

OpenStack 101

Sébastien Badia

`sebastien.badia@enovance.com`

eNovance – 2 avril 2014



Outline

- 1 Cloud computing
- 2 Openstack
- 3 Technical overview
- 4 Deploy, learn and tips

Cloud computing

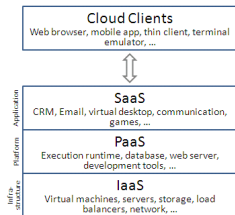
Definition

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources.

- ▶ **On Demand** : Resources are dynamically created.
- ▶ **Multi-tenant** : Resources are shared between users.
- ▶ **Broad network access** : Network, standard mechanisms
- ▶ **Elasticity** : Infrastructure is flexible (grow/reduce).
- ▶ **Measured service** : Users pay what they use.

Cloud computing

- ▶ Service models by the NIST¹ (24 July 2011)
 - ▶ Infrastructure as a service (IaaS)
 - ▶ Platform as a service (PaaS)
 - ▶ Software as a service (SaaS)
- ▶ XaaS, a comprehensive taxonomy model
 - ▶ Database as a service (DaaS)
 - ▶ Network as a service (NaaS)
 - ▶ ...



Outline

- 1 Cloud computing
- 2 Openstack
- 3 Technical overview
- 4 Deploy, learn and tips

OpenStack

- ▶ Infrastructure as a service (**laaS**) cloud middleware
- ▶ Open Source software (***Apache License***)
- ▶ Derived from **Nebula** (*NASA*) and **Cloud Files** (*Rackspace*)
- ▶ Written in **Python**
- ▶ Stable release : ***Havana*** (*October 13, 2014*)

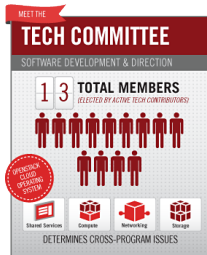


OpenStack Foundation

- ▶ More than **9500 individual members**
- ▶ **100 countries**
- ▶ **850 different organizations**
- ▶ Secured more than **\$10 million** in funding
- ▶ Managed by 3 committee

Technical Committee

- ▶ TC manage software development and direction
- ▶ 13 members elected by active contributors
- ▶ Each OpenStack project has a PTL²
- ▶ 5 are directly elected, and 8 are PTL
- ▶ Enforce OpenStack ideals (Openness, Transparency, Commonality, Integration, Quality)



2. Project Technical Leader

Board of Directors

- ▶ Protect, promote and empower OpenStack
- ▶ 24 members to provides strategic and financial oversight of Foundation
 - ▶ 8 platinum (appointed by members)
 - ▶ 8 gold (elected by member class) (eNovance 😊)
 - ▶ 8 individual (elected by individual members)
- ▶ Led by Alan Clark (Suse)



User Committee

- ▶ User advocacy and feedback, anybody can join
- ▶ Represent a broad set of enterprise, academic and service provider users
- ▶ Led by Tim Bell (CERN)

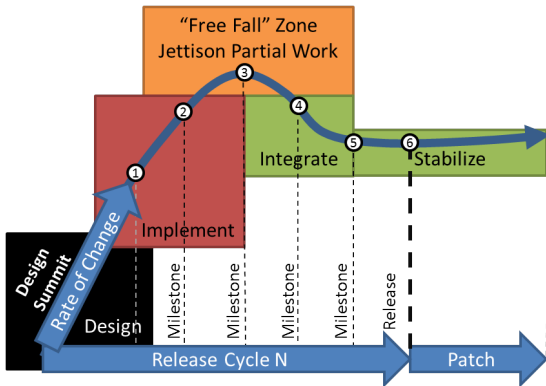


Community

- ▶ **Core** developers (elected) Grant for merging into master
- ▶ Anyone can contribute
 - ▶ Code
 - ▶ Documentation
 - ▶ Support (irc, mail)
 - ▶ ...
- ▶ <https://www.openstack.org/community/>



Releases cycle



- ▶ A release every 6 months
- ▶ Alphabetical order 😊
- ▶ <https://wiki.openstack.org/Releases>

Outline

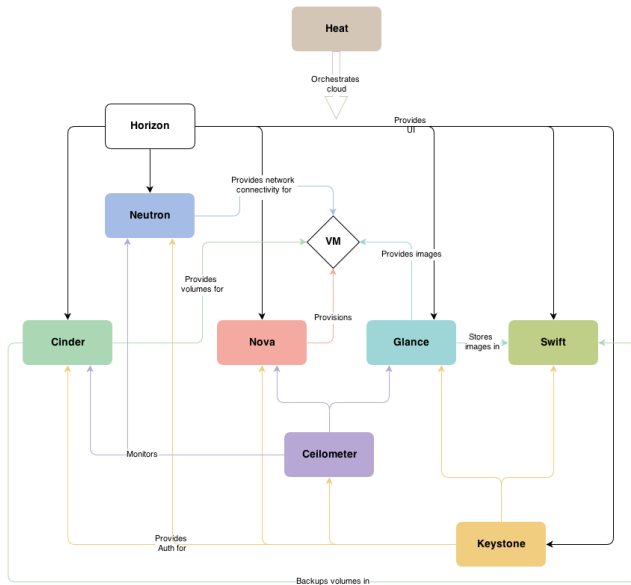
- 1 Cloud computing
- 2 Openstack
- 3 Technical overview**
- 4 Deploy, learn and tips

OpenStack IaaS

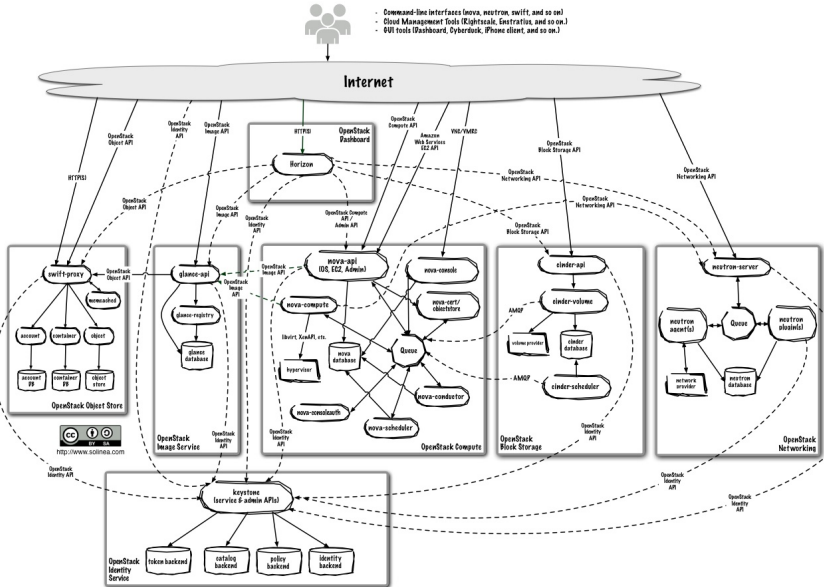
IaaS component vs. OpenStack component

Compute	⇒	Nova
Images	⇒	Glance
Identity	⇒	Keystone
Storage	⇒	Swift (object), Cinder (block)
Networking	⇒	Neutron
Dashboard	⇒	Horizon
Telemetry	⇒	Ceilometer
Orchestration	⇒	Heat

OpenStack conceptual architecture



OpenStack logical infrastructure



OpenStack Compute (Nova)



- ▶ Provision and manage virtual machines
- ▶ Hypervisor support : **XEN/XCP**, **KVM**, QEMU, LXC, ESX, ESXi³
- ▶ OpenStack API (Compute API, Rackspace Cloud Server API)
- ▶ Support **live migration** (need a shared storage for instances) (*Ceph*, *Swift*, *NFS*⁴, *GlusterFS*⁵)
- ▶ Bare-metal, Cells

3. <http://wiki.openstack.org/HypervisorSupportMatrix>

4. <http://docs.openstack.org/trunk/openstack-compute/admin/content/configuring-live-migrations.html>

5. <http://gluster.org/community/documentation/index.php/OSConnect>

Nova Scheduler



Nova-scheduler implements a few basic scheduling algorithms

- ▶ **Simple** : hosts whose **load is least** are chosen to run the instance. The load information may be fetched from a load balancer
- ▶ **Chance** : a compute host is chosen **randomly** across **availability zones**
- ▶ **Zone** : Similar to chance, but the compute host is chosen **randomly** from within a **specified zone**



- ▶ Compute scheduler support **filtering** and **weighting** to make informed decisions⁶
- ▶ nova.scheduler.filters (core,compute,ram,cidr,different/same host)⁷

6. <http://ibm.co/LUvm2n>

7. http://nova.openstack.org/devref/filter_scheduler.html

OpenStack Object storage (Swift)

- ▶ Object storage (Swift) : Redundant (Object/DB)⁸ and scalable
 - ▶ Storage via an API (**not a FS**)
 - ▶ **Long-term** storage system for **large amounts** of data
 - ▶ Storage abstraction (Ring concept⁹, zone and weight of storage)
 - ▶ Works with auth **token and HTTP API (RESTFull)**
 - ▶ **Similarity with Amazon S3 (bucket)**



8. http://swift.openstack.org/overview_replication.html

9. http://swift.openstack.org/overview_ring.html

OpenStack Image service (*Glance*)

- ▶ Image service (Glance) : **Catalog** and **manage library** of **server images**
 - ▶ Interaction between Nova-compute and Swift or Ceph
 - ▶ Image format
 - ▶ Container (bare, **ovf**, aki, ari, ami)
 - ▶ Disk (raw, vhd, vmdk, vdi, iso, **qcow2**, aki, ari, **ami**)¹⁰
 - ▶ Manageable by a CLI or an API Rest
 - ▶ **Image Store (Ceph, Swift, FS, S3)**
 - ▶ Glance support image caching¹¹



10. <http://glance.openstack.org/formats.html>


11. <http://glance.openstack.org/cache.html>

OpenStack Block storage (*Cinder*)

- ▶ Enables management of volumes, volume snapshots, and volume types
- ▶ Support : **RBD**, **iSCSI**, Sheepdog, AoE, LeftHand
- ▶ Similar to Amazon EBS



OpenStack Identity management (*Keystone*)

- ▶ Provide an **unified authentication** across **all openstack projects**
- ▶ Keystone concepts (*User managment*)
 -  ▶ **Users** : a human user (login,password,email)
 - ▶ **Tenants** : a group of users (project or organization)
 - ▶ **Roles** : determine what operations an user is permitted to perform in a given tenant
- ▶ Keystone manage also services (services,endpoint,catalog)
- ▶ OpenStack's **Identity API** (XML/JSON API)
- ▶ Can be backed by LDAP

OpenStack Network (*Neutron*)

- ▶ Provides networking connectivity to VMs
- ▶ **Manage network** (L2 and L3) with a Rest API
- ▶ Networking backend by **plugins** :
 - ▶ Open-vSwitch
 - ▶ Linux Bridge
 - ▶ OpenFlow
 - ▶ Floodlight
 - ▶ ...

OpenStack Telemetry (*Ceilometer*)

- ▶ Provide **efficient** collection of **metering data**, (CPU and network costs)
- ▶ Custom data by **plug-ins**.
- ▶ Produces **signed** metering messages that cannot be repudiated

OpenStack Orchestration (*Heat*)

- ▶ Provide a **template-based** for describing a cloud application.
- ▶ Integrated with **all** OpenStack ressources
- ▶ Provide advanced features (**ha**, **auto-scaling**, . . .)
- ▶ REST API, compatible with AWS CloudFormation

OpenStack Dashboard (*Horizon*)

Launch Instances

Server Name

grid5000_ocw

User Data

OpenCloudWare, Grid5000 Demo (sophia 21/06/2012)

Description:

Specify the details for launching an instance. The chart below shows the resources used by this project in relation to the project's quotas.

Project Quotas

Instance Count (1)

9 Available

VCPUs (1)

19 Available

Disk (0 GB)

1000 GB Available

Memory (512 MB)

50688 MB Available

Flavor

m1.tiny (1VCPU / 0GB Disk / 512MB Ram)

Size of image to launch.

Keypair

key_cirros

Instance Count

1

Security Groups

☒ cirros_test

☐ default

Cancel

Launch Instance

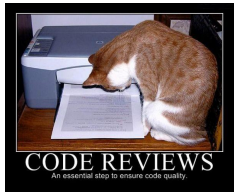
- ▶ Access and provision **cloud resources** through a web portal
- ▶ **Credentials, users** and **projects** management
- ▶ **Django module** for **easy** integration/creation



Outline

- 1 Cloud computing
- 2 Openstack
- 3 Technical overview
- 4 Deploy, learn and tips**

Development



- ▶ Integrated with launchpad (bug, milestone, cas)
- ▶ Code-review <https://review.openstack.org/>
 - ▶ Git-review¹² (patch set lists / project setup / ease submit)
 - ▶ In console, fgerrit¹³
- ▶ Status <http://status.openstack.org/zuul/>
- ▶ Docs <http://wiki.openstack.org>
- ▶ IRC (oftc and freenode)

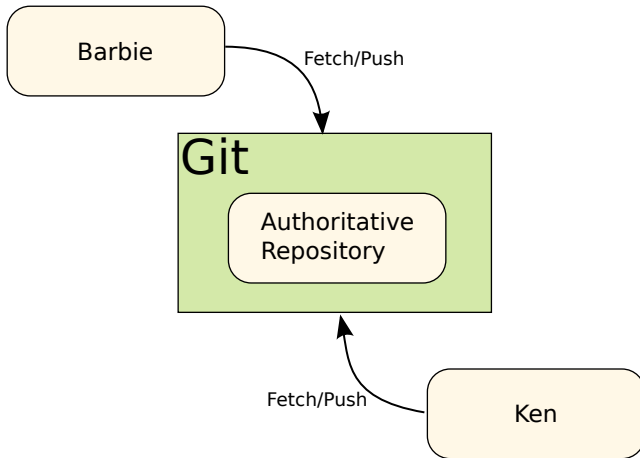


12. <http://www.mediawiki.org/wiki/Gerrit/git-review>

13. <https://github.com/pandemicsyn/fgerrit>

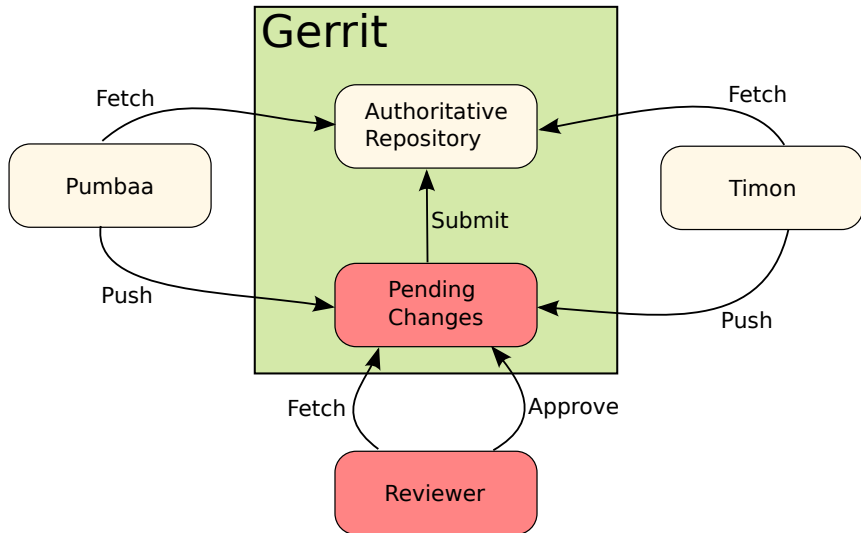
Git vs. Gerrit

Git workflow



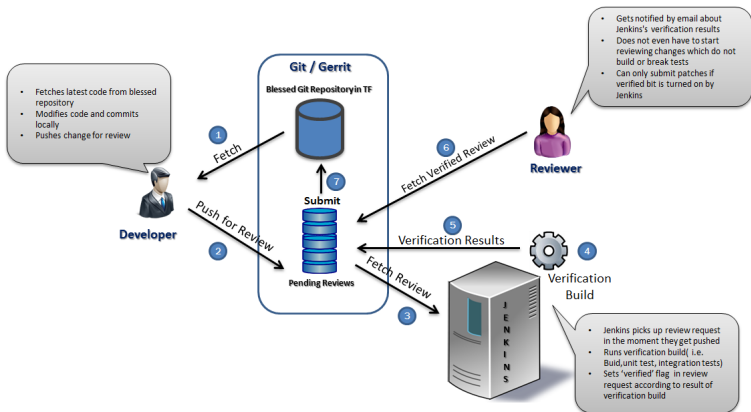
Git vs. Gerrit

Gerrit workflow



Gerrit and CI workflow

Git / Gerrit Work Flow with Jenkins Continuous Integration



Courtesy of <http://blogs.collab.net/teamforge/teamforge-git-gerrit-integration-with-jenkins-ci>

Let's start



- ▶ Documentation, démo, videos
 - ▶ https://wiki.openstack.org/wiki/Main_Page
 - ▶ <http://docs.openstack.org/>
- ▶ Source code
 - ▶ <https://github.com/openstack>
 - ▶ <https://github.com/stackforge>
- ▶ Development environment
 - ▶ <http://devstack.org/>

Sources, code

- ▶ Images and sources
 - ▶ `http://ken.pepple.info/`
 - ▶ `http://openstack.org/`
- ▶ Slides `https://github.com/sbadia/slides`

OpenStack 101

Sébastien Badia

`sebastien.badia@enovance.com`

eNovance – 2 avril 2014

