Intel IT OpenStack Practice

Shuquan Huang, Intel IT Engineer Computing

Intel Information Technology

Agenda

- About Me
- Intel with OpenStack
- The way we do Cloud
- OpenStack Practice
- Integration with Intel Validation Cloud
- Q & A

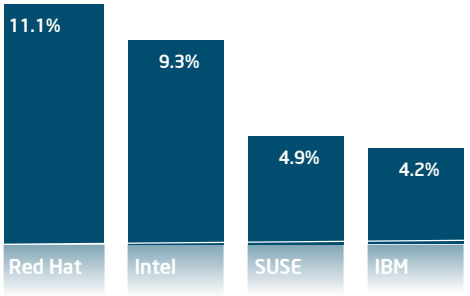
Huang, Shuquan

- Join Intel since 2011
- Software Engineer, Certified Scrum Master
- Leads Intel IT Engineer Computing OpenStack Team
- Focus on Cloud Solution and Agile

Coming from Intel IT Engineer Computing

- Focus on Intel private cloud solution – Intel Validation Cloud.
- Start OpenStack journey from 2011.

Helping Fuel Innovation—and Opportunities



#2 Linux Contributor

improving performance, stability & efficiency

X.org GNU Eclipse

Webkit JQuery kernel.org

Yocto Project OpenStack

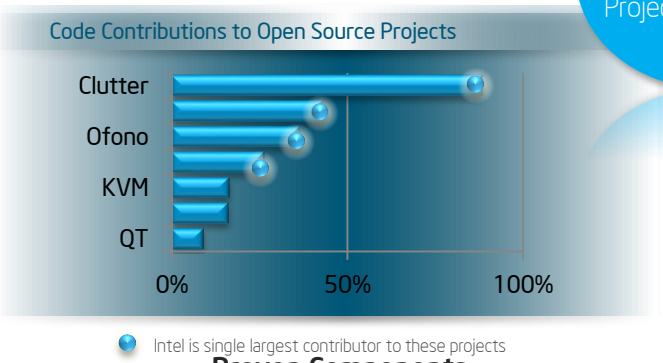
01.org Hadoop

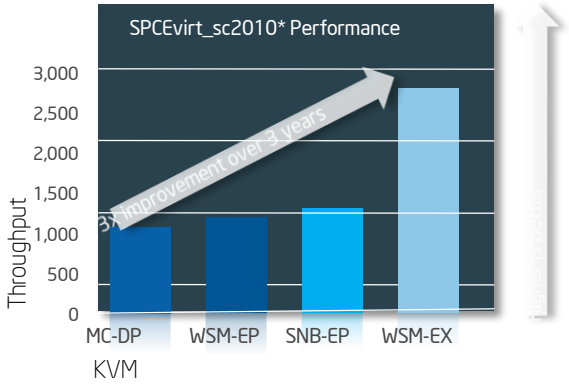
Across the Stack

contributions span every layer of the stack

Intel in Open Source

Project Contributor





Proven Components

building blocks simplify development, reduce costs and speed time-to-market

Intel® Enables OpenStack* Cloud Deployments

Contributions

- Across OpenStack projects
- Open Source Tools
- Top 10 contributor to Grizzly and Havana releases[1]
- Optimizations, validation, and patches

Intel® IT Open Cloud

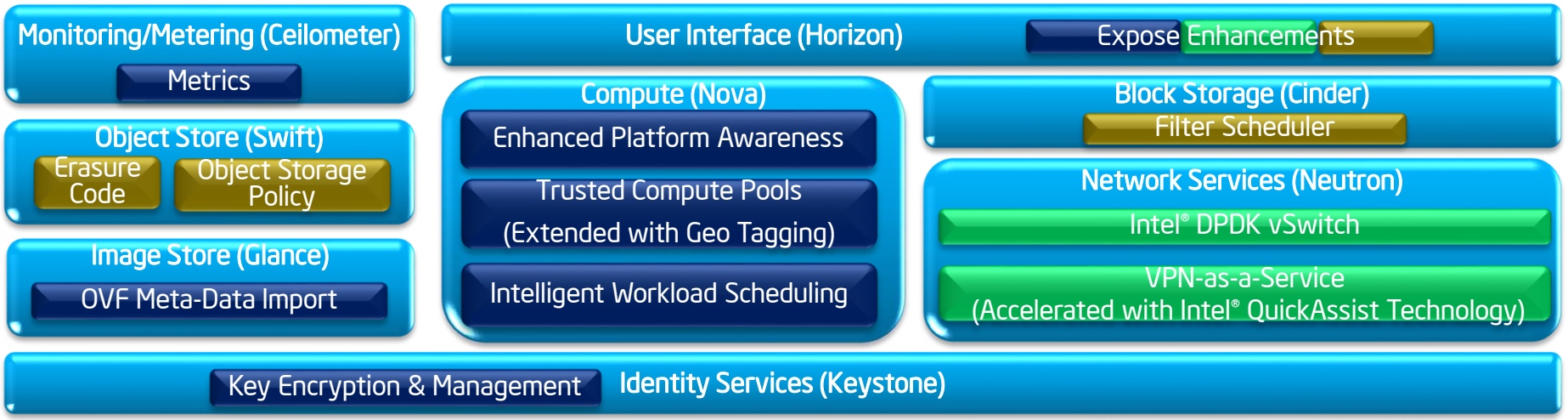
- Intel IT Open Cloud with OpenStack
- Deliver Consumable Services
- Automated Management of Cloud

Intel® Cloud Builders

- Collection of best practices
- Intel IT Open Cloud Reference Arch
- Share best practices with IT and CSPs
- http://www.intel.com/cloudbuilders

[1]Source: www.stackalytics.com

Intel Contributions* to OpenStack

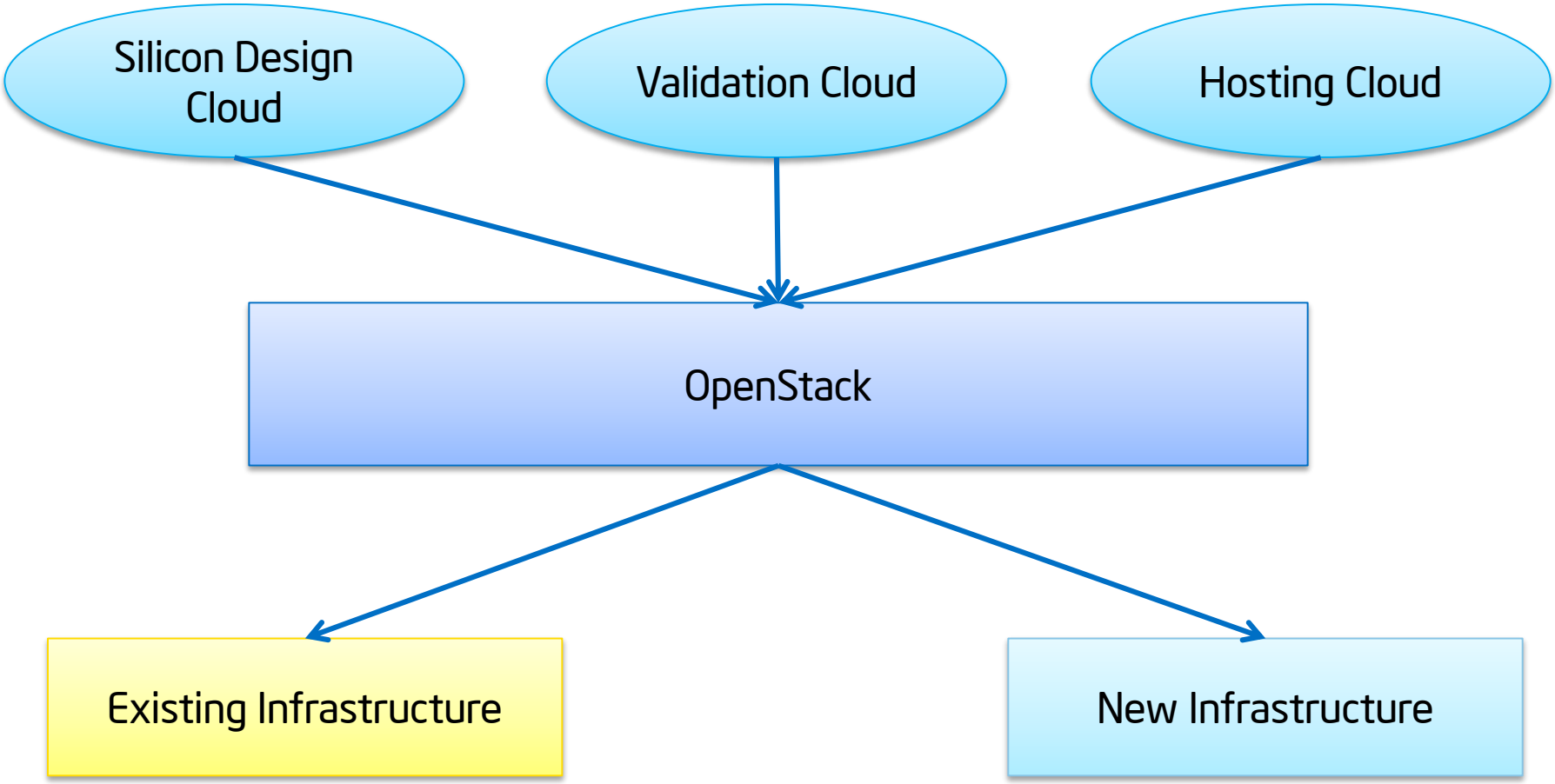


Compute		Networking	Storage

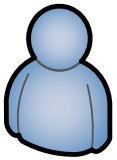
• Enhanced Platform Awareness
  • CPU Feature Detection
  • PCIe SR-IOV Accelerators
  • OVF Meta-Data Import
	• Trusted Compute Pools
	  • With Geo Tagging
	• Key Management
	• Intelligent Workload Scheduling (Metrics)
		• Intel® DPDK vSwitch
		• VPN-as-a-Service with Intel® QuickAssist Acceleration
			• Filter Scheduler
			• Erasure Code
			• Object Storage Policies

Note: A mixture of features that are completed, in development or in Planning (not PoR)

OpenStack – Intel IT Convergence Platform



The Way we do Cloud



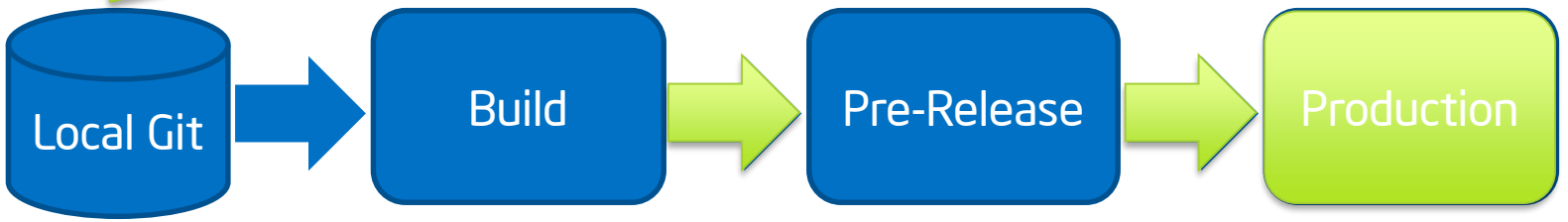
Our Developers

Snapshot
Improvement

Bug Fixed &
new Features

Deployment Solution

Cloud Data Analysis

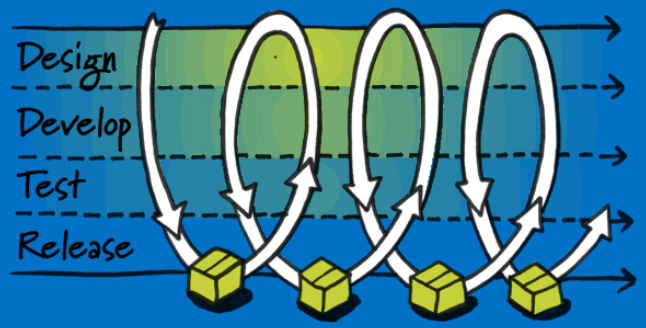


Tempest Launcher

One-Stop-Shop
Admin Portal

Continuous Delivery

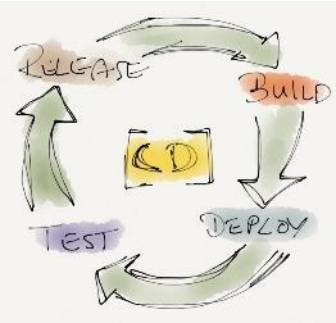
Continuous Delivery (CD) is a pattern language used in software development to automate and improve the process of software delivery. - wiki



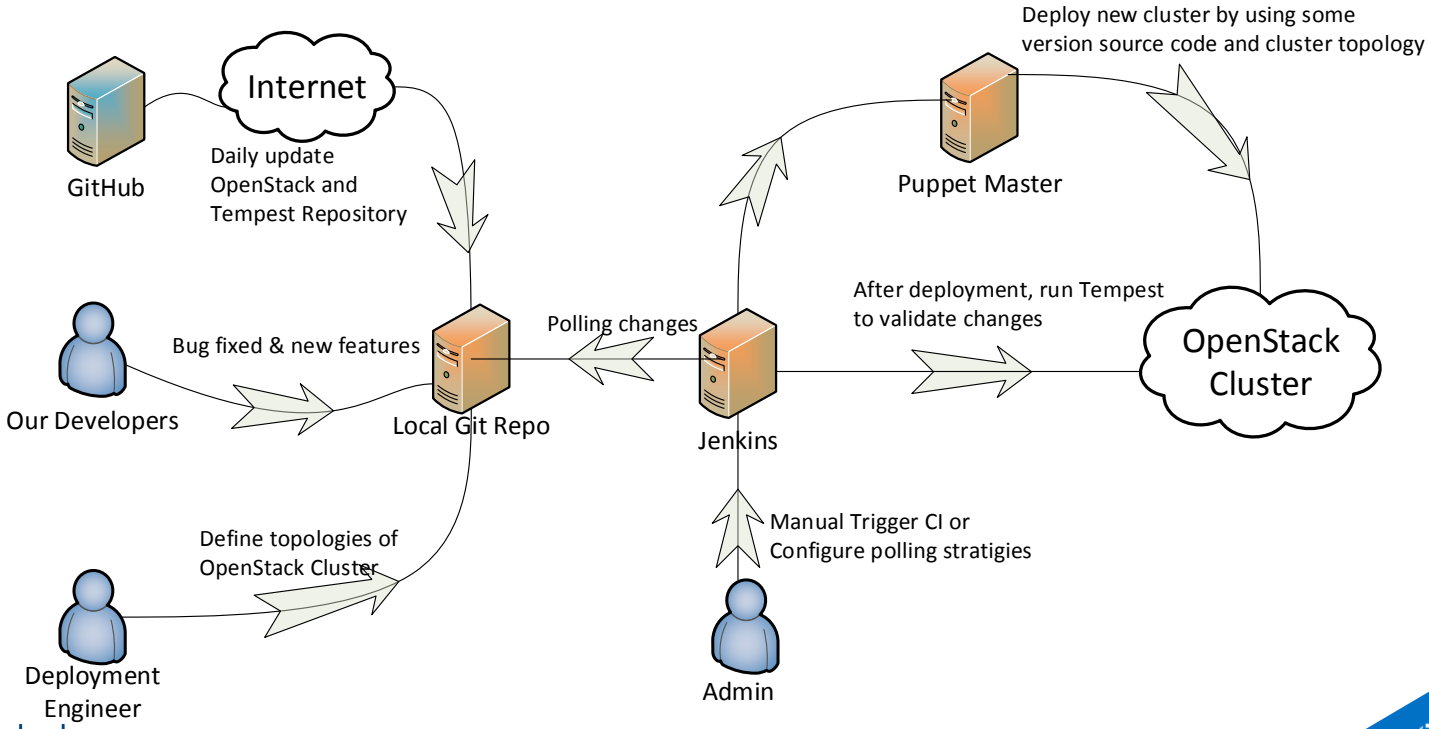
Problem Statement

- Configuration is easy to lost.
- It takes long time to get the feedback after developers make some customized changed in OpenStack.
- Track and taste the latest update from community.
- Automated run tests after each code changed.

Solution



- Refer to http://ci.openstack.org/ and build our own system.
- Configuration as Code.
- Version Control: Code, Configuration, Environment and anything else.



Challenges

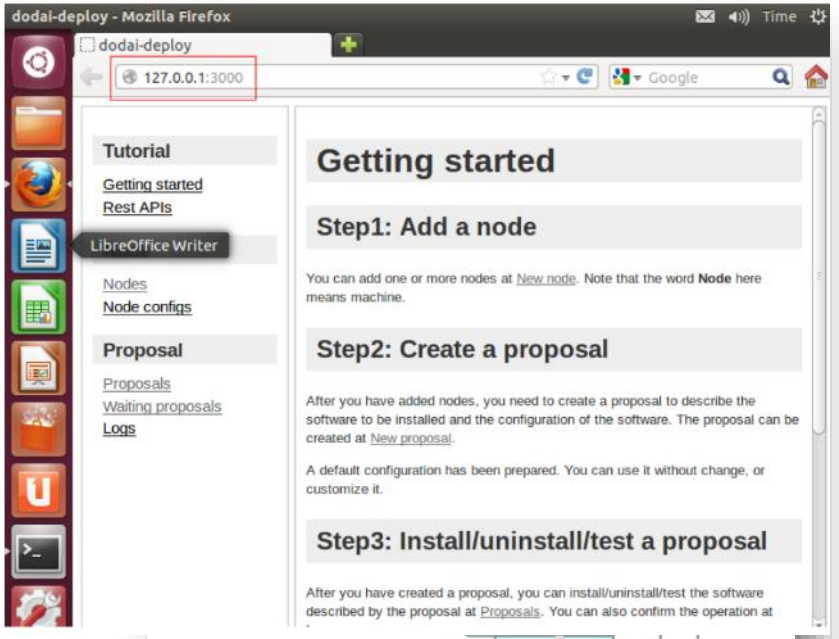
- Installation by source code.
- Environment cleanup before each build.
  - Live upgrade
  - Rebuild totally
- Configuration as Code

Deployment Solution

Rapid deployment different topologies OpenStack from bare metal

Deployment Solution

- Open Source Solution:
  - PXE boot up
  - puppet/dodai deployment
- Fast Deployment
  - Deploy full stack of OpenStack on mult nodes bare-metal with several mins.
- Auto deployment:
  - Only one 4G U-Disk is needed



Tempest Launcher

Have tempest more stable and visible to verify cloud functionalities

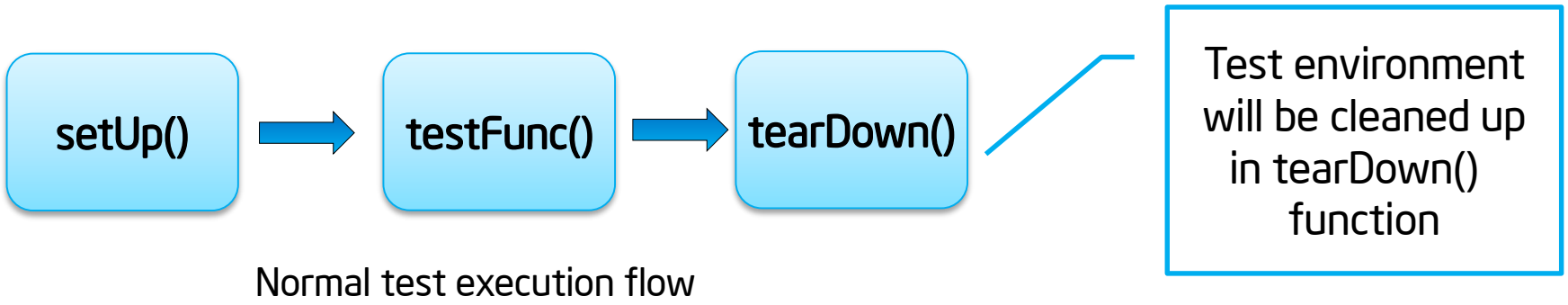
Problem Statement & Objective

- Problem Statement:
  - Result is not user unfriendly and exception may happen during running tempest scripts in batch directly.

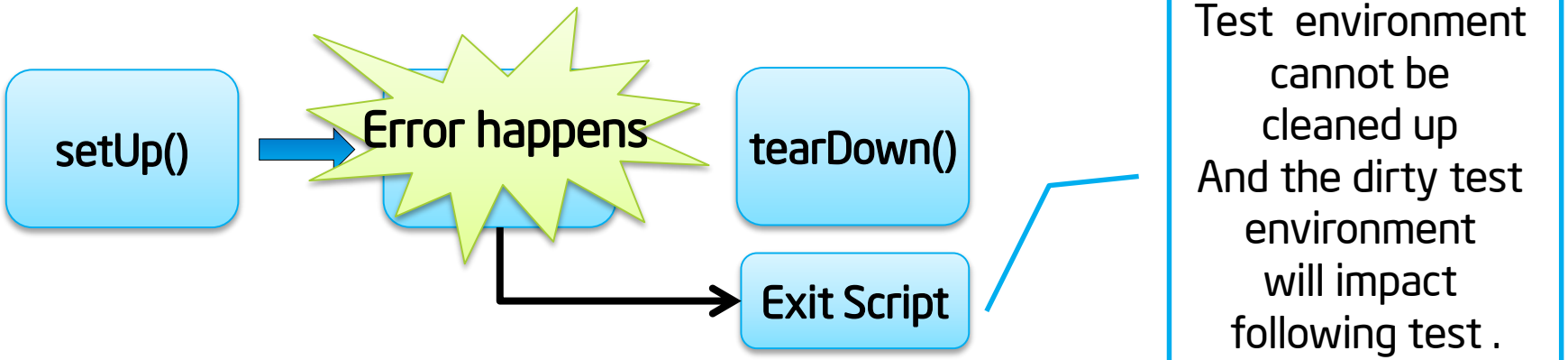
- Objective
  - Address the problems mentioned above.
  - Improve and enhance the capability of tempest automatic launching and handling test in batch execution.

Problem 1

Dirty test environment when previous test teardown function is skipped for unexpected errors



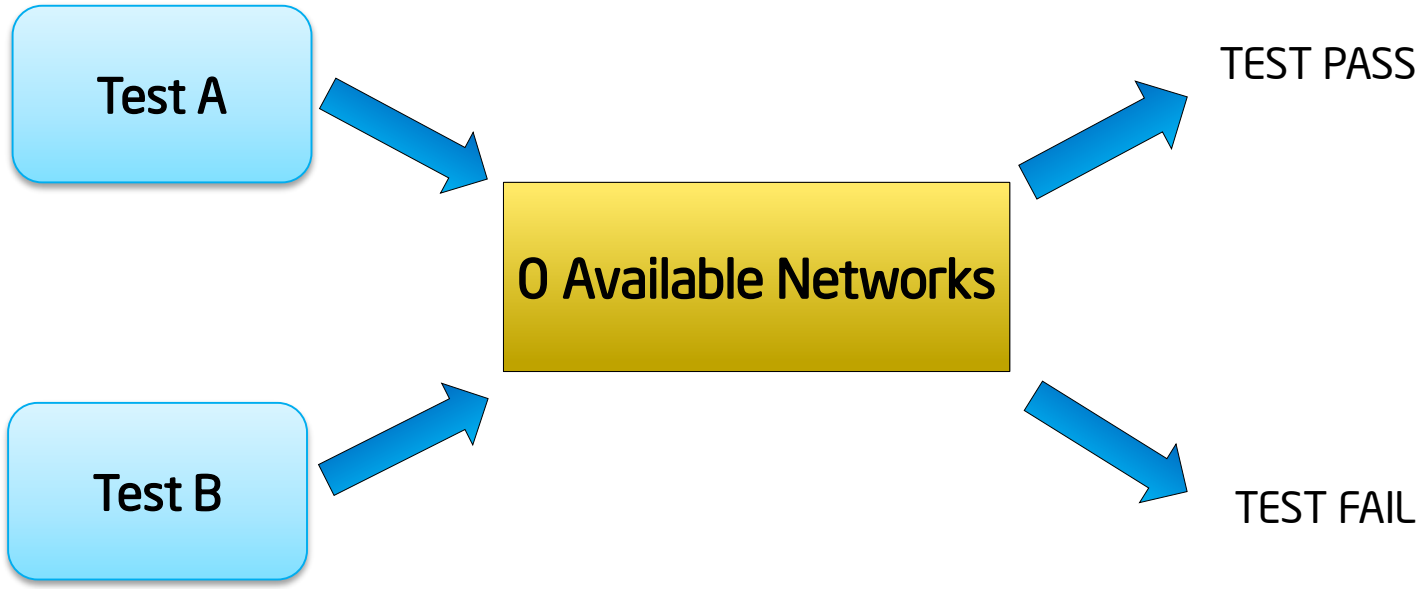
Normal test execution flow



Test execution flow with exception

Problem 2

Test script requires test pre-environment



Test A didn't release the networks from the test specific tenant.
Test B failed since no available network.

Problem 3

Cannot provide fully customized or prioritized test execution plan when run test in bunch. It's hard to run the tests across folders or scripts in one time.

Run the tests from 1 script:

```
root@hostname:~/tempest# nosetests -v tempest.tests.compute.test_images
```

Run the tests from 1 folder:

```
root@hostname:~/tempest# nosetests -v tempest.tests.compute
```

Problem 4

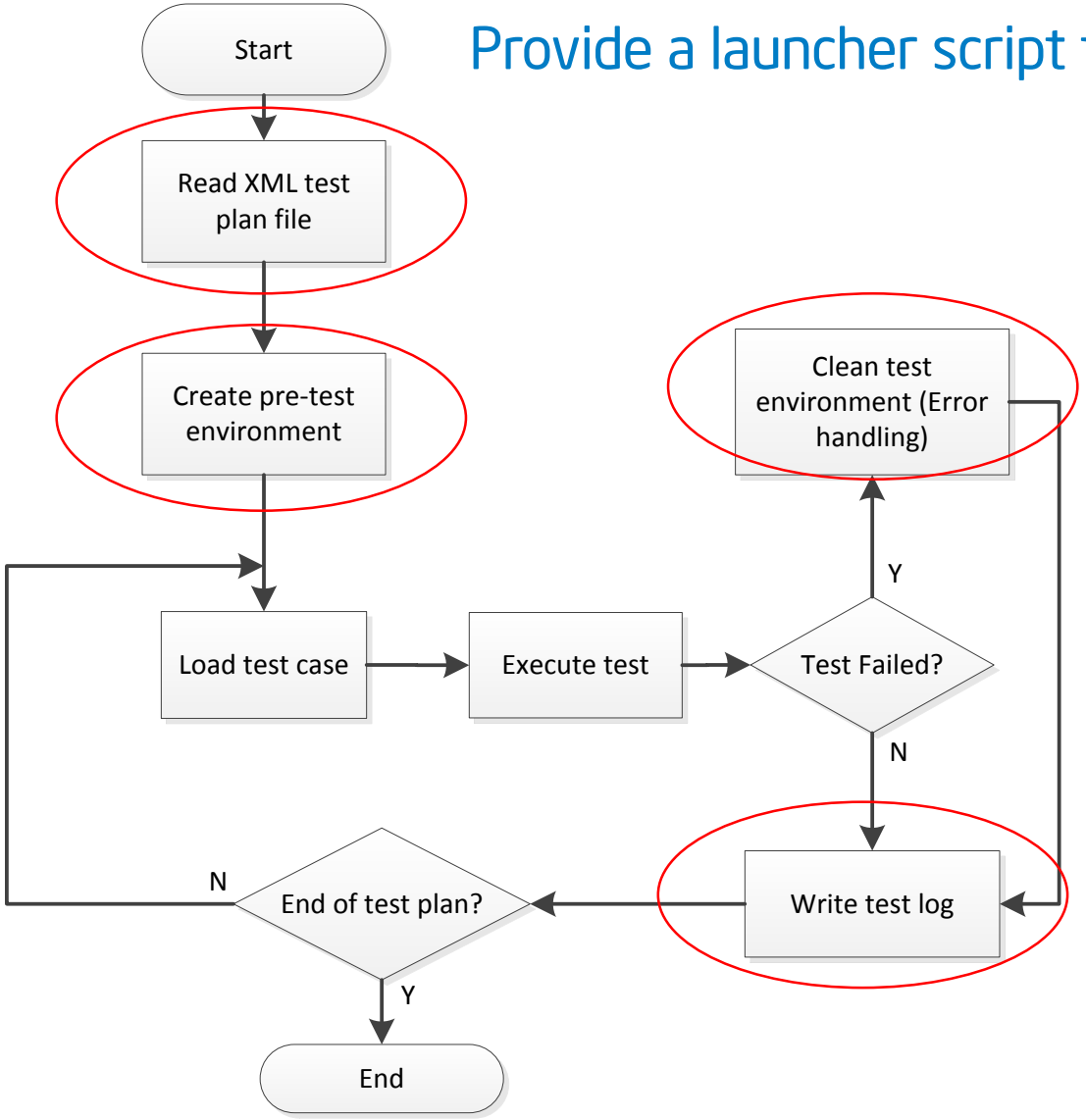
Test result and error information is separated. XML report with '--with-xunit' can't import as test report in excel directly.

```
An image for the provided server should be created ... ok
An image should not be created if the server instance is removed ... ERROR
An image should not be created with invalid server id ... ok
Return error when creating an image of a server that is building ... ok
Return an error when creating image of server that is terminating ... ERROR
Disallow creating another image when first image is being saved ... ok

======================================================================
ERROR: An image should not be created if the server instance is removed
----------------------------------------------------------------------
Traceback (most recent call last):
  File "/root/tempest/tempest/tests/compute/test_images.py", line 84, in
test_create_image_from_deleted_server
    self.servers_client.wait_for_server_status(server['id'], 'ACTIVE')
  File "/root/tempest/tempest/services/nova/json/servers_client.py", line 145, in
wait_for_server_status
    raise exceptions.BuildErrorException(server_id=server_id)
BuildErrorException: Server 3d7a3d34-7715-40ef-bcc5-6246d0cc2cbf failed to build and
is in ERROR status
```

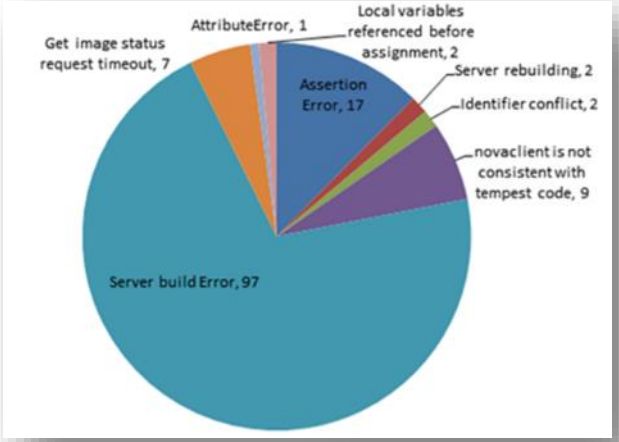
Solution

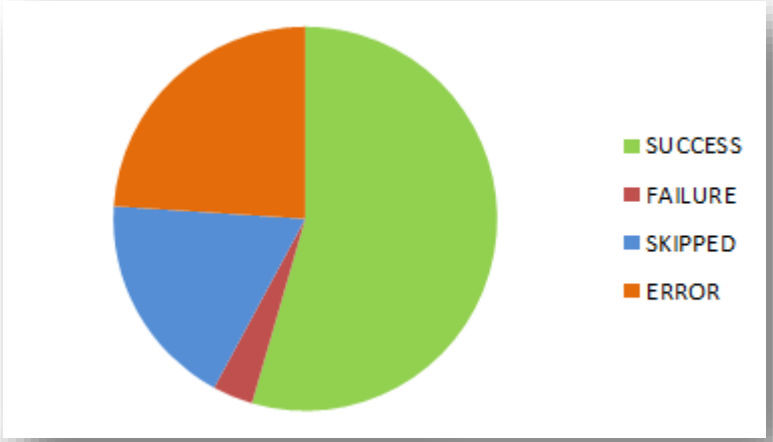
Provide a launcher script to address these problems



Result

- Customized test plan can be made on demand
- Minimize the impact of test failure to follow up test cases
- Excel report template for visual result





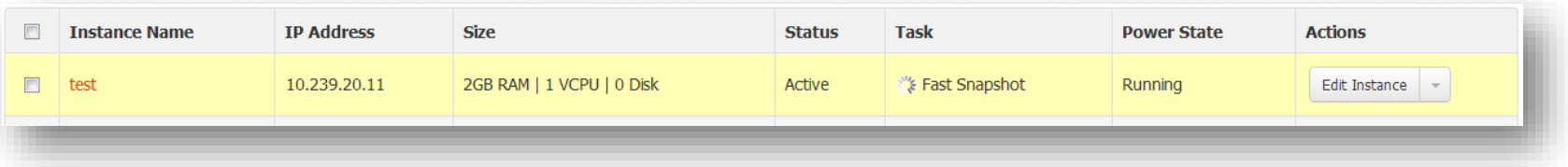
Total	Count	Percentage
SUCCESS	273	54.49%
FAILURE	17	3.39%
SKIPPED	91	18.16%
ERROR	120	23.95%
TOTAL	501	100%

Snapshot Improvement

Reduce the time of original OpenStack Snapshot

Problem Statement & Objective

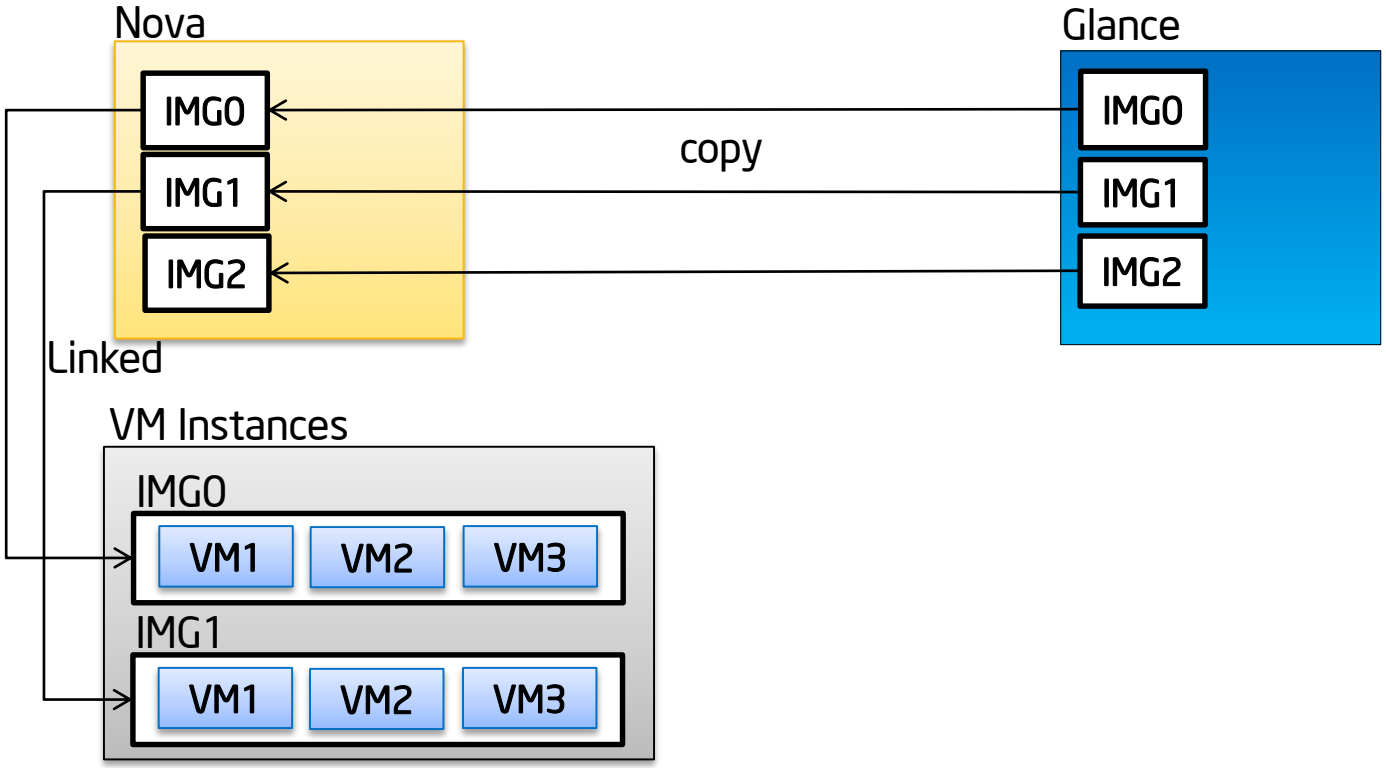
- Snapshot in OpenStack
  - Takes 20 mins for the snapshot of a 20GB VM
- Fast Snapshot in OpenStack
  - Self-customized feature
  - Takes less than 1 min for the snapshot of a 20GB VM



Solution

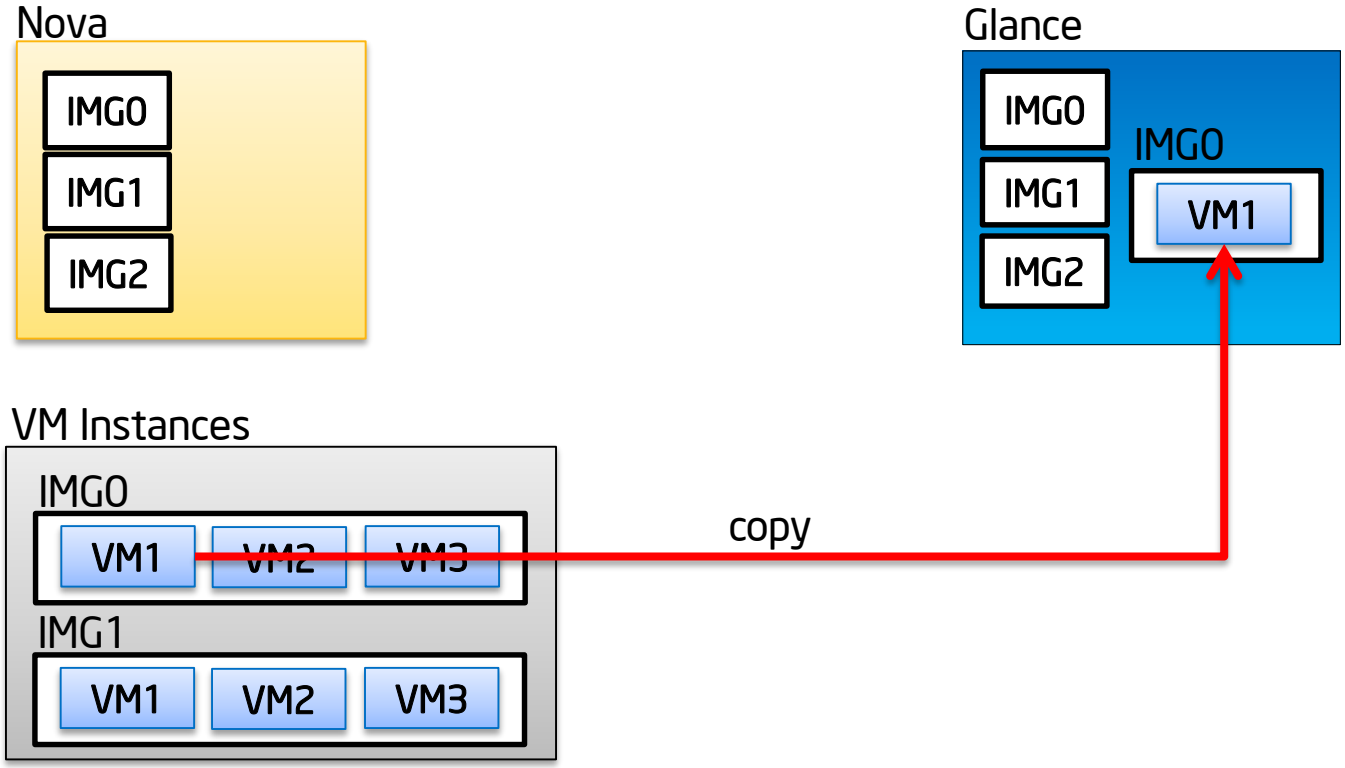
- Analysis each step of original snapshot.
- Reduce the time consuming stage.
- Optimized the workflow to fit our own requirement.

Fast Snapshot Diagram



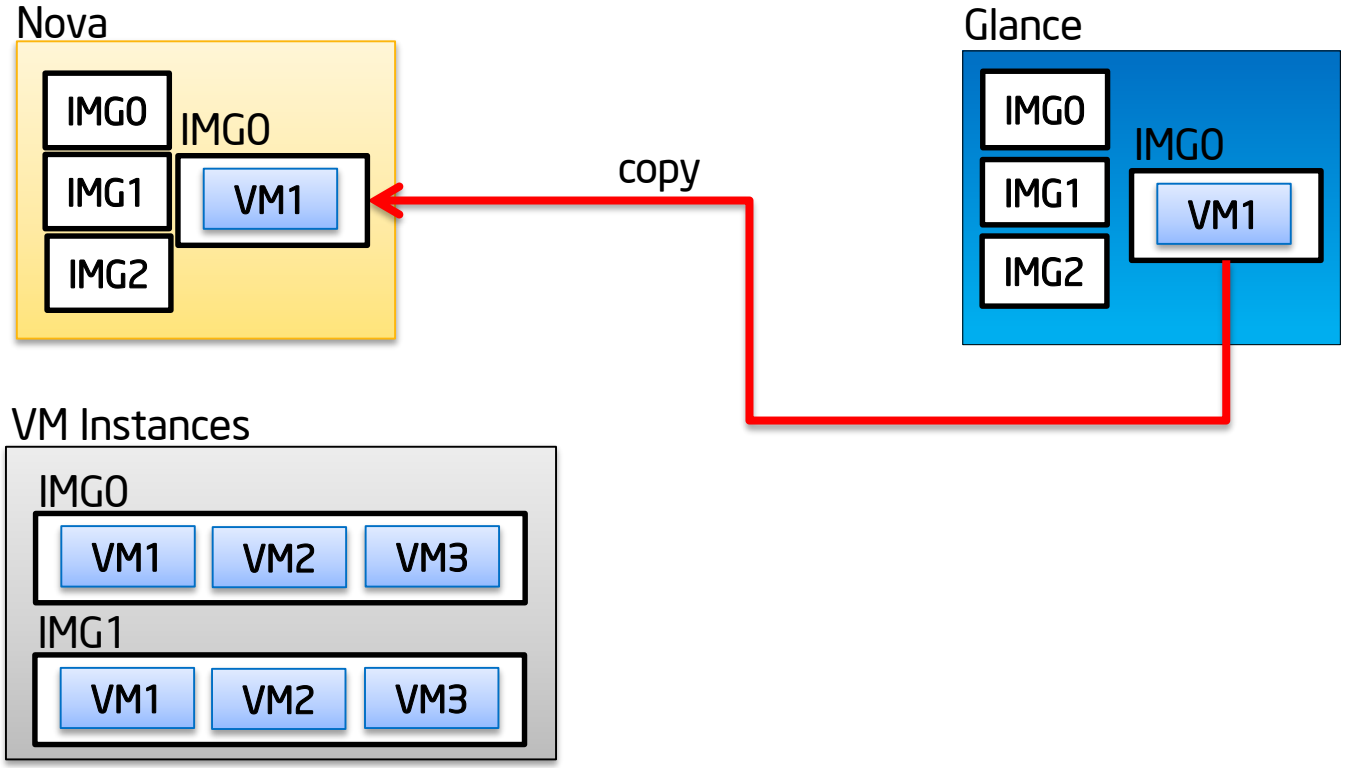
- This is the relation between VM Instances, Nova and Glance.
- When we launch an instance, the image from Glance, eg IMG0 will be copied into Nova.
- Then the nova will create a linked image as “disk” for VM Instances.

Fast Snapshot Diagram



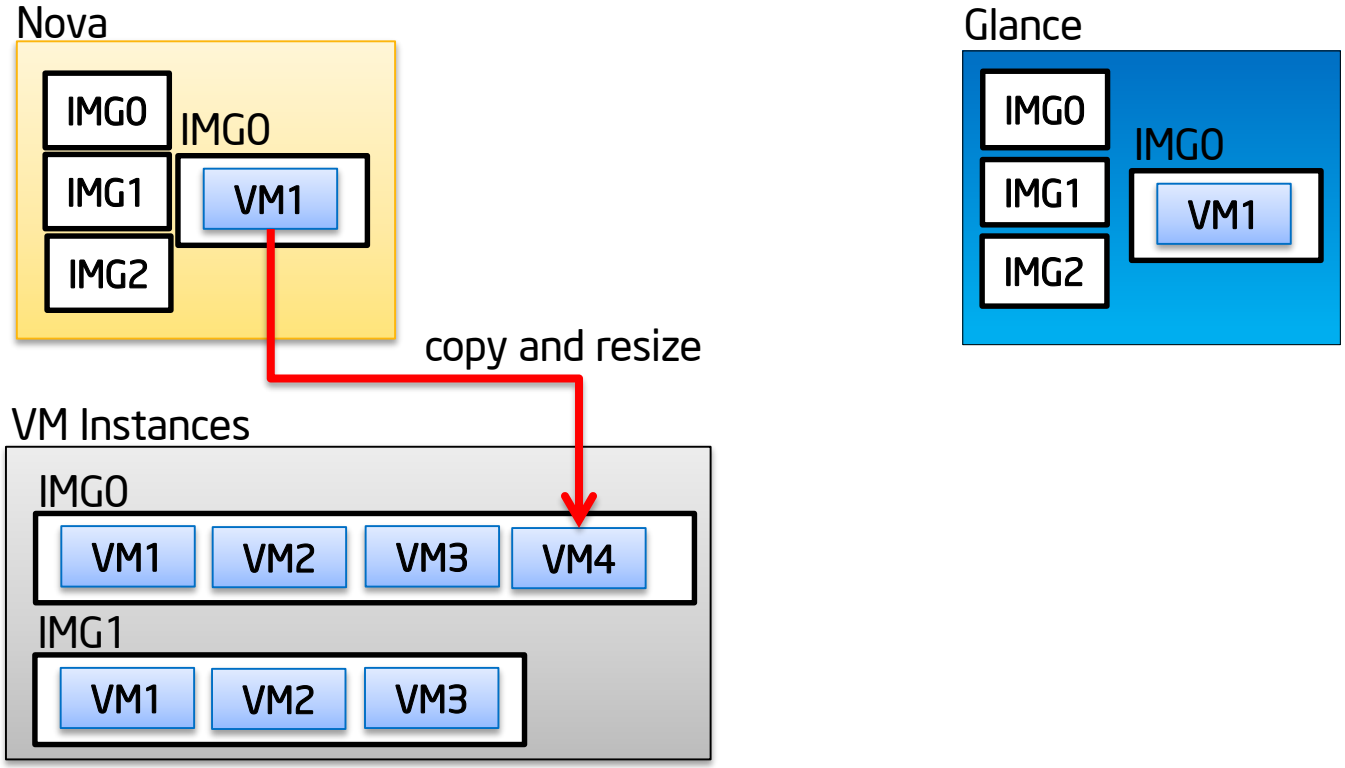
- When user run fast snapshot for VM1-IMG0, the instances "disk" will be copied and stored in Glance.
- The speed of this process is depend on the size of linked image. Size = VM1 which is smaller if we compare to the original snapshot, Size = VM1 + IMG0

Fast Snapshot Diagram



- When the user launch the created snapshot, the image from Glance will be copied into Nova.
- The speed of the copy is depending on the size of the image.

Fast Snapshot Diagram



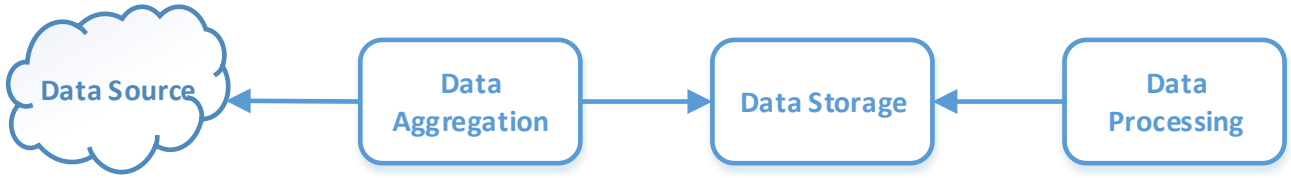
- The VM1 from the Nova will be copied into a newly created VM (VM4) instance folder as a “disk”.
- The resize of the instance also being done during this process if the user pick bigger “flavor” for their instance
- The speed in creating a new VM is depending on the size of IMG0-VM1

Cloud Data Analysis

Operational Intelligence in OpenStack: Listen to your cloud data

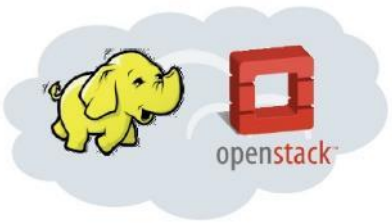
Problem Statement

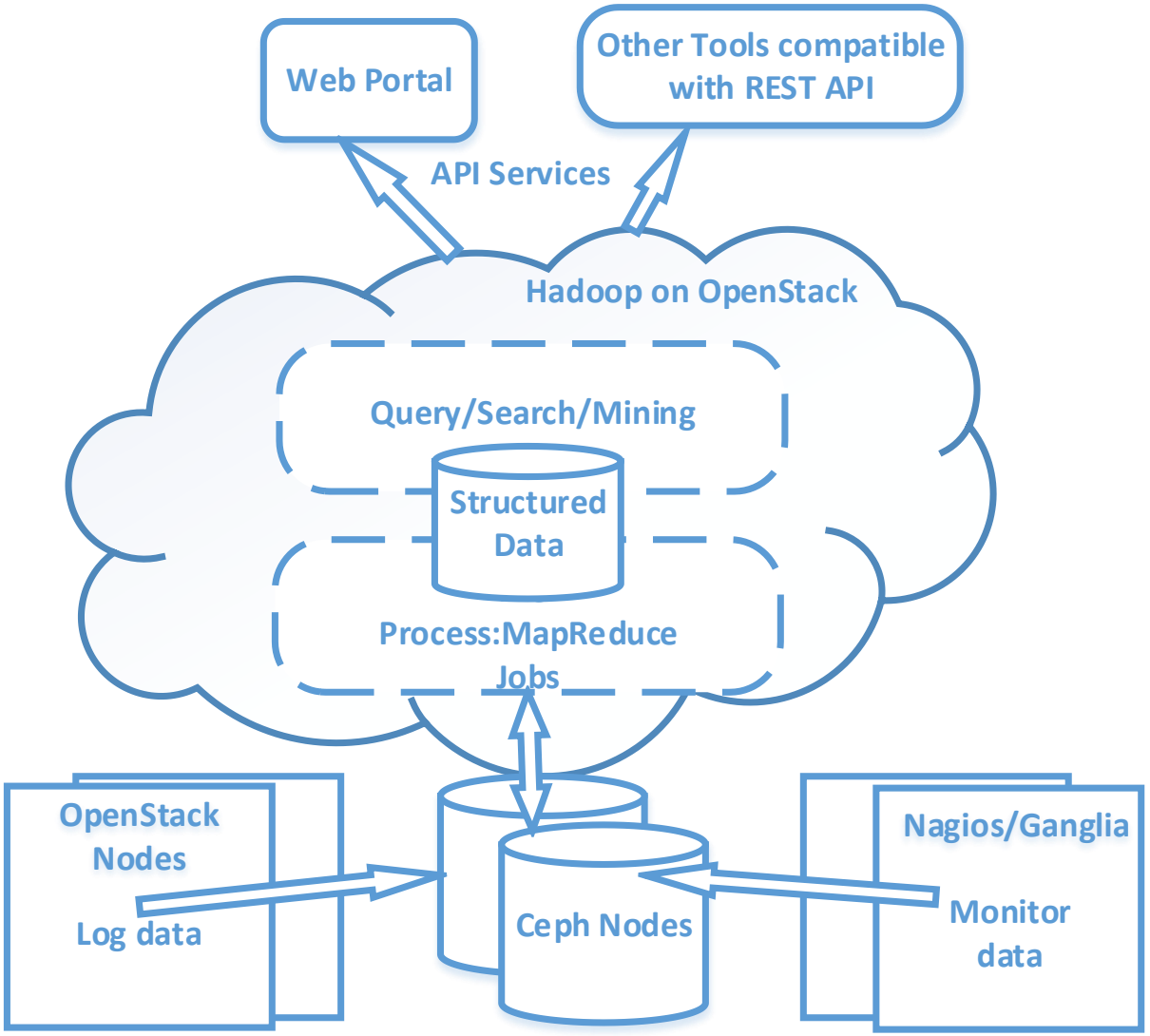
- Machine data provides insights into cloud status and plays a fundamental role in the cloud administration.
  - user behavior
  - machine behavior
  - security threats
  - system health
- Data are Separated.
- Data are too large to parse.
- Hard to query data in various format



Solution

- Data aggregation
  - CephFS
- Data storage
  - CephFS
- Data processing
  - MapReduce





Challenges

- Enable MapReduce in CephFS
  - Hadoop 1.1.x & Ceph 0.61.8
  - Add Hadoop-cephfs plugin to $HADOOP_HOME/lib
  - Create pool on Ceph MON Node for Hadoop use
- Log Mining
  - Construct the Vector Space
  - Clustering
  - Training

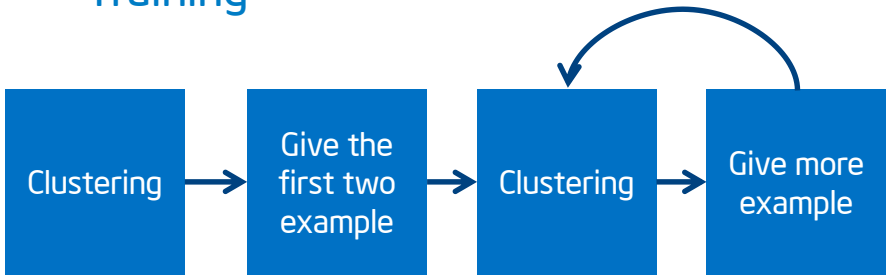


Table 1. Action Cluster

instance.create	instance.shutdown	network.create
instance.delete	instance.power_off	network.update
instance.rebuild	instance.power_on	network.delete
instance.resize.prep	instance.snapshot	subnet.create
instance.resize.confirm	instance.resize	subnet.update
instance.resize.revert	instance.finish_resize	subnet.delete
instance.exists	volume.create	port.create
instance.update	volume.delete	port.update
	volume.exists	port.delete

Result

- As an operational intelligence helper in our day-to-day cloud operation work help administrator analyze unreadable machine data and reduce a lot effort on understanding machine data.
- Because machine data aggregation, administrator can look into those data through one place and can see the connection easily.
- It helps a lot in the OpenStack cloud troubleshooting and increase the cloud utilization especially at midnight.

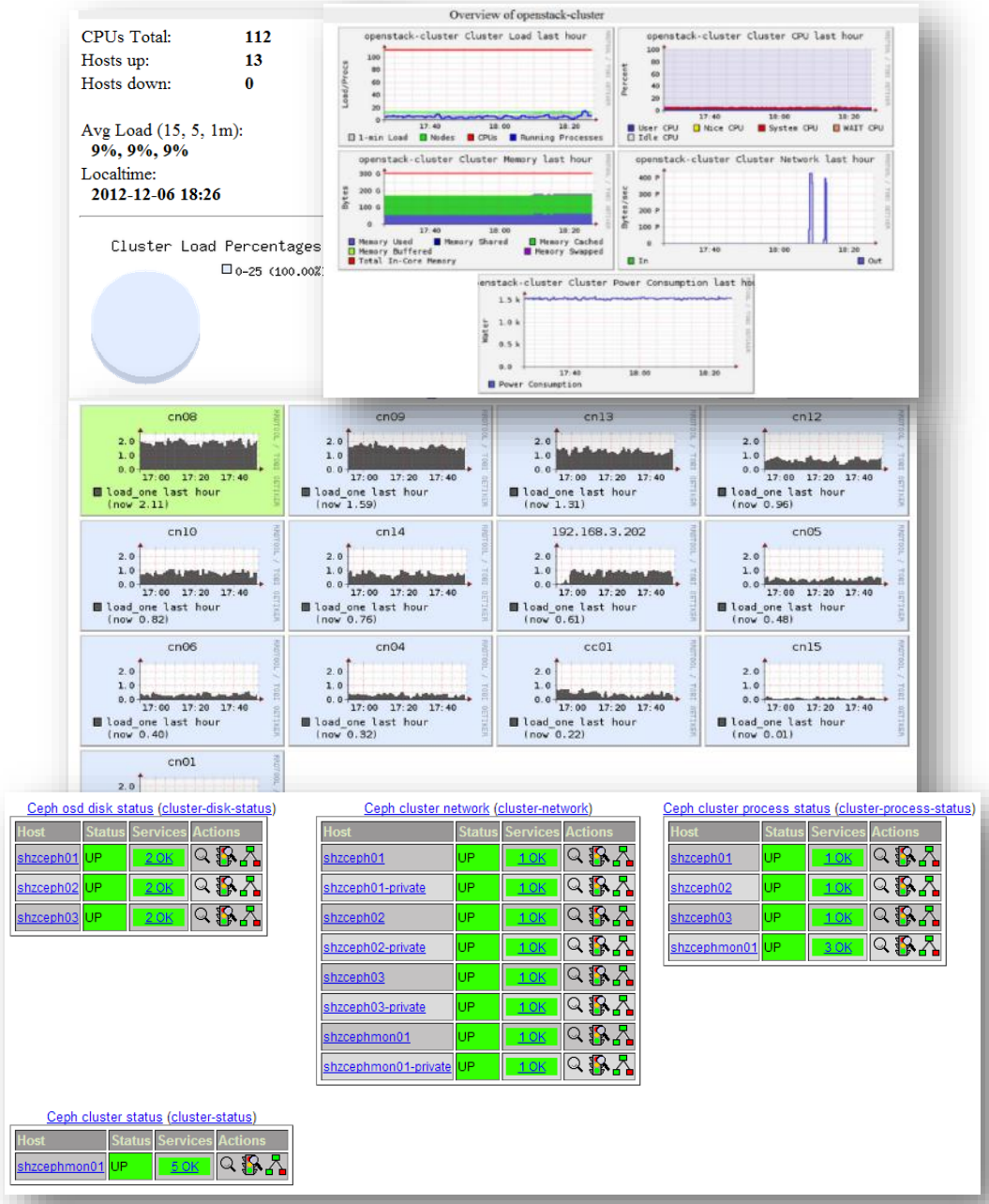
One-Stop-Shop Admin Portal

One-Stop-Shop Admin Portal

- VMs and Hosts status
- Cluster Monitor & Alarm
  - CPU, Memory, Disk, Network
- Log Browser
- Cloud Test
- Power Package Integration
  - Power Awareness Scheduler
  - Active Standby Management
  - Policy Based Migration

Monitoring & Alarm

- Cloud infrastructure monitoring on workload, CPU, memory utilization.
- Customized report to read the power consumption and CPU temperature metrics.
- Greatly help the operation team aware the healthy status of the cloud.

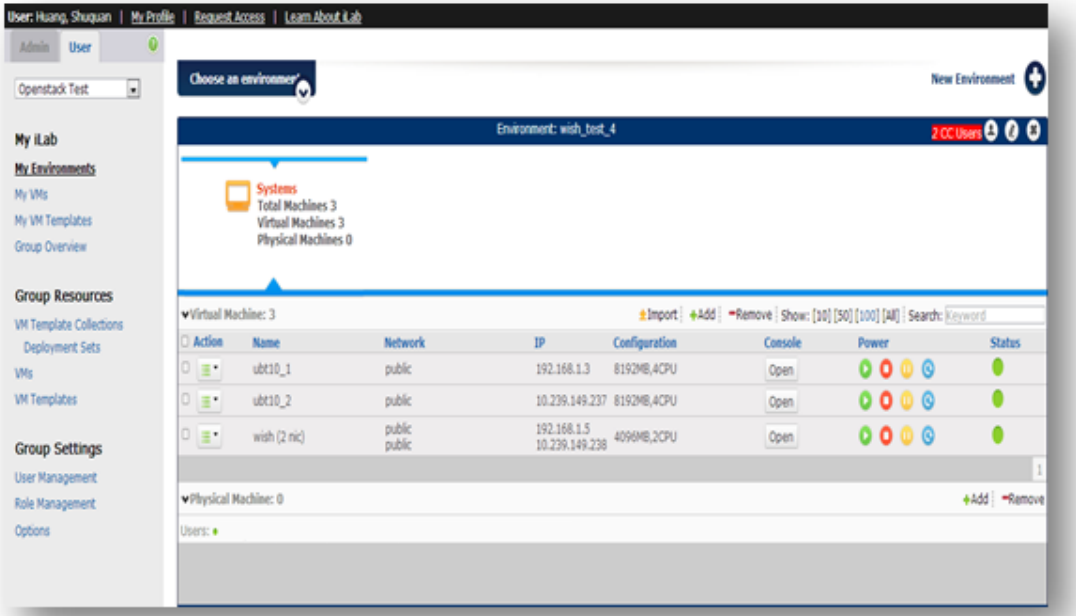


Integration with Intel Validation Cloud

Leveraging existing system and provide user the consistent experience

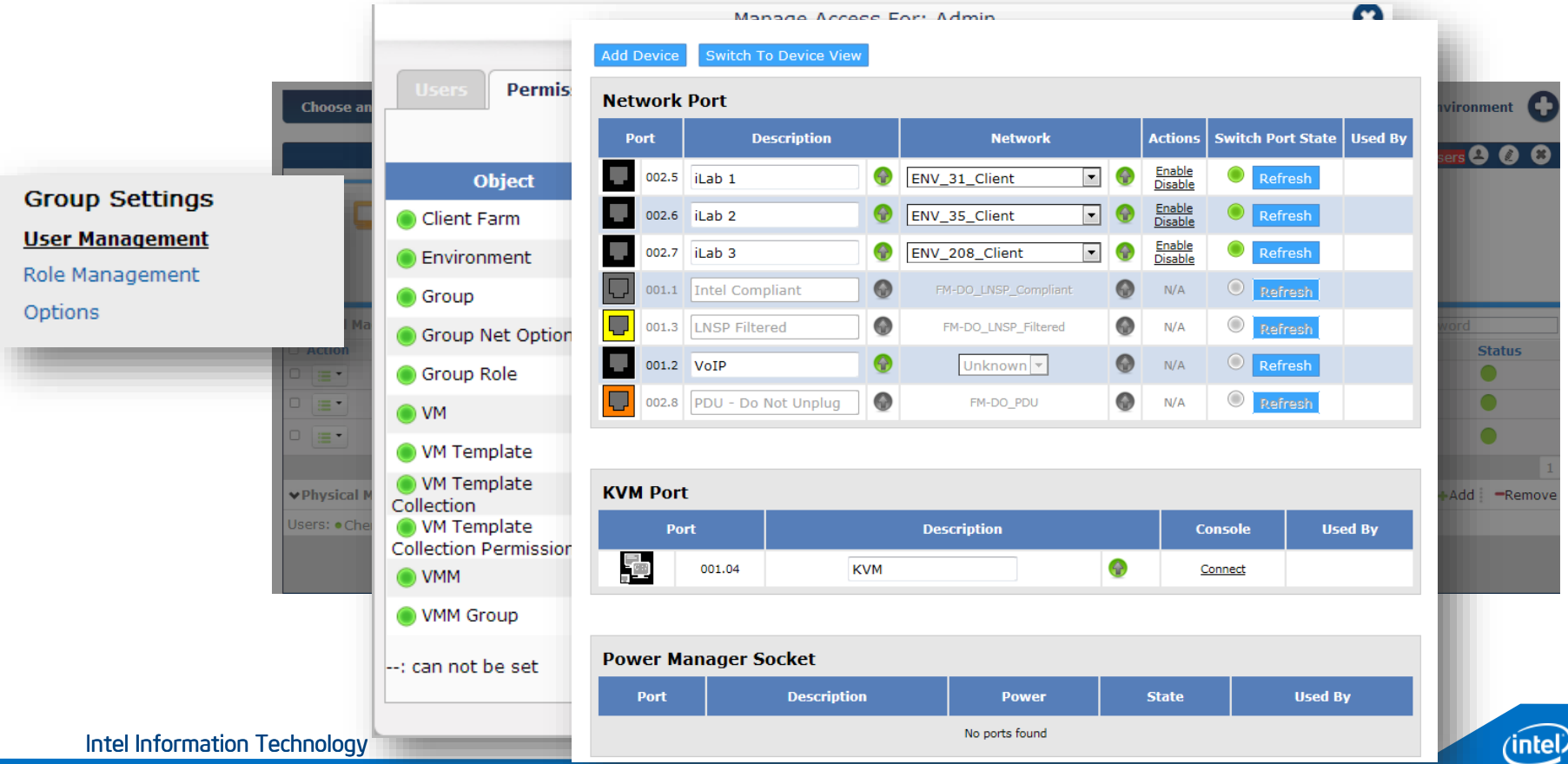
Leveraging existing Portal

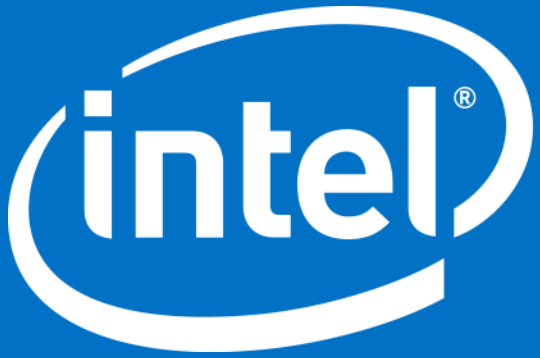
- Simple
- Smooth
- Customer-Designed
- Collaboration



Fit for Intel internal business needs

- Users doesn't need to care the physical locations of VM
- More powerful permission control
- Physical machine & device management





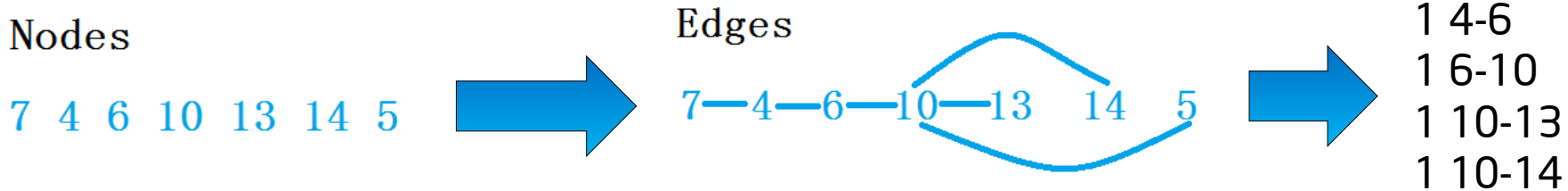
Backup

Configuration of Hadoop on Ceph

- Hadoop 1.1.x & Ceph 0.61.8
- Install Packages:libcephfs1 libcephfs-java
- Add Hadoop-cephfs plugin to $HADOOP_HOME/lib
- Create pool on Ceph MON Node for Hadoop use
- Edit core-site.xml
  - fs.ceph.impl = org.apache.hadoop.fs.ceph.CephFileSystem
  - fs.default.name = Ceph MON Server(ceph://10.239.149.9:6789)
  - ceph.data.pools = name of the pool you create

Construct the Vector Space

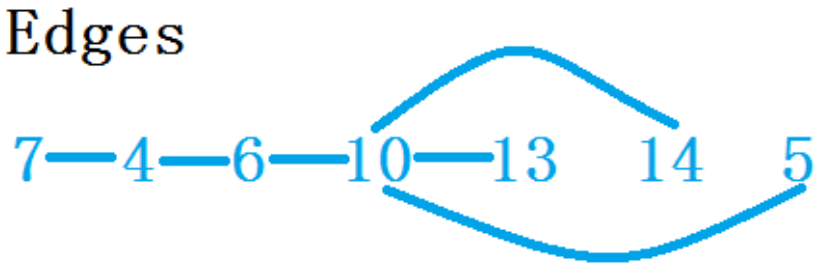
name_seq	relatedFieldNa	relatedValue	
7-4-4-4-4-4-4-4-4-4-4-6-4-4-4-4-4-4-4-10-...	port	d94829b6-30aa-4365-ad4b-b3c68a6155c8	0
7-4-4-4-4-4-4-4-6-10-13-10-14-10-5-10-...	port	0326097b-dc64-41b0-ac92-1eaac0495de6	0
[illegible]	port	7171bf08-a024-4255-8b2e-24ba32e823b3	0
7-4-4-4-4-6-10-13-10-14-10-5-10-5-10-...	port	43f4a8ff-26cc-4c5c-a96f-2997283dc788	0
[illegible]	port	ce8f6186-b1c3-483c-86d9-aa56f8099945	1
7-4-4-6-10-13-10-14-10-5-10-5-10-5-10...	port	62f20761-b965-4dc5-8a7e-e11aae56af36	1
7-4-4-6-10-13-10-14-10-5-10-5-10-5-10...	port	54daffe8-e4cc-4aaa-a004-303518ae17bb	1
7-4-4-6-10-13-10-14-10-5-10-5-10-5-10...	port	1f79332f-29c2-4704-8072-2172d17e27e1	1
7-4-4-6-10-13-10-14-10-5-10-5-10-5-10...	port	d099f48b-b697-4b9e-876a-2e1a966d19ce	2
7-4-4-6-10-13-10-14-10-5-10-5-10-5-10...	port	8ab84e91-e4e9-4dc1-956e-0136c494d27e	3



Matrix

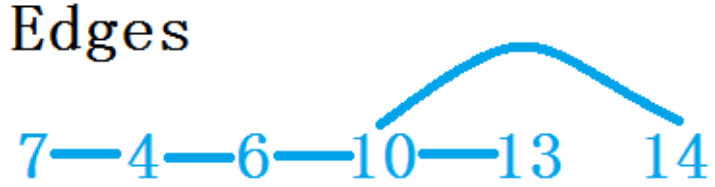
1 7-4
1 4-6
1 6-10
1 10-13
1 10-14
1 10-5

Construct the Vector Space



Matrix 1

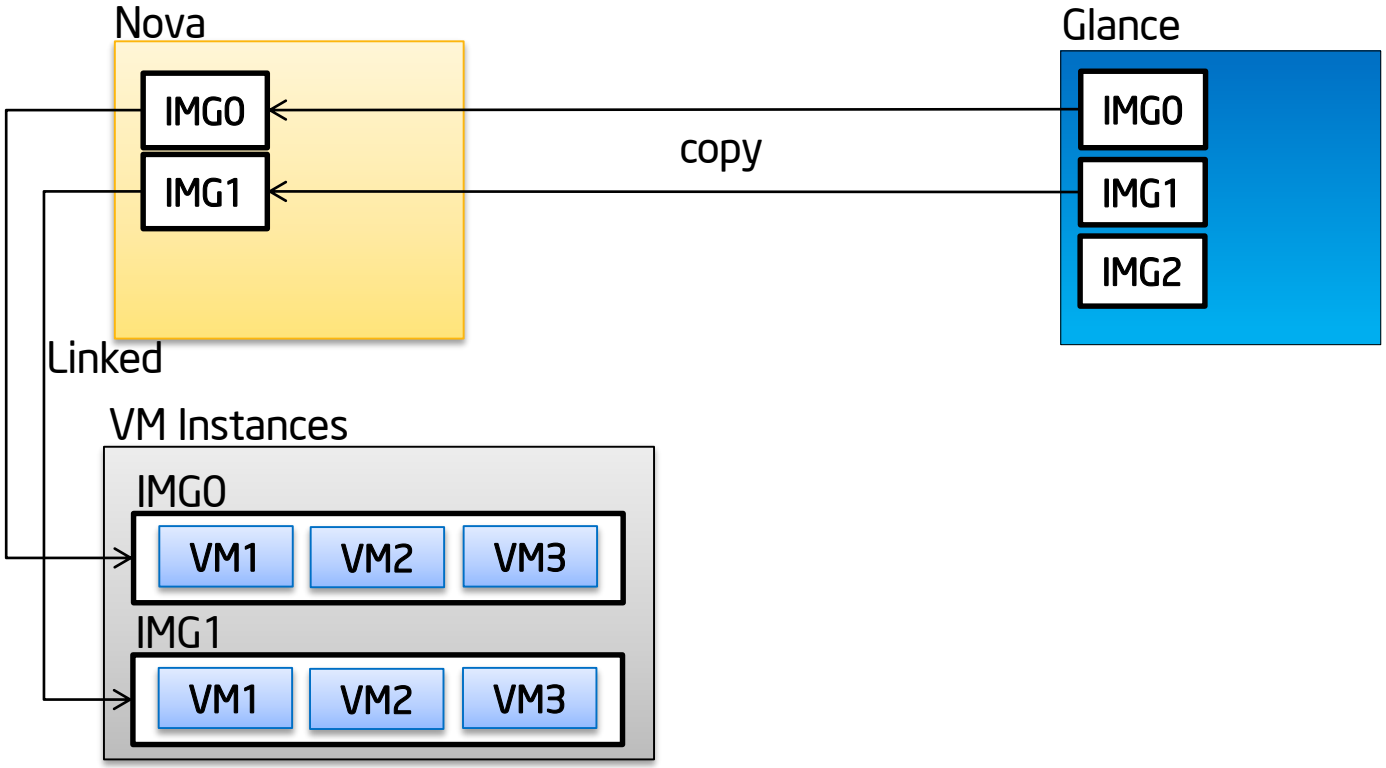
1 7-4
1 4-6
1 6-10
1 10-13
1 10-14
1 10-5



Matrix 2

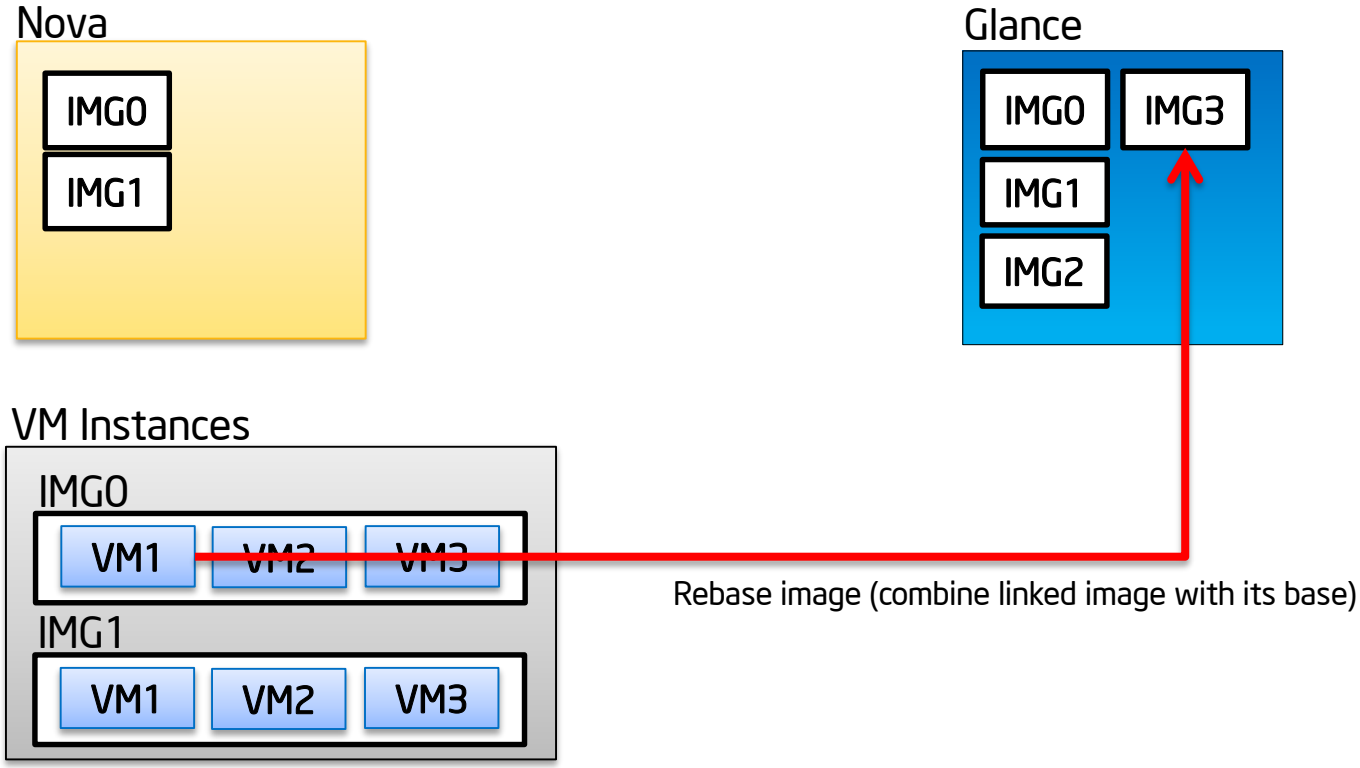
1 7-4
1 4-6
1 6-10
1 10-13
1 10-14
0 10-5

Snapshot Diagram



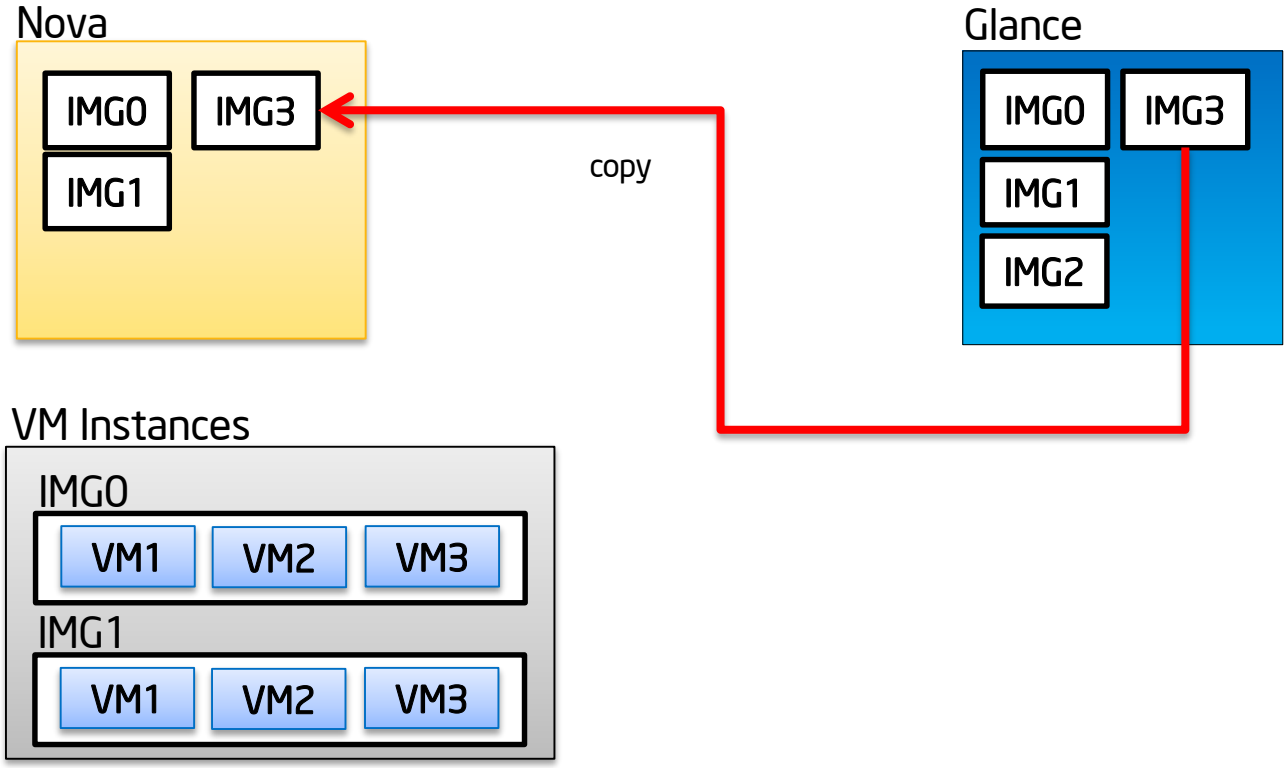
- This is the relation between VM Instances, Nova and Glance.
- When we launch an instance, the image from Glance, eg IMG0 will be copied into Nova.
- Then the nova will create a linked image as “disk” for VM Instances.

Snapshot Diagram



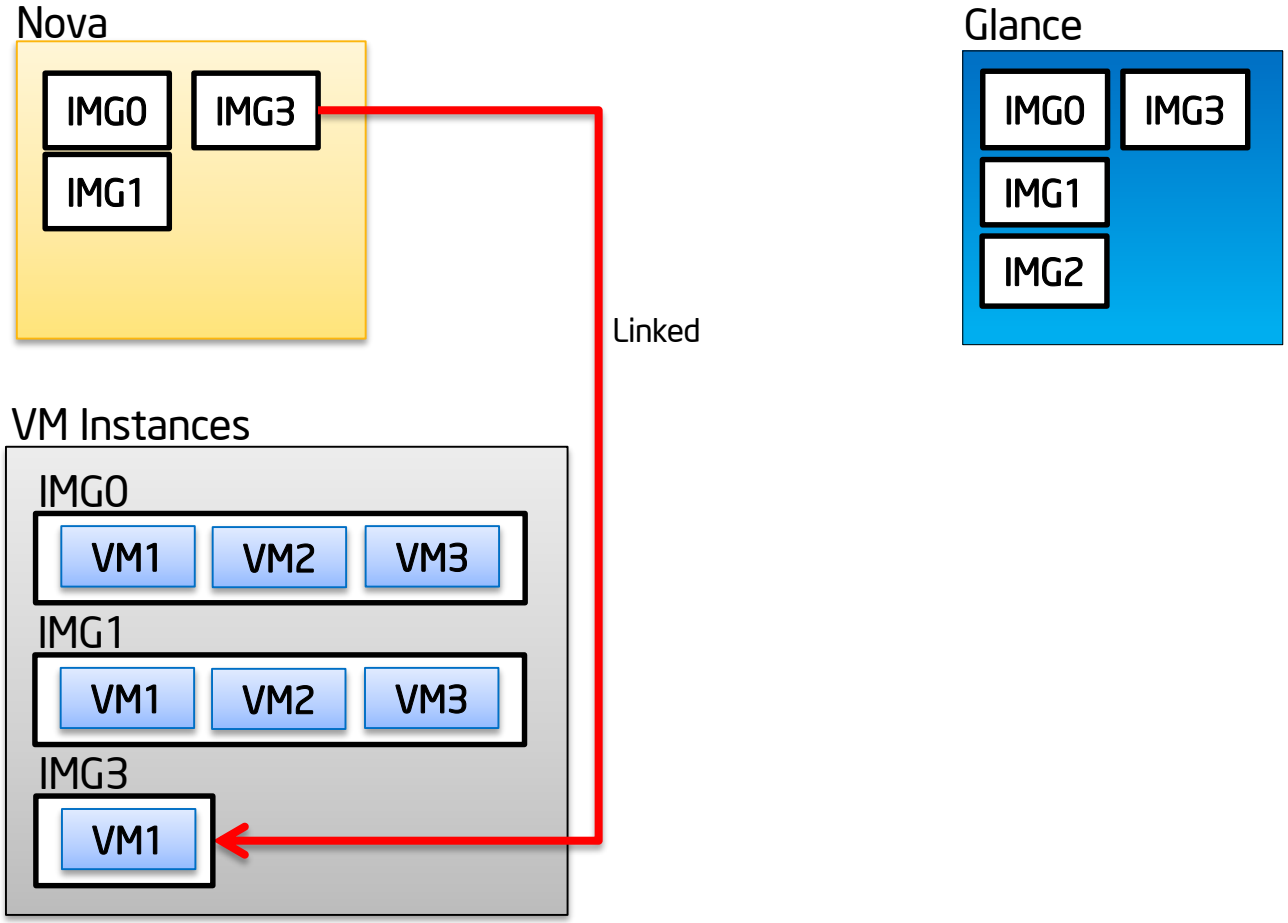
- When we run the snapshot, the Nova will rebase the instance into full image, and upload it to glance.
- It is really time consuming process if the size of the rebased image is huge (IMG0 + VM1)

Snapshot Diagram



- Now the new snapshot image is in database, when we run a VM from that image, it will copy the new image from Glance to Nova.
- The time taken to copy is depend on the size of the image.

Snapshot Diagram



- A new VM instance launched based on the snapshot that have been created
- It will create a linked image based on IMG3
- This is a really quick process since it is just create an empty linked clone image into VM Instances area.