-Rens, M. Raúl-Martínez, C. Soriente, and P. Valduriez, "StreamCloud: An Elastic and Scalable Data Streaming System," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 12, pp. 2351–2365, Dec. 2012.
- [21] L. Neumeyer, B. Robbins, A. Nair, and A. Kesari, "S4: Distributed stream computing platform," in *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*. IEEE, 2010, pp. 170–177.
- [22] OpenStack. <https://www.openstack.org/>.
- [23] "Grid'5000," <http://www.grid5000.fr/>.
- [24] R. Cherreau, D. Pertin, A. Simonet, A. Lebre, and M. Simonin, "Toward a Holistic Framework for Conducting Scientific Evaluations of OpenStack," in *2017 17th IEEE/ACM International Conference on Grid Computing (Grid'17)*, pp. 1–8, IEEE, 2017.

Not really rocket science but ... it is used !

Multi-Level Elasticity for Data Stream Processing

Vania Marangozova-Martin, Noël de Palma and Ahmed El Rhedane
 Univ. Grenoble Alpes, CNRS, LIG, F-38000 Grenoble France
 E-mail: firstName.secondName@imag.fr

Abstract-
 large amount of data management, execution and performance management are provided by cloud providers.

Index

This article has bee

SUBMISSIONS
 would
 capacity
 packing

ACKNOWLEDGMENTS

The experimental work presented in this paper would not have been possible without the existence of the Grid'5000 platform and the help of the supporting teams. The authors would also like to thank the *enos* team who made the Openstack deployment process a child's play.

Riza O. Suminto
 To: users@lists.chameleoncloud.org
 [Chameleon-users] Solution to deploy OpenStack on Chameleon

16 March 2018 at 15:41

Trash - @Inria

RO

Hi,

Does anyone have good solution to deploy OpenStack in ChameleonCloud baremetal?
 My goal is to create several medium size VMs that have internet connectivity (just for download).

Before, I was able to use Devstack template from Chameleon Appliances page, but it looks like none of the template works anymore.

I have tried using [enos](#), looks like it successfully deploy everything well. But I have no idea how to configure it such that the VM instance will have internet connectivity to outside world.

Thank you,
 Riza

Chameleon Users mailing list
 Manage your subscription at <https://www.chameleoncloud.org/user/profile/subscriptions/>



"ChameleonCloud: An Elastic and Scalable Data Streaming System," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 12, pp. 2351–2365, Dec. 2012.

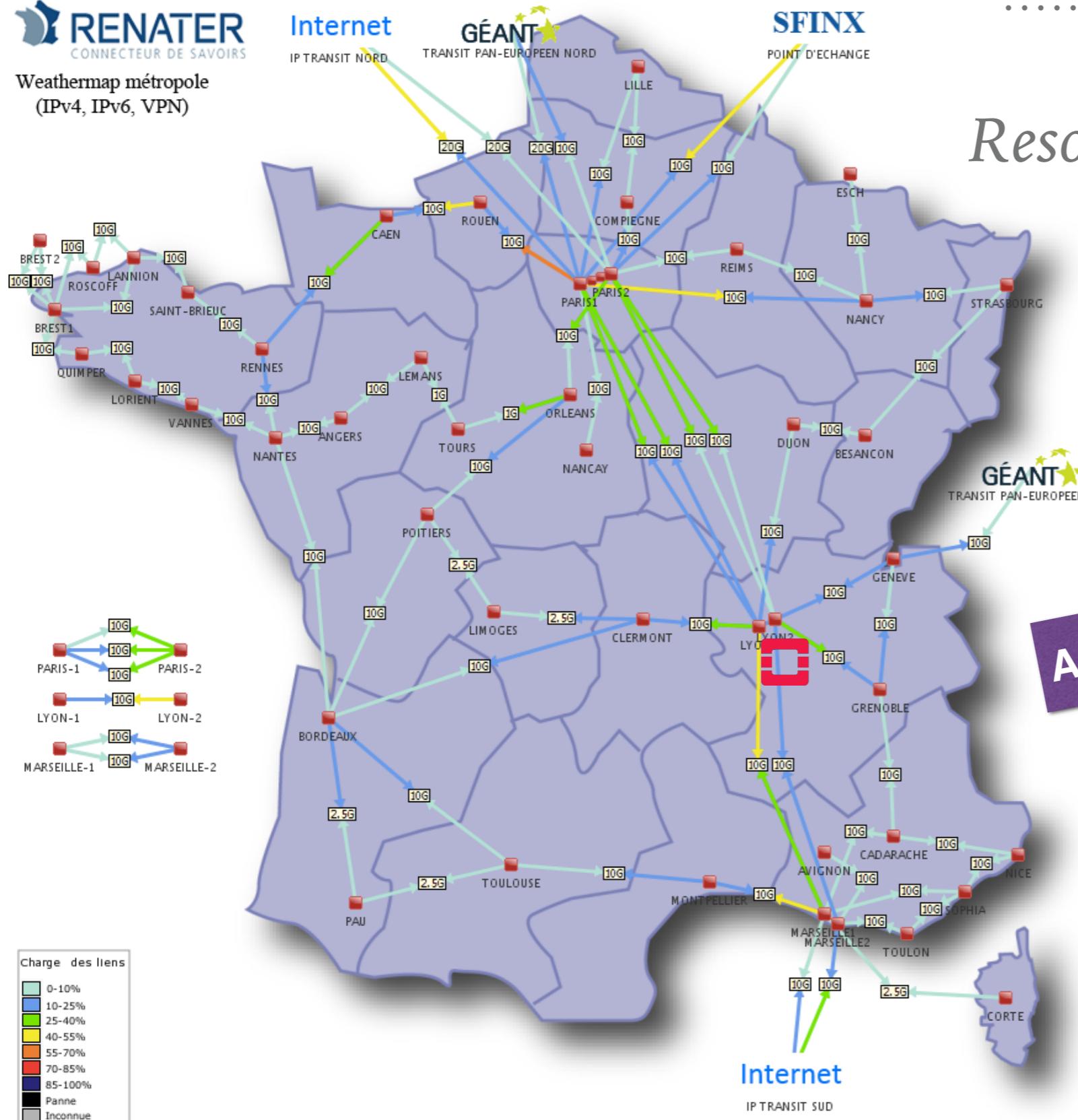
- [21] L. Neumeyer, B. Robbins, A. Nair, and A. Kesari, "S4: Distributed stream computing platform," in *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*. IEEE, 2010, pp. 170–177.
- [22] OpenStack. <https://www.openstack.org/>.
- [23] "Grid'5000," <http://www.grid5000.fr/>.
- [24] R. Cherreau, D. Pertin, A. Simonet, A. Lebre, and M. Simonin, "Toward a Holistic Framework for Conducting Scientific Evaluations of OpenStack," in *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*.

Not really rocket science but ... it is used !

LET'S GO BACK TO DISCOVERY



Weathermap métropole
(IPv4, IPv6, VPN)



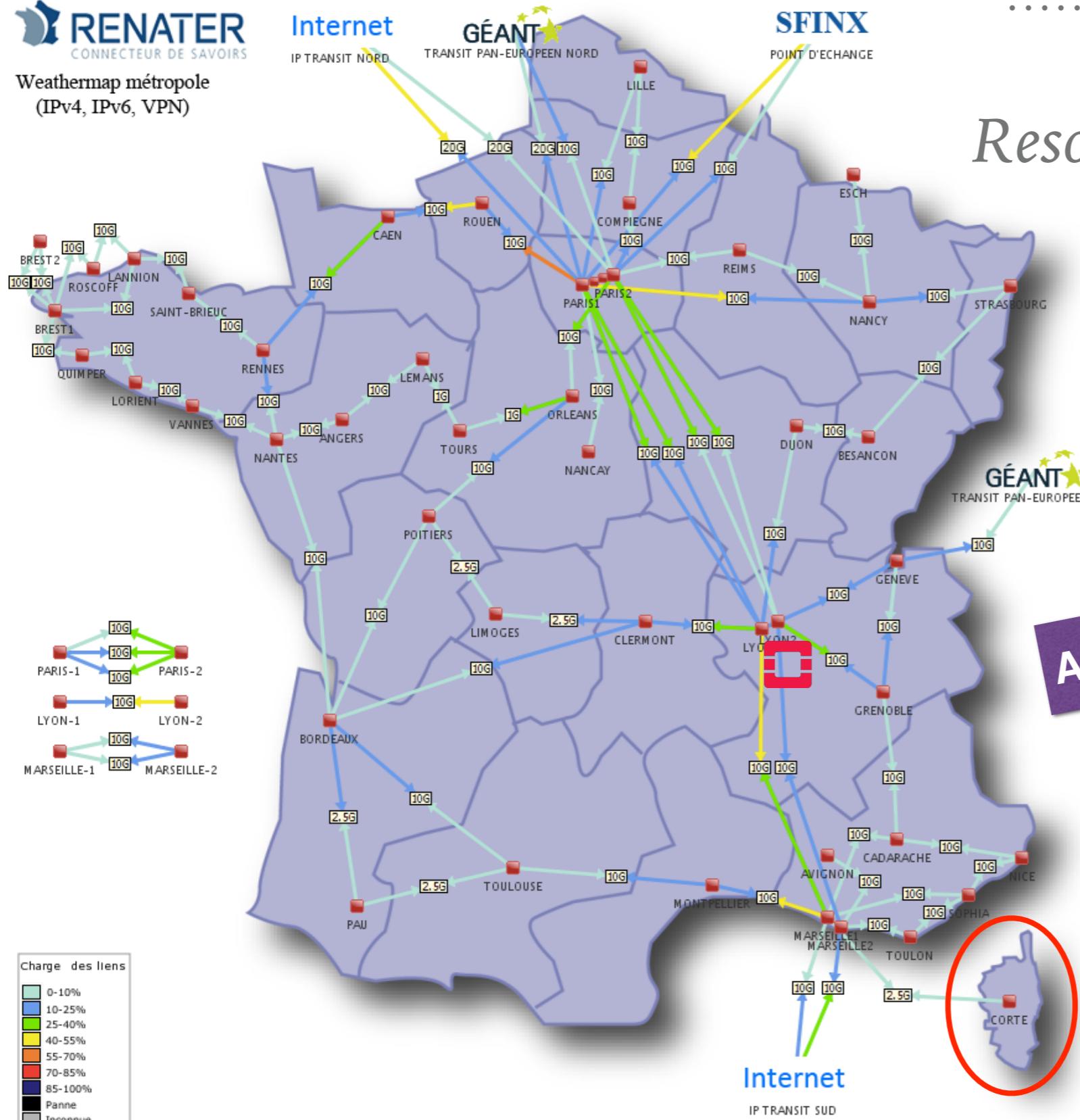
Edge Computing
Resource Management System:
a Critical Building Block!
HotEdge2018

A single control plane is not enough !

LET'S GO BACK TO DISCOVERY



Weathermap métropole
(IPv4, IPv6, VPN)



Last update: Mon Apr 08 13:49:02 CEST 2013

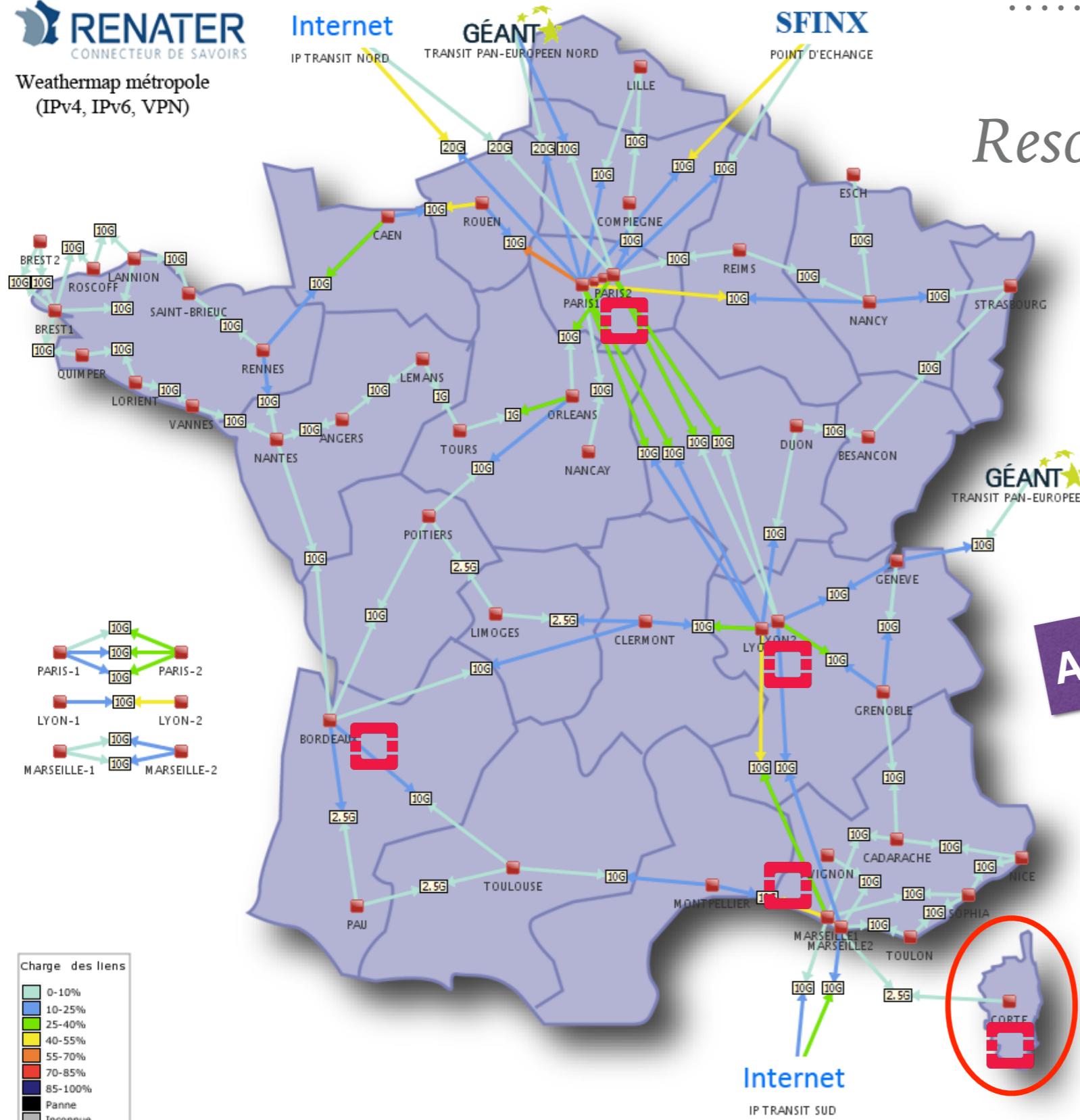
Edge Computing
Resource Management System:
a Critical Building Block!
HotEdge2018

A single control plane is not enough !

LET'S GO BACK TO DISCOVERY

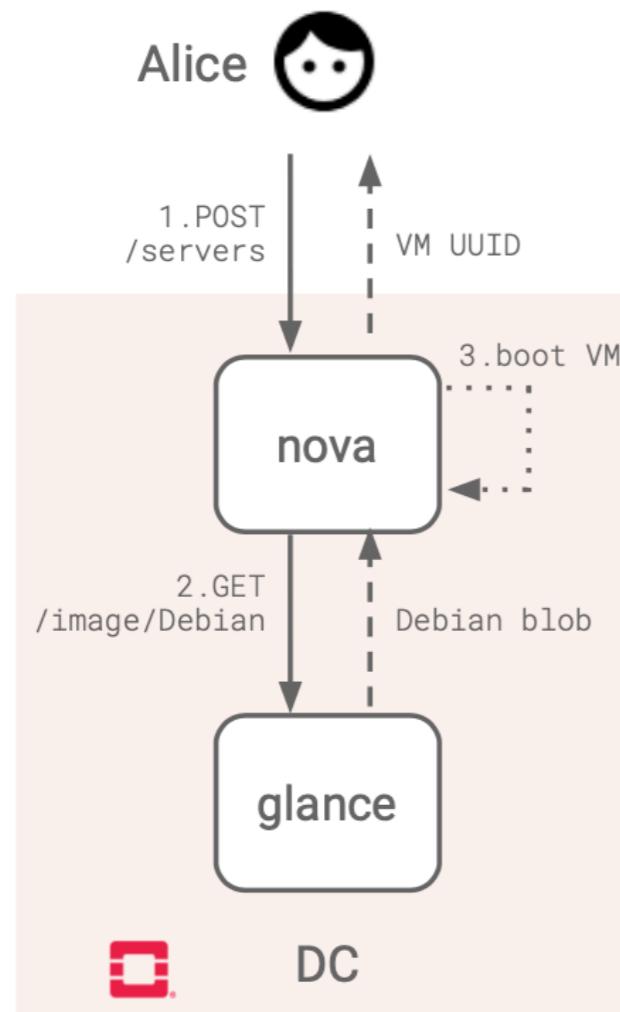


Weathermap métropole
(IPv4, IPv6, VPN)



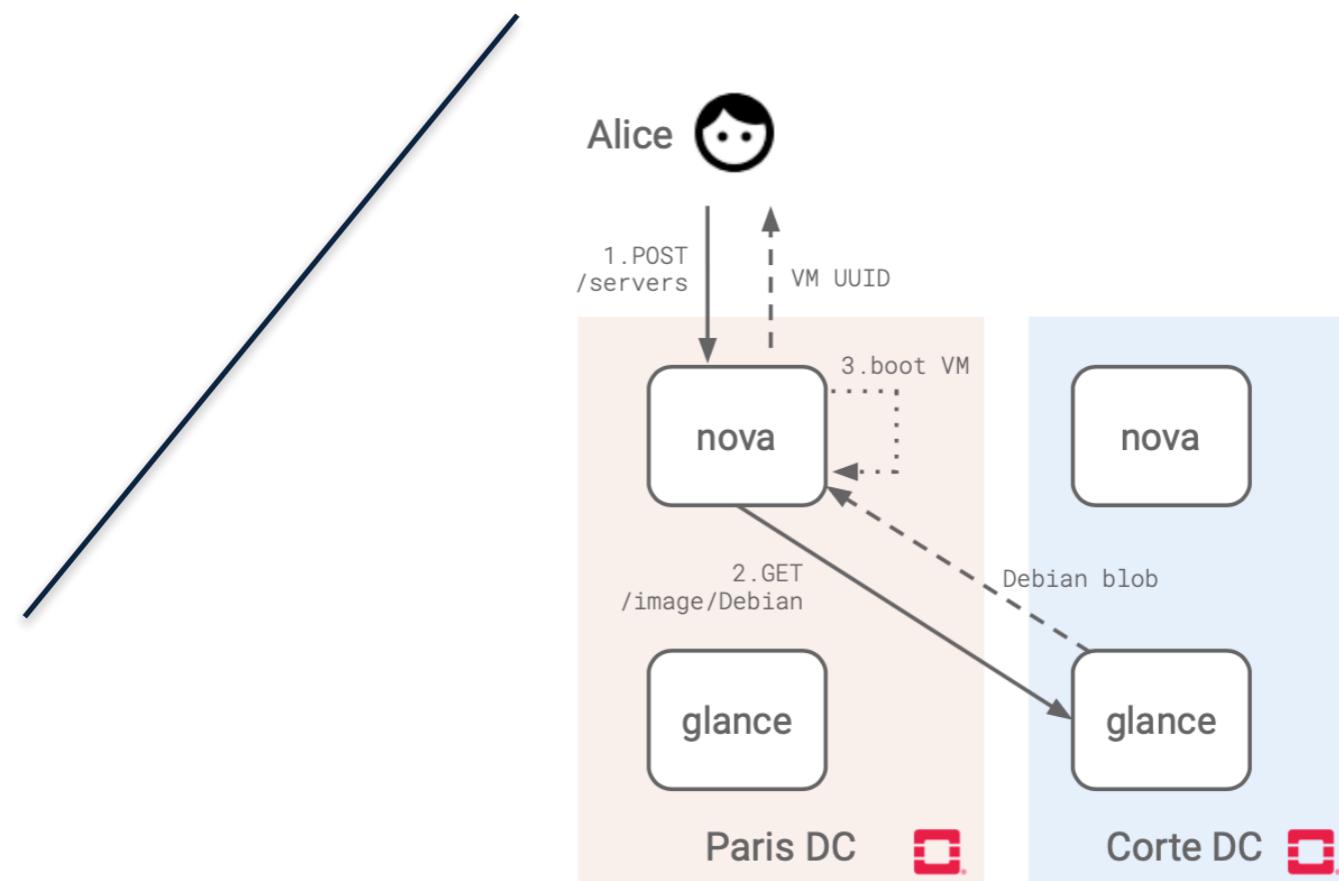
Edge Computing
Resource Management System:
a Critical Building Block!
HotEdge2018

A single control plane is not enough !



.....

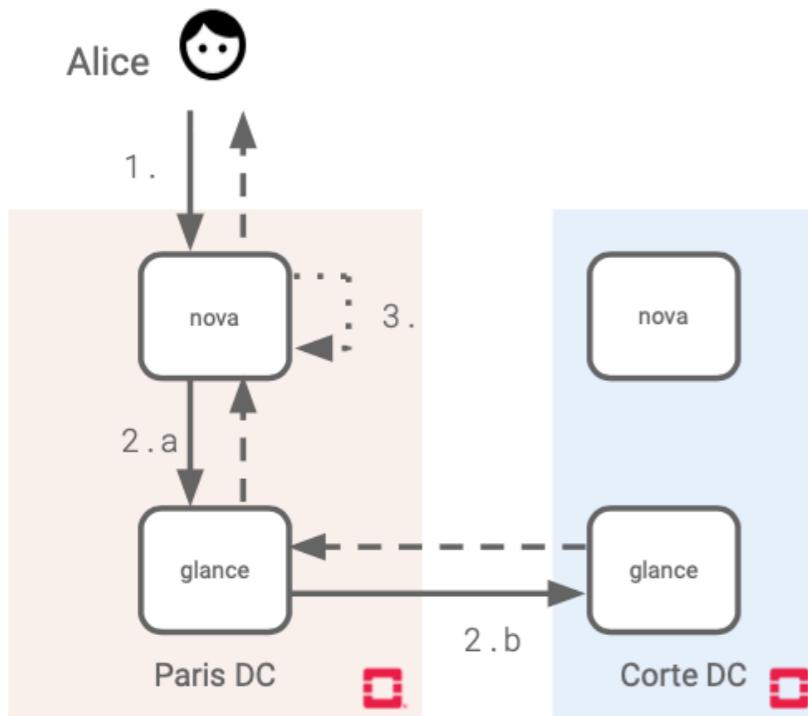
Red thread: Boot a VM with an image (i.e. a disk) from a distinct DC



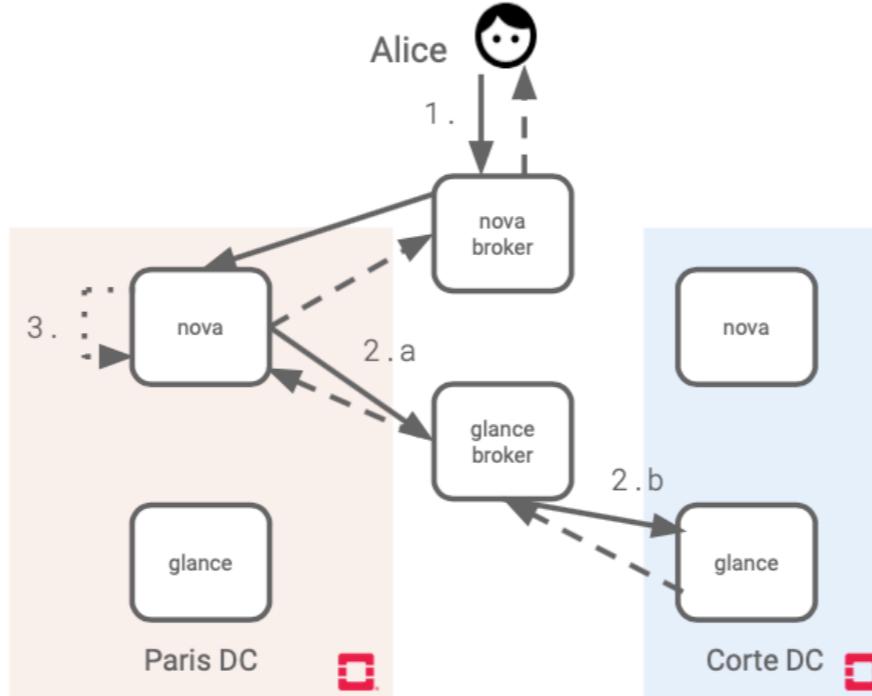
How can we achieve collaborations between multiples OpenStack instances ?

From the OpenStack Vanilla code to ad-hoc pieces of components

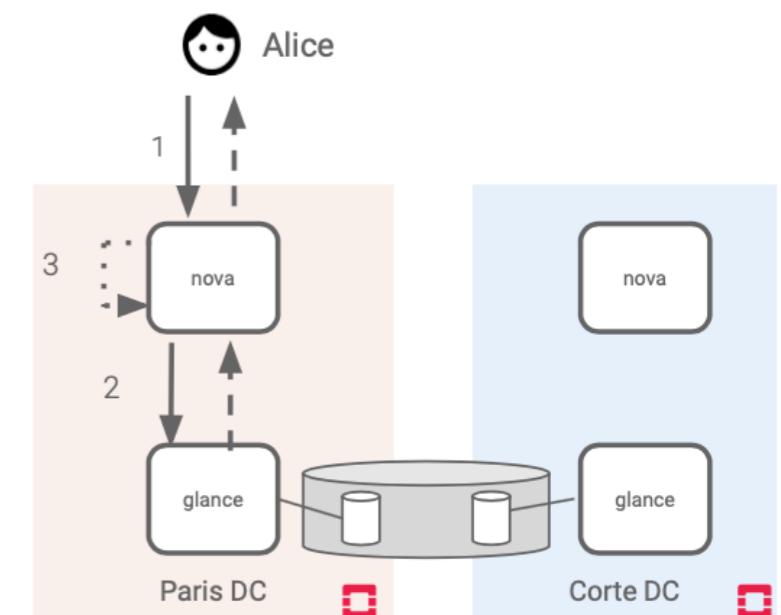
Service-to-service collaboration



Broker-based collaboration



Database collaboration



Pros/Cons of each approach?

Can we evaluate different approaches/building blocks?

Toward a generalisation of the ENOS Concepts

How we used Grid'5000

- Resource reservation and machine setup**
 - > OAR in deploy mode (get root access and full control on reserved machine)
 - > Kadeploy to install dependencies on reserved machines
- Building a VNF Router topology**
 - > Rely on Distem to deploy the VNF Router topology
 - Topology described in yaml
 - Deploy LXC on machines
 - Install VXLAN tunnels to connect LXC
- Automating experiments**
 - > Resource reserved in container mode
 - > Use Execo script to build a pool of experiment worker running a list of experiment tasks

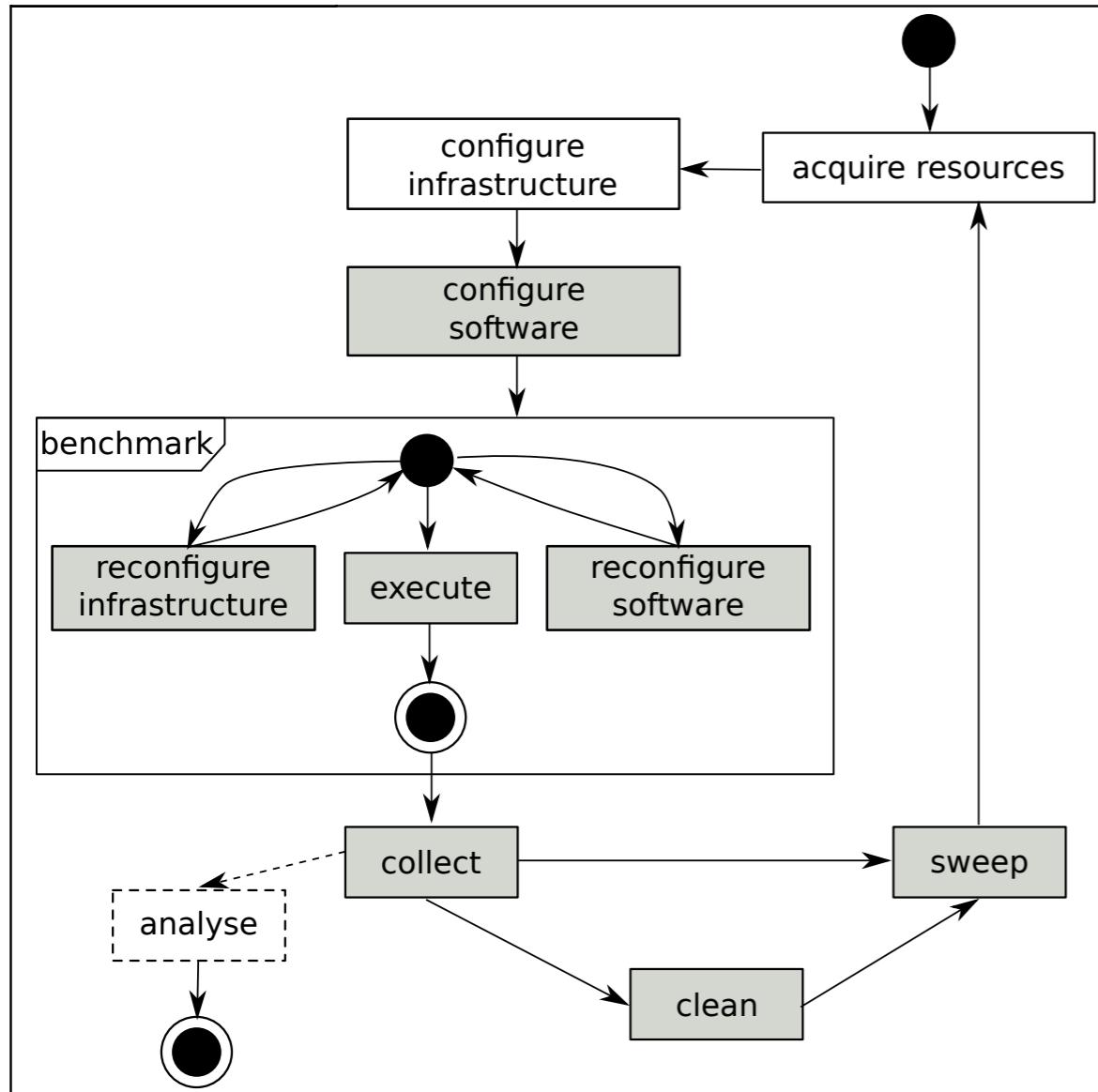
12 Grid'5000 helped us a lot!

```
graph LR; Topo[Topo.yaml] --> Python((Python Script)); Python -- "1) REST API call" --> Distem[Distem]; OAR[OAR] --> Distem; Kade[Kade ploy] --> Distem; Distem --> ResourcePool((Resource Pool)); ResourcePool -- "2) Deploy LXC" --> ResourcePool; ResourcePool -- "3) Setup VXLAN" --> ResourcePool; ResourcePool -- "4) Configure VNF Routers" --> ResourcePool; ResourcePool -- "5) Launch experiment" --> ResourcePool; ResourcePool -- "6) Get experiment data" --> ResourcePool;
```

THALES

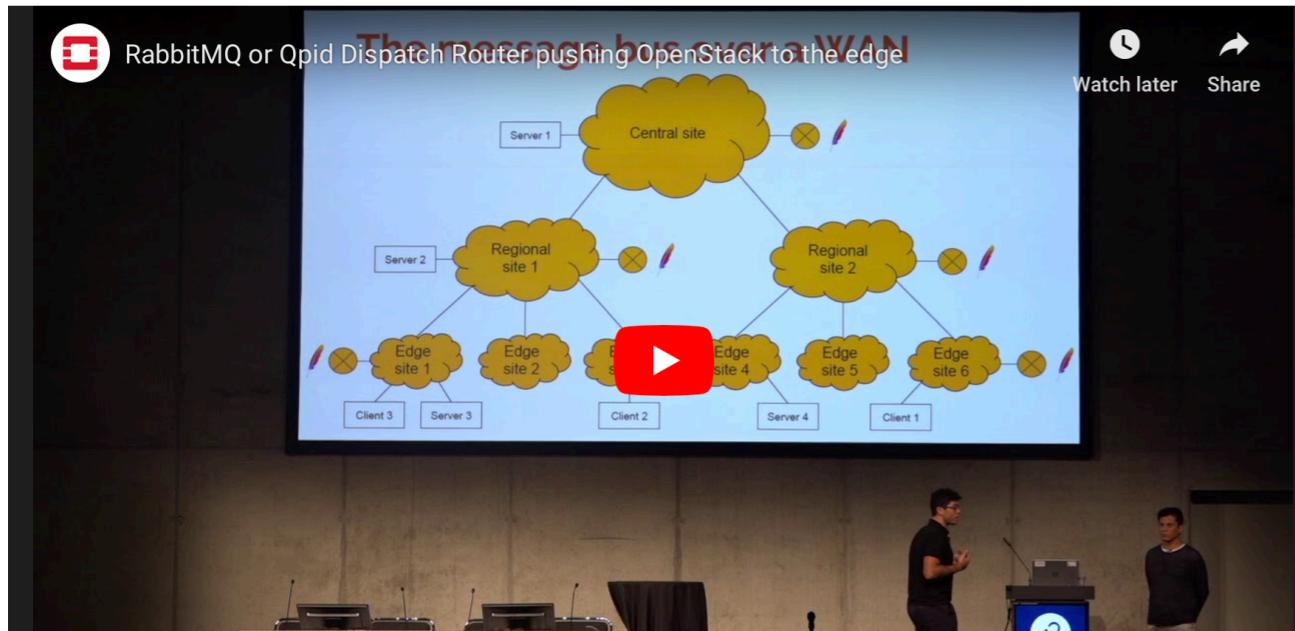
ENOSLIB

Toward a generalisation of the ENOS Concepts

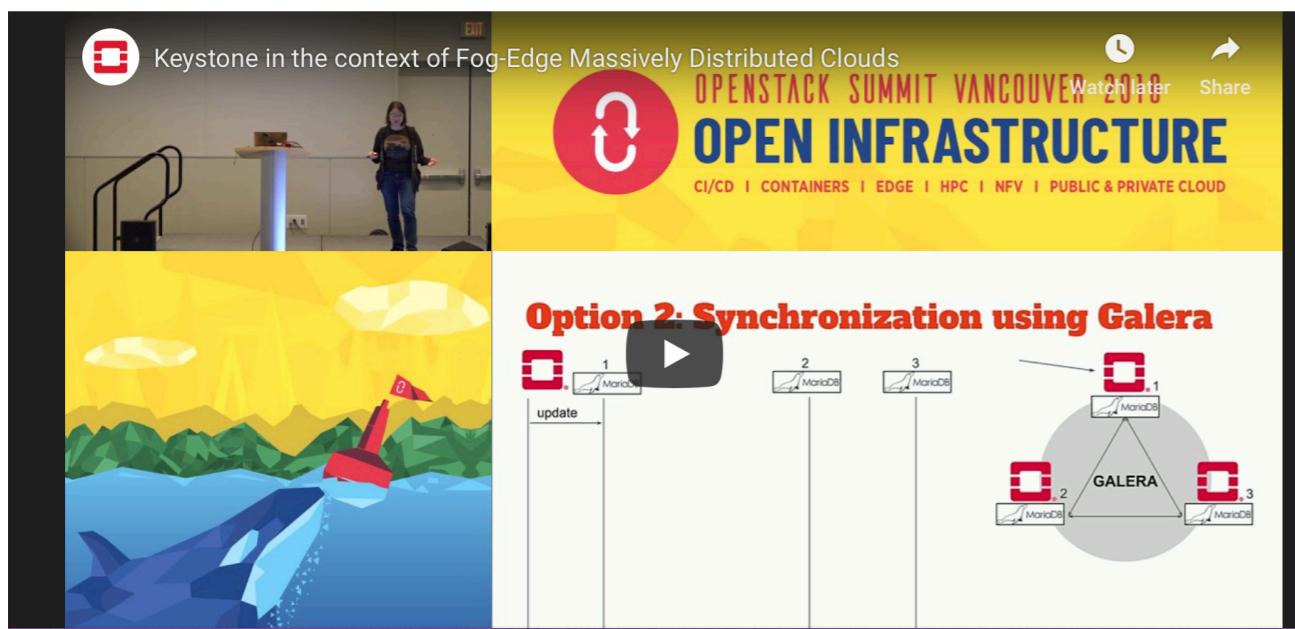


- Built on top of
 - EnOSTask
 - Resource (Provider)
 - Role (Ansible)
- ```
machines:
- flavor: parasilo
count: 1
roles:
- leader
- database
- monitoring
networks: [cn, dn]
- flavor: paravance
count: 10
roles:
- agents
- database
networks:
- control_network
- database_network
- flavor: tiny
count: 10
roles:
- agents
- database
networks:
- control_network
```
- ```
machines:  
- flavor: large  
count: 1  
roles:  
- leader  
- database  
- monitoring  
networks:  
- control_network  
- database_network  
- flavor: tiny  
count: 10  
roles:  
- agents  
- database  
networks:  
- control_network
```
- ```
machines:
- id: cn
roles:
- control_network
type: prod
- id: dn
roles:
- database_network
type: kavlan
```

# Toward a generalisation of the ENOS Concepts



Communication bus for the edge (Inria/Orange/Redhat)



Active/active replication (Inria)



## ► MADEUS

- A formal deployment model
- Fine grain modelization of the deployment process
- Performance / Safety
- H. Couillon, C. Perez et al.

# TODAY

---

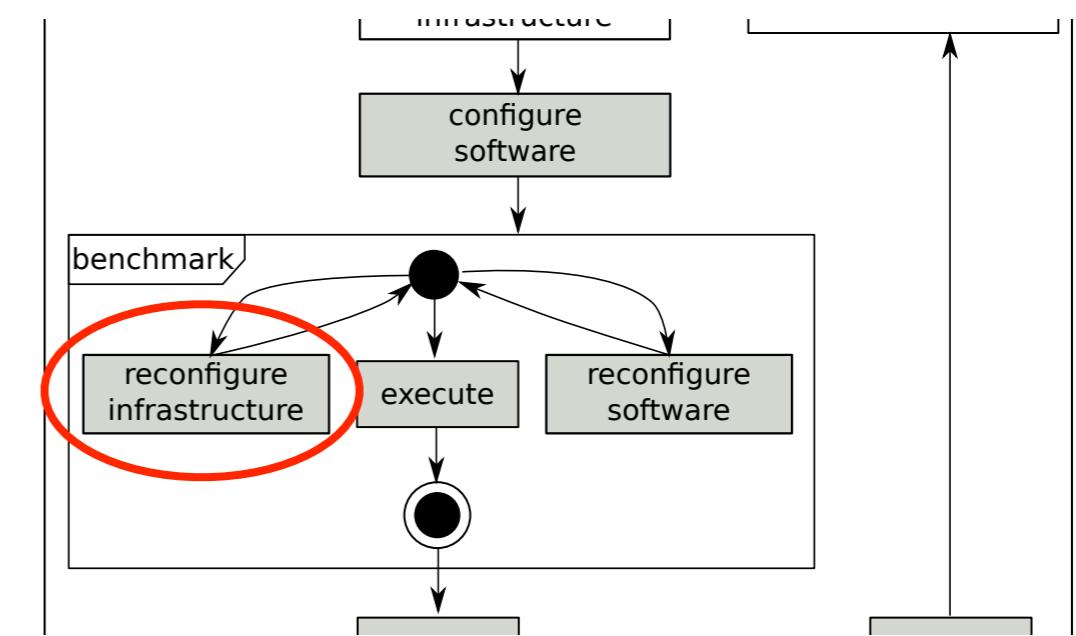
- Within the OpenStack ecosystem
  - Several experiment engines
  - Performance regression tests
- Enos-Kubernetes
- EnosLib and IPFS
- EnosLib and blockchain systems



# TOMORROW



- Fine monitoring of each role
- CPU, Memory, I/O (disks and net)
- Observe and identify applications patterns
- A injector of infrastructure events (representative to edge platforms)



# WHISHED LIST

---

- Emulate small devices/servers  
(CPU, memory, energy?)
- Wireless/xDSL/... networks
- Plug on-demand external  
servers/devices  
(The freebox use-case)





*Le présentateur ;)*

# LA PRESENTATION EST UN FAKE

---

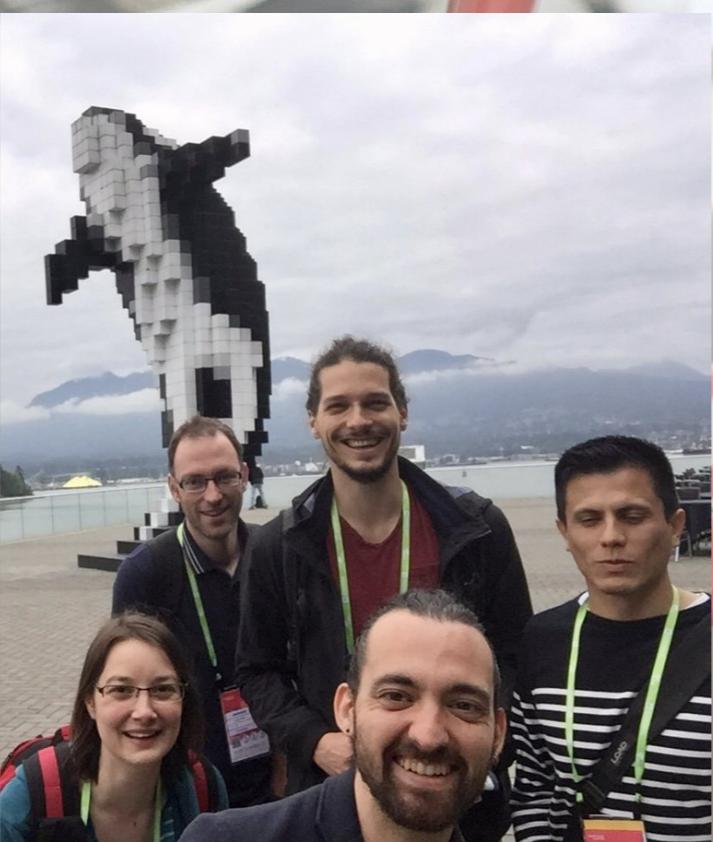
*Ronan-Alexandre Cherreau, Matthieu Simonin, Anthony Simonet, Alexandre Van Kempen, Marie Delavergne, Javier Rojas Balderrama, Dimitri Pertin, et les très nombreux beta-testeurs, Pierre Riteau, Jonathan Pastor, Adwait Bauskar, Ali Sanhaji ...*



*Le présentateur ;)*

# LA PRESENTATION EST UN FAKE

*Ronan-Alexandre Cherreau, Matthieu Simonin, Anthony Simonet, Alexandre Van Kempen, Marie Delavergne, Javier Rojas Balderrama, Dimitri Pertin, et les très nombreux beta-testeurs, Pierre Riteau, Jonathan Pastor, Adwait Bauskar, Ali Sanhaji ...*



*Le présentateur ;)*

# LA PRESENTATION EST UN FAKE

*Ronan-Alexandre Cherreau, Matthieu Simonin, Anthony Simonet, Alexandre Van Kempen, Marie Delavergne, Javier Rojas Balderrama, Dimitri Pertin, et les très nombreux beta-testeurs, Pierre Riteau, Jonathan Pastor, Adwait Bauskar, Ali Sanhaji ...*



Orange Labs

Inria  
inventeurs du monde numérique

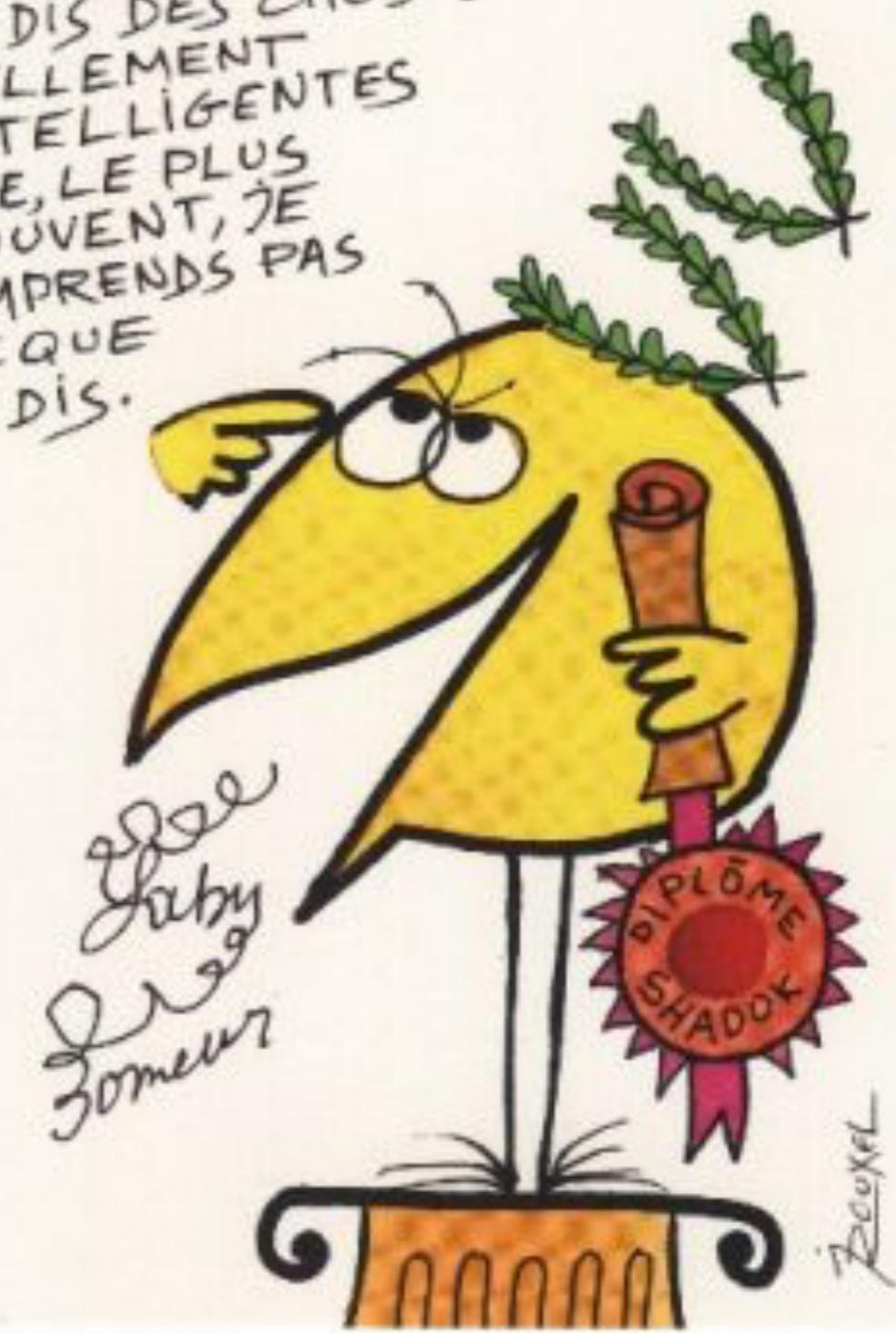
# QUESTIONS

---

► Merci

Les devises Shadok

JE DIS DES CHOSES  
TELLEMENT  
INTELLIGENTES  
QUE, LE PLUS  
SOUVENT, JE  
COMPRENDS PAS  
CE QUE  
JE DIS.



# QUESTIONS

---

► Merci

Les devises Shadok

JE DIS DES CHOSES  
TELLEMENT  
INTELLIGENTES  
QUE, LE PLUS  
SOUVENT, JE  
COMPRENDS PAS  
CE QUE  
JE DIS.

