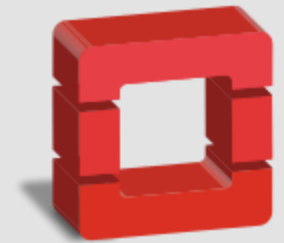




Openstack Compute Service Nova

«OpenStack Administration 101» , 30 Nov. – 3 Dec. 2021

Alessandro Costantini, Diego Michelotto – INFN CNAF



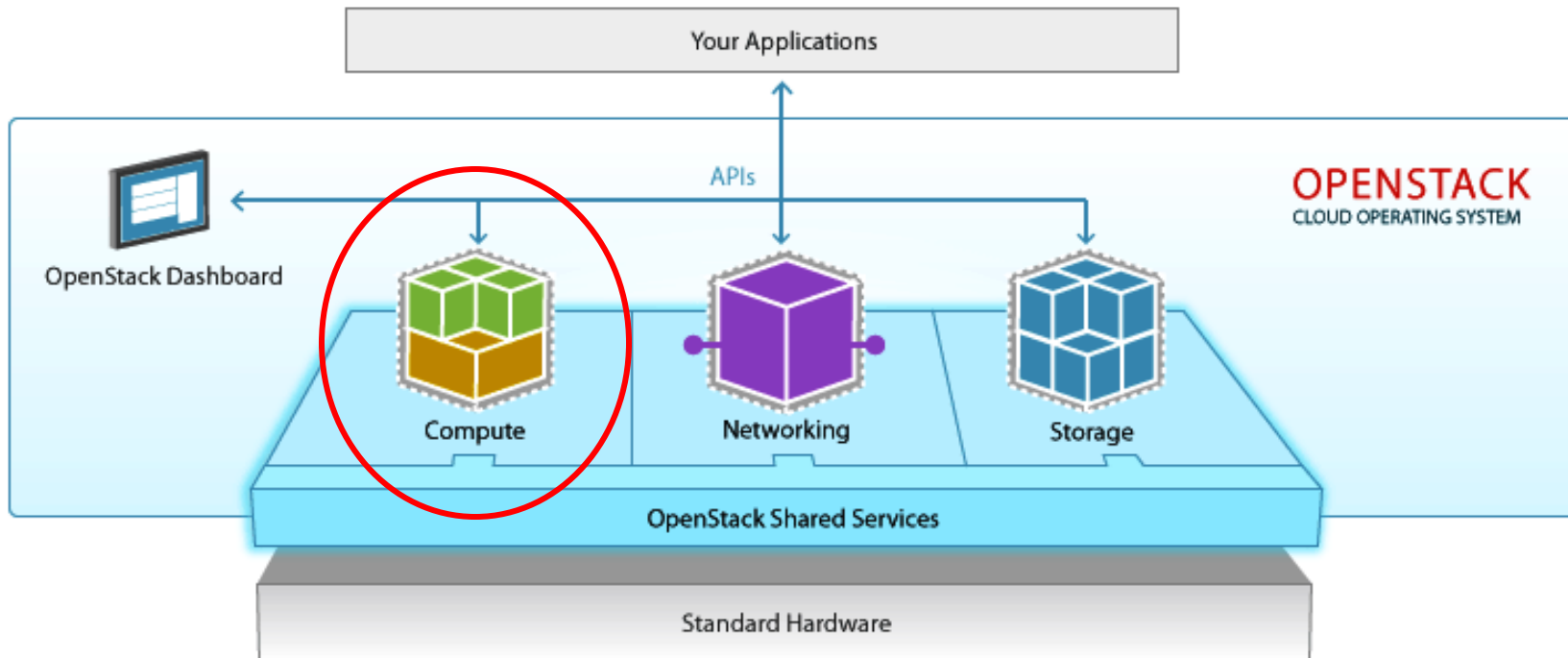
openstack
CLOUD SOFTWARE



Overview

- Openstack components
 - Openstack Compute Service (Nova)
 - Interaction with other services
- Filters and Weight
- Virtualization support
- Resource segregation
- VM Migration
- Nova Install&configure
 - Services, files, verification
- Ironic

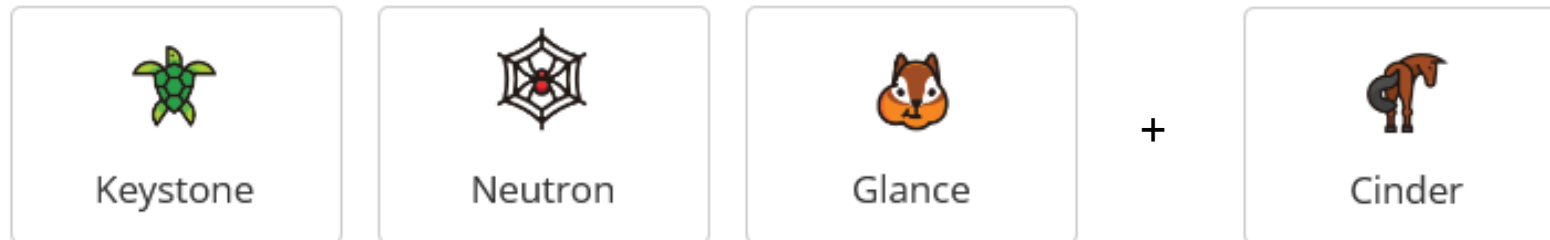
Openstack components



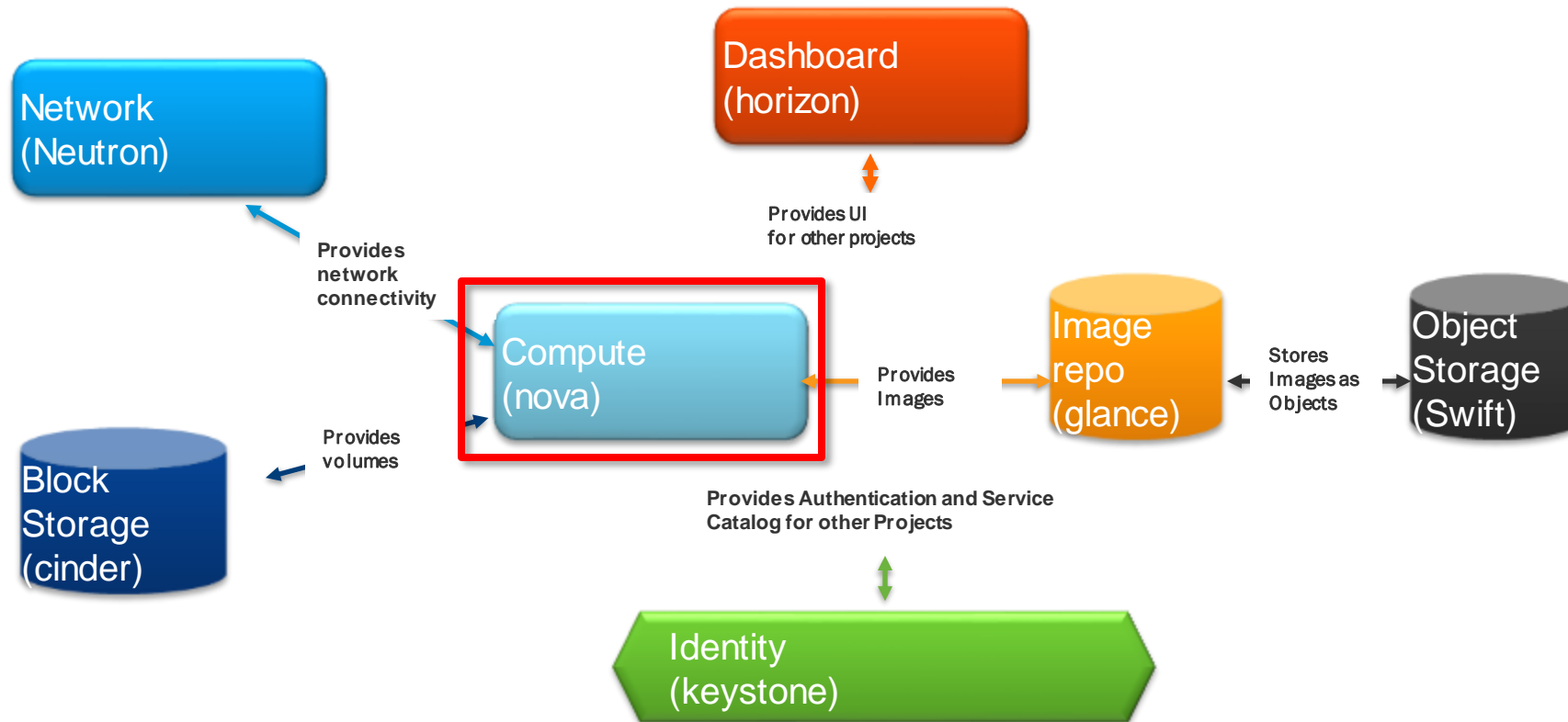
Openstack Compute service (Nova)

Nova is the OpenStack project that provides a way to provision compute instances (aka virtual servers). Nova supports creating virtual machines, baremetal servers (through the use of ironic)

Depends on



Openstack Compute service (Nova)



Openstack Compute services

Nova is managing basic virtual machine functions like creating, starting, stopping, etc.

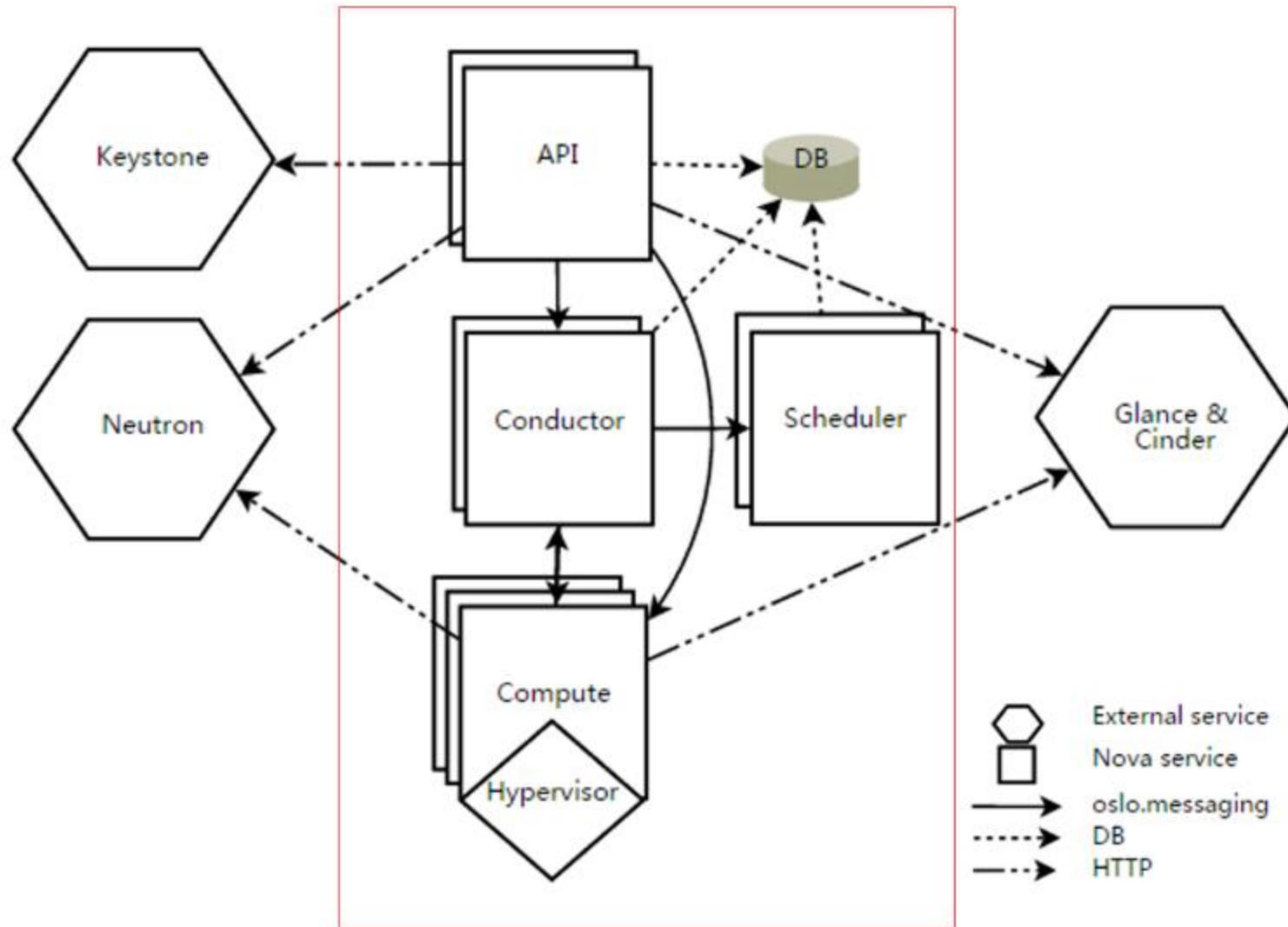
Nova uses a message broker (RabbitMQ) and a SQL database (MariaDB)

The main services that support Nova are:

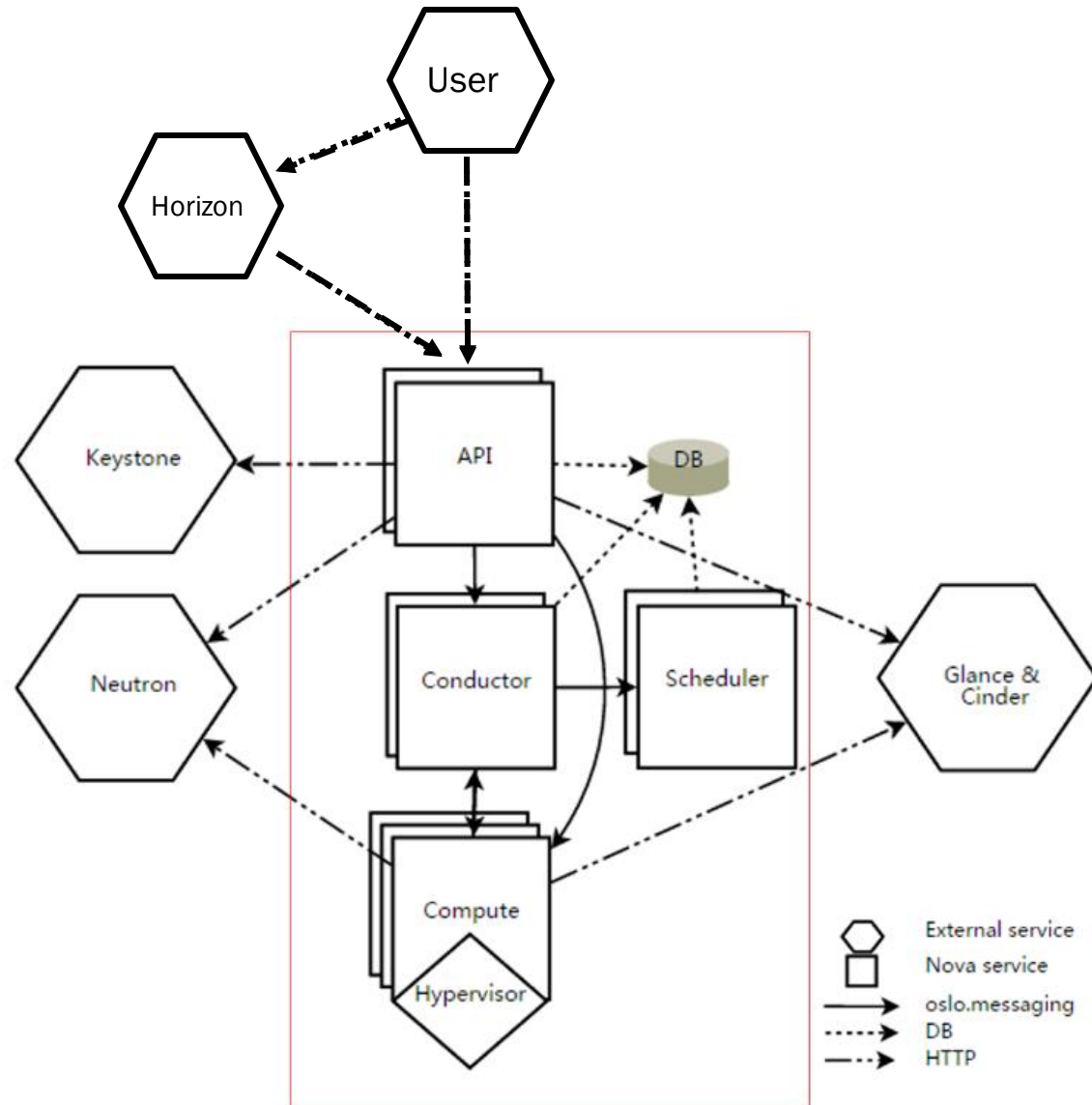
- **nova-api**: receives REST API calls from other services and clients and respond to them
- **nova-api-metadata**: accepts metadata requests from instances
- **nova-scheduler**: takes requests for starting instances from the queue and selects a compute node (hypervisor) for running a virtual machine on it
 - filters (CPU, RAM, etc)
 - weight
- **nova conductor**: proxy between the database and the other nova services
- **nova-compute**: daemon that runs on hypervisors, manages pool of computing resources and works with the available virtualization technologies (libvirt for KVM or QEMU, VMwareAPI for Vmware)
- **nova-novncproxy**: provides a proxy for accessing running instances through a VNC connection

Nova compute **does not include** the virtualization software, it uses related drivers to interact with the virtualization mechanisms available on the hypervisor

Nova services Architecture



Nova services Architecture



Filters and Weight

nova-scheduler: takes requests for starting instances from the queue and selects a compute node (hypervisor) for running a virtual machine on it

- Filters

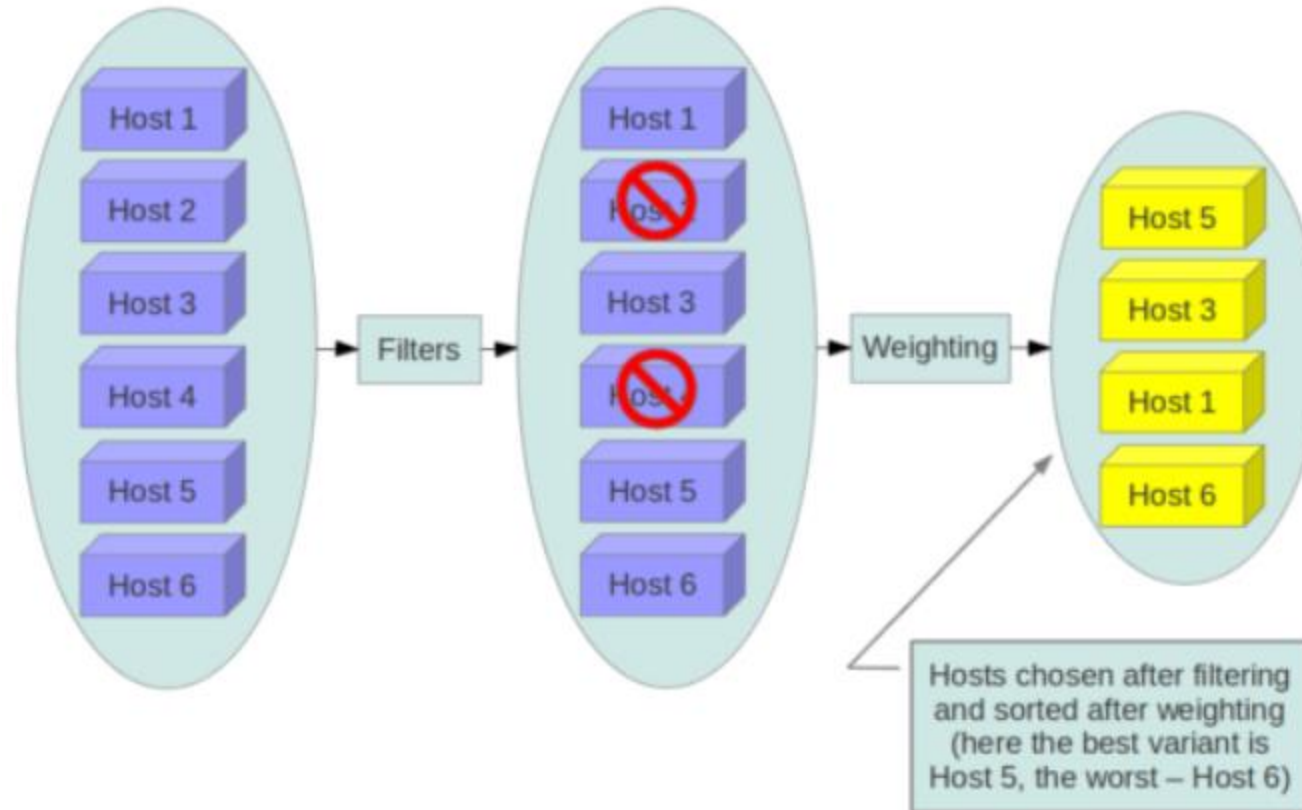
- The filter scheduler will filter in the following order:

- Availability zone
 - Ram
 - Disk
 - Core
 - Compute capabilities
 - Image properties

- Weight

- The default implementation is ramweight, which calculates the weight value according to the amount of free memory of the computing node. The more free, the greater the weight, and the instance will be deployed to the computing node with the most free memory

Filters and Weight



Filters: Flavors

In OpenStack, flavors define the compute, memory, and storage capacity of nova computing instances. To put it simply, a flavor is an available hardware configuration for a server. It defines the *size* of a virtual server that can be launched.

Flavor ID

- Unique ID (integer or UUID) for the new flavor.

Name

- Name for the new flavor.

Flavor	VCPUs	Disk (in GB)	RAM (in MB)
m1.tiny	1	1	512
m1.small	1	20	2048
m1.medium	2	40	4096
m1.large	4	80	8192
m1.xlarge	8	160	16384

VCPUs

- Number of virtual CPUs to use.

Memory MB

- Amount of RAM to use (in megabytes).

Root Disk GB

- Amount of disk space (in gigabytes) to use for the root (/) partition.
- The root disk is an ephemeral disk that the base image is copied into.
- When booting from a persistent volume it is not used.

Filters: Flavors

Ephemeral Disk GB

Amount of disk space (in gigabytes) to use for the ephemeral partition. This property is optional.

If unspecified, the value is 0 by default.

Ephemeral disks offer machine local disk storage linked to the lifecycle of a VM instance. When a VM is terminated, all data on the ephemeral disk is lost. Ephemeral disks are not included in any snapshots.

Swap

Amount of swap space (in megabytes) to use. This property is optional. If unspecified, the value is 0 by default.

Is Public

Boolean value that defines whether the flavor is available to all users or private to the project it was created in.

This property is optional. In unspecified, the value is True by default.

By default, a flavor is public and available to all projects. Private flavors are only accessible to those on the access list for a given project and are invisible to other projects.

Extra Specs

Key and value pairs that define on which compute nodes a flavor can run. These are optional.

Virtualization support

Nova supports different virtualization software (through related drivers) such as:

- **KVM** - Kernel-based Virtual Machine. The virtual disk formats that it supports is inherited from QEMU since it uses a modified QEMU program to launch the virtual machine. The supported formats include raw images, the qcow2, and VMware formats.
- **LXC** - Linux Containers (through libvirt), used to run Linux-based virtual machines.
- **QEMU** - Quick EMUlator, generally only used for development purposes.
- **VMware vSphere** 5.1.0 and newer - Runs VMware-based Linux and Windows images through a connection with a vCenter server.
- **Hyper-V** - Server virtualization with Microsoft Hyper-V, use to run Windows, Linux, and FreeBSD virtual machines. Runs nova-compute natively on the Windows virtualization platform.
- **Virtuozzo** 7.0.0 and newer - OS Containers and Kernel-based Virtual Machines supported. The supported formats include ploop and qcow2 images.
- **PowerVM** - Server virtualization with IBM PowerVM for AIX, IBM i, and Linux workloads on the Power Systems platform.
- **zVM** - Server virtualization on z Systems and IBM LinuxONE, it can run Linux, z/OS and more.
- **Ironic** - OpenStack project which provisions bare metal (as opposed to virtual) machines.

Why segregate resources?

Infrastructure

- Expose logical groupings of infrastructure based on physical characteristics
- Expose logical groupings of infrastructure based on some abstract functionality/capability

Workloads

- Ensure an even spread of a single workload
- Ensure close placement of related workloads

Resource segregation

How resource segregation usually works

- **Infrastructure segregation:**
 - Logical data center constructs
 - Contain some number of logical clusters
 - Clusters typically:
 - Are relatively small (0's to 00's of nodes per cluster)
 - Are tightly coupled to physical storage and network layout
- **Workload segregation:**
 - Host-level affinity/anti-affinity
 - CPU-level affinity/anti-affinity

Resource segregation

Segregation in OpenStack

Infrastructure segregation

- Cell
- Regions
- Host aggregates
- Availability zones

Resource segregation

Region Vs Cell

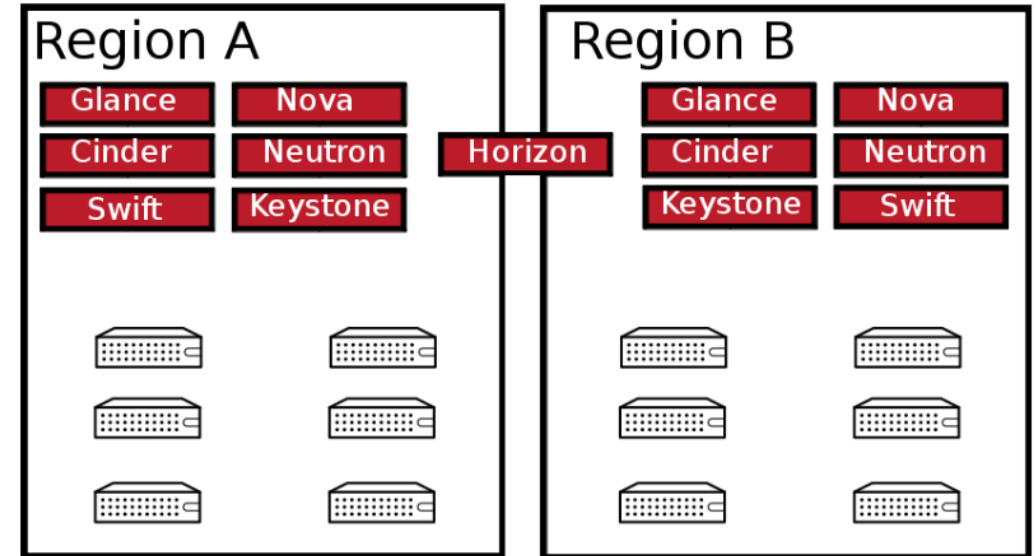
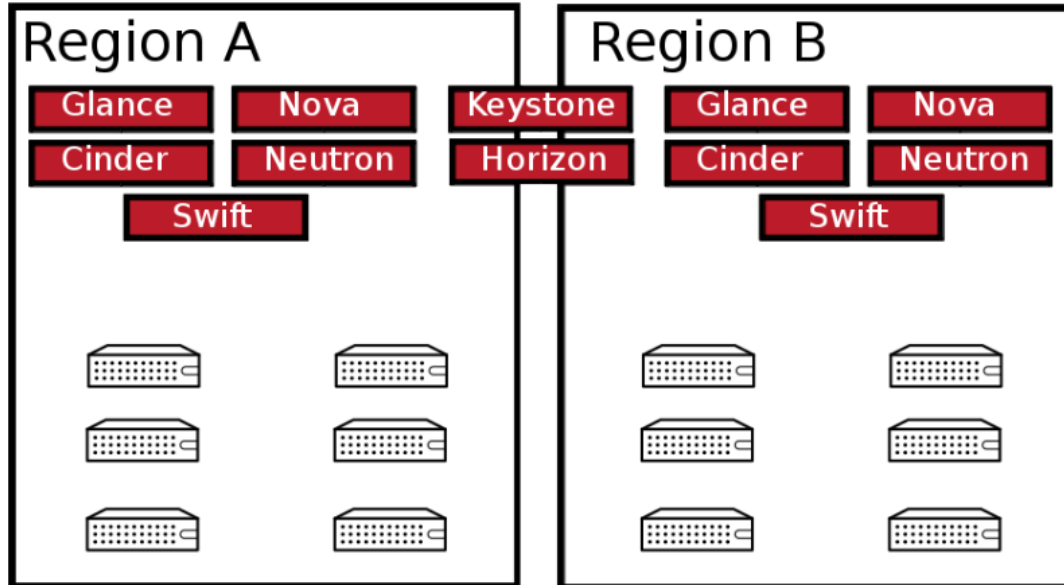
Region

- Present in every OpenStack deployments
- Groups services and resources
 - Share at least a Keystone and Horizon installation
- Implement their own targetable API endpoints

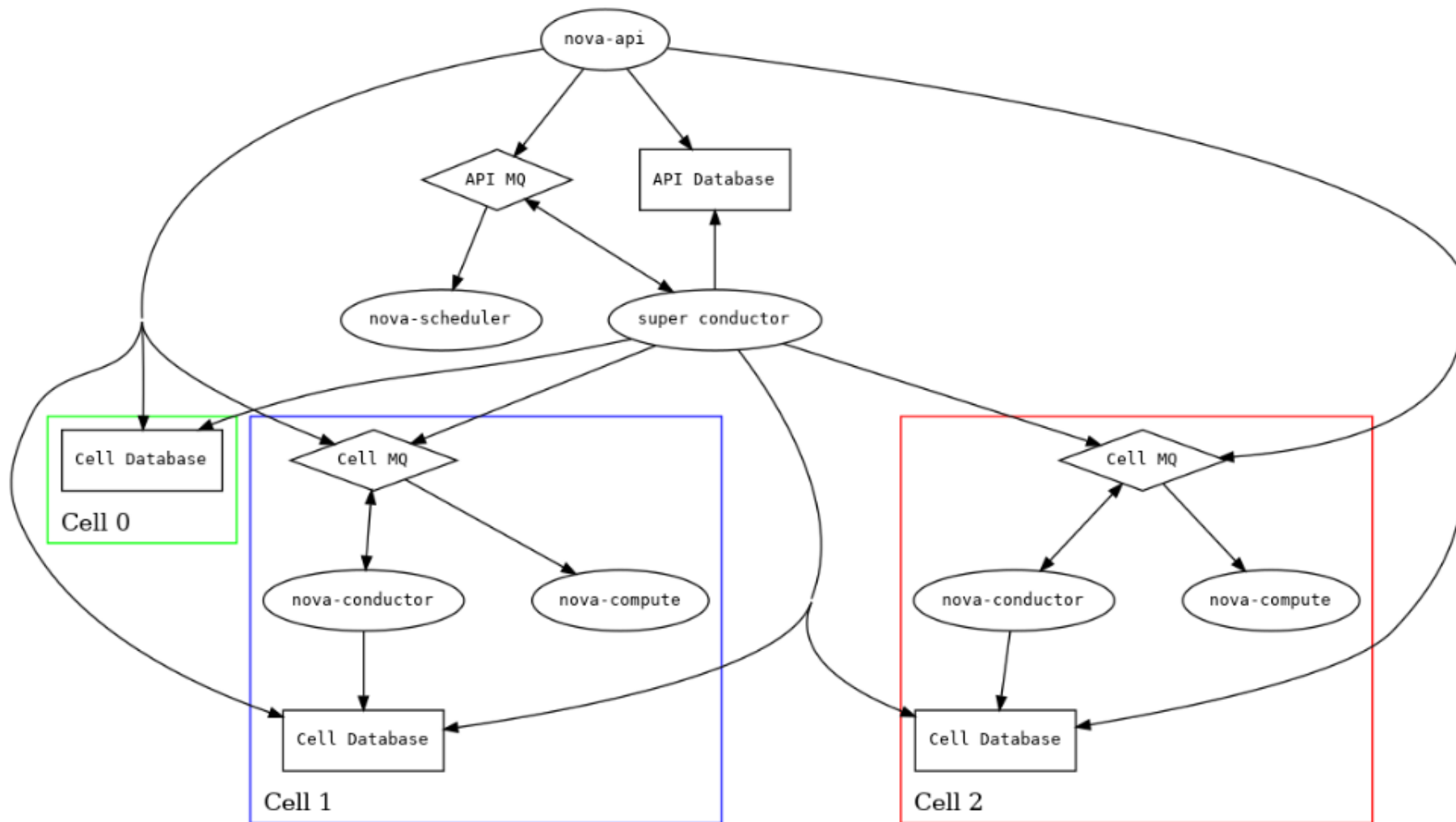
Cell

- Cells are a Nova feature that allow the partition of Nova deployments
- Has a single compute endpoint
- Each cell has their DB and AQMP

Resource segregation: Region



Resource segregation: Cell



Resource segregation

Availability zone Vs Host Aggregate

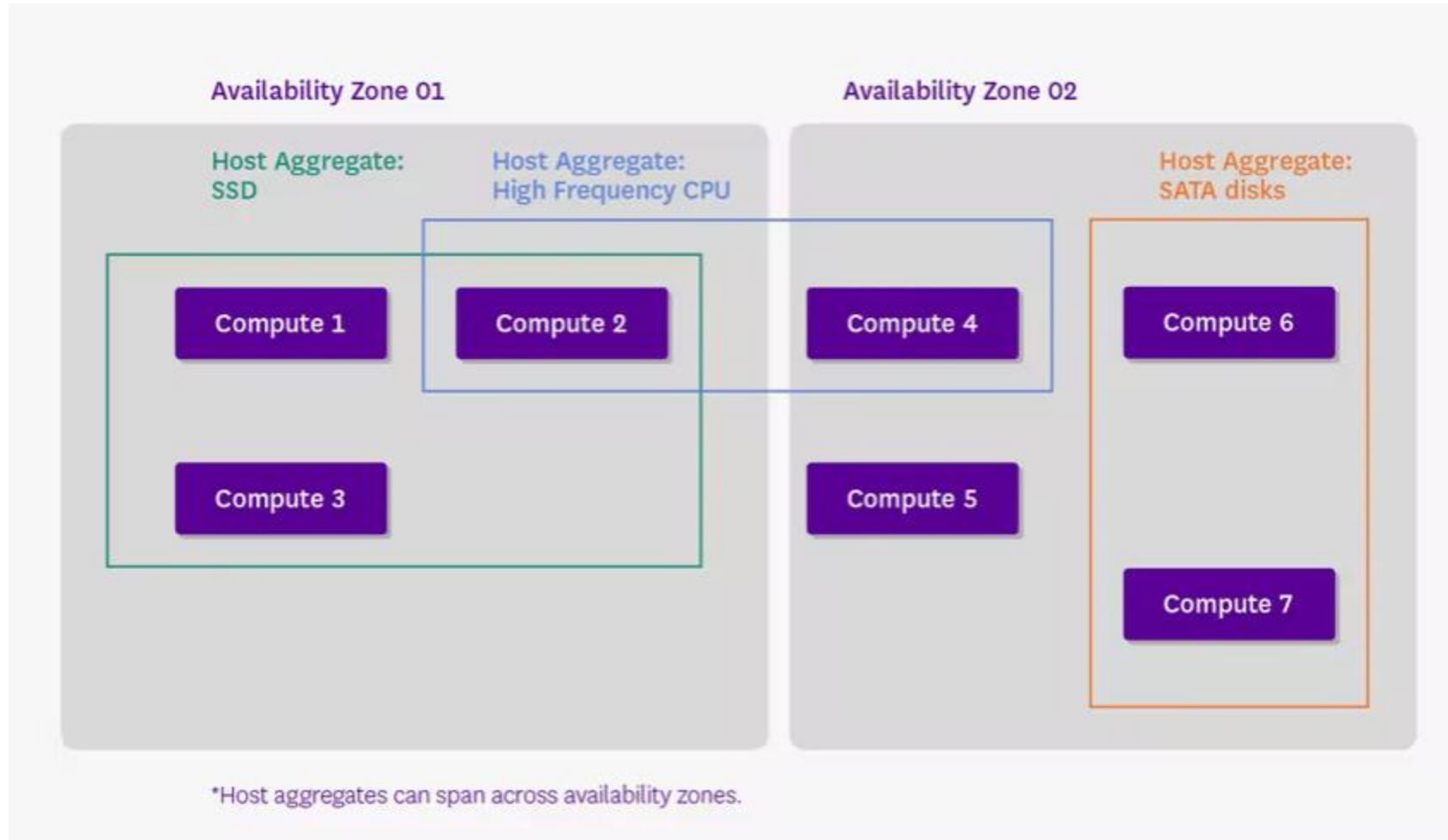
Availability zone

- Availability Zones are an end-user visible logical abstraction for partitioning a cloud
- The addition of this specific metadata to an aggregate makes the aggregate visible from an end-user perspective and consequently allows users to schedule instances to a specific set of hosts

Host aggregate

- Mechanism for partitioning hosts in an OpenStack cloud
- Allow administrators to assign key-value pairs to groups of machines.
- Each node can have multiple aggregates, each aggregate can have multiple key-value pairs, and the same key-value pair can be assigned to multiple aggregates.

Resource segregation: AZ vs HA

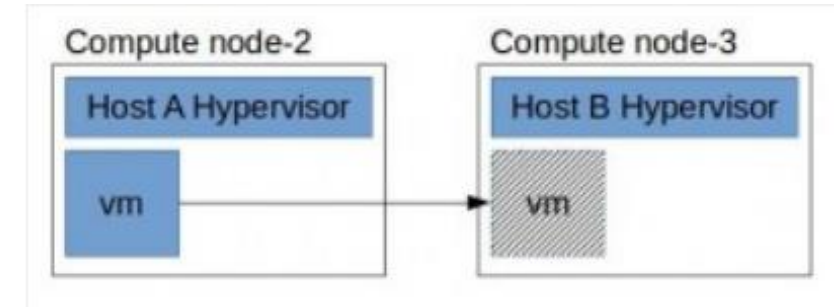


- A host can be part of multiple aggregates but it can only be in one availability zone.
- By default a host is part of a default availability zone even if it doesn't belong to an aggregate.

VM migration

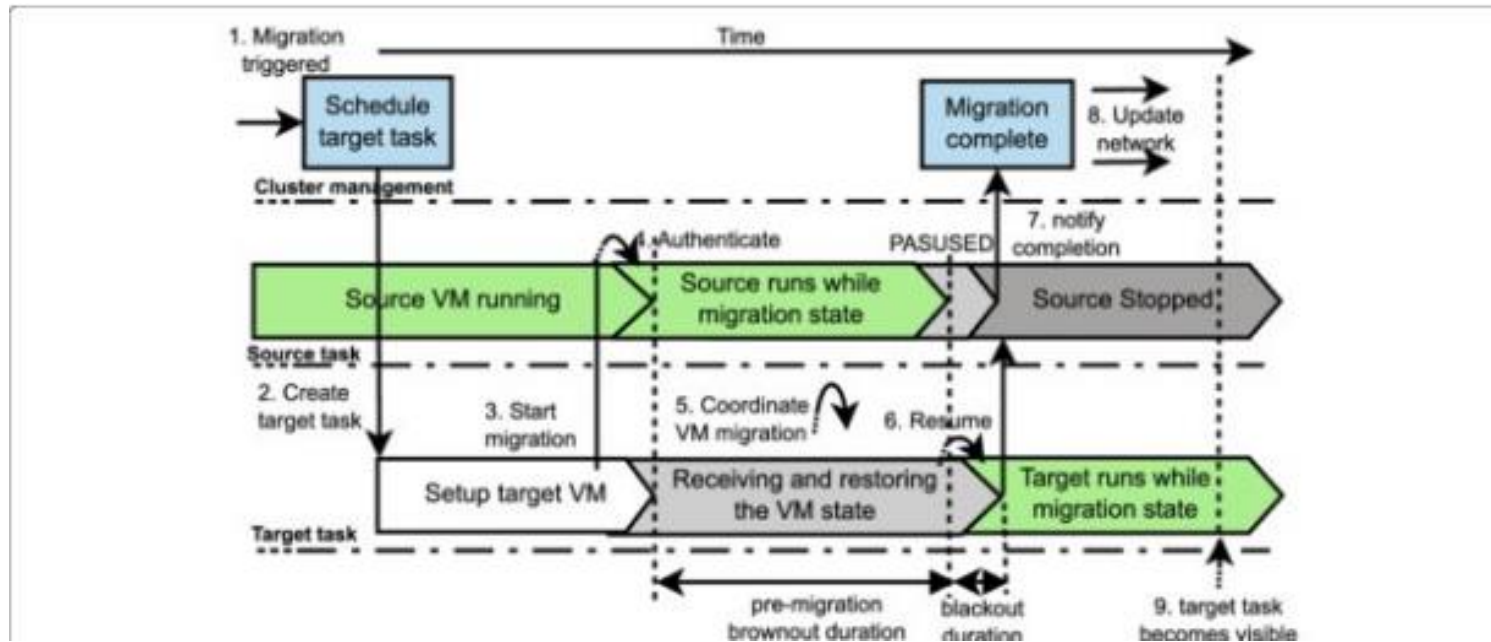
Migration: Move an instance from one compute host to another

Migration in Openstack



- **Migration**
 - The instance is shut down, then moved to another hypervisor and restarted. The instance recognizes that it was rebooted, and the application running on the instance is disrupted.
- **Live migration**
 - The instance comprising of its states, memory and emulated devices, is moved from one hypervisor to another with ideally no downtime
 - **Shared storage-based live migration.** The instance has ephemeral disks that are located on storage shared between the source and destination hosts.
 - **Block live migration,** or simply block migration. The instance has ephemeral disks that are not shared between the source and destination hosts.
 - **Volume-backed live migration.** Instances use volumes rather than ephemeral disks.
- **Evacuate**
 - The instance is restarted when it was running on a compute node that has gone down.

VM migration



Openstack compute (Controller): installation and configuration

All nova deployments must now have the following databases available and configured:

- The “nova” database
- The “API” database
- One special “cell” database called “cell0”
- One (or eventually more) “cell” databases

```
mysql -uroot -p

CREATE DATABASE nova_api;
GRANT ALL PRIVILEGES ON nova_api.* TO 'nova'@'localhost' IDENTIFIED BY 'NOVA_DBPASS';
GRANT ALL PRIVILEGES ON nova_api.* TO 'nova'@'%' IDENTIFIED BY 'NOVA_DBPASS';

CREATE DATABASE nova;
GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'localhost' IDENTIFIED BY 'NOVA_DBPASS';
GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'%' IDENTIFIED BY 'NOVA_DBPASS';

CREATE DATABASE nova_cell0;
GRANT ALL PRIVILEGES ON nova_cell0.* TO 'nova'@'localhost' IDENTIFIED BY 'NOVA_DBPASS';
GRANT ALL PRIVILEGES ON nova_cell0.* TO 'nova'@'%' IDENTIFIED BY 'NOVA_DBPASS';
flush privileges;
exit
```


Openstack compute: installation and configuration

Create nova user

```
$ openstack user create --domain default --password-prompt nova
```

```
User Password:
```

```
Repeat User Password:
```

```
+-----+-----+
| Field          | Value          |
+-----+-----+
| domain_id      | default        |
| enabled        | True           |
| id             | 8a7dbf5279404537b1c7b86c033620fe |
| name           | nova           |
| options        | {}             |
| password_expires_at | None          |
+-----+-----+
```

Openstack compute: installation and configuration

Add the admin role to the nova user

```
$ openstack role add --project service --user nova admin
```

Create nova service

```
$ openstack service create --name nova \  
  --description "OpenStack Compute" compute
```

Field	Value
description	OpenStack Compute
enabled	True
id	060d59eac51b4594815603d75a00aba2
name	nova
type	compute

Openstack compute: installation and configuration

Create nova endpoints

- public
- internal
- admin

```
$ openstack endpoint create --region RegionOne \  
compute public http://controller:8774/v2.1
```

Field	Value
enabled	True
id	3c1caa473bfe4390a11e7177894bcc7b
interface	public
region	RegionOne
region_id	RegionOne
service_id	060d59eac51b4594815603d75a00aba2
service_name	nova
service_type	compute
url	http://controller:8774/v2.1

Openstack compute: installation and configuration

Install the packages:

```
# yum install openstack-nova-api openstack-nova-conductor \  
openstack-nova-novncproxy openstack-nova-scheduler
```

Edit the Configuration files:

Nova Config files

Located in `/etc/nova/` on nodes running Nova services

Controller Nodes:

Conf-file

nova.conf, api-paste.ini, policy.json

nova.conf, policy.json

nova.conf

nova.conf, policy.json

--

nova.conf, policy.json

Service/Description

- nova-api service config file

- nova-cert service config file

- nova-conductor service config file

- nova-consoleauth service config file

- nova-manage utility config file

- nova-scheduler service config file

Compute Nodes:

Conf-file

nova.conf, policy.json

Service/Description

- nova-compute config file

nova-manage controls cloud computing instances by managing various admin-only aspects of Nova.

Nova Config files

```
# nova.conf

[DEFAULT]

#
# From nova.conf
#

#
# Availability zone for internal services. For more information, refer to the
# documentation. (string value)
internal_service_availability_zone=internal
...
[compute]

#
# From nova.conf
#

#
# Enables reporting of build failures to the scheduler. For more
information,
# refer to the documentation. (integer value)
#consecutive_build_service_disable_threshold=10
```

```
# api-paste.ini

#####
# OpenStack #
#####

[composite:osapi_compute]
use = call:nova.api.openstack.urlmap:urlmap_factory
/: oscomputeversions
/v2: oscomputeversion_legacy_v2
/v2.1: oscomputeversion_v2
```

```
# policy.json

{
    "alias 1" : "definition 1",
    "target 1" : "rule 1",
}

{
"admin_required": "role:admin or is_admin:1",
"owner": "user_id:%(user_id)s",
"identity:change_password": "rule:admin_or_owner"
}
```

Openstack compute: installation and configuration

Populate databases

```
# su -s /bin/sh -c "nova-manage db sync" nova
```

Start the services

```
# systemctl enable \  
  openstack-nova-api.service \  
  openstack-nova-scheduler.service \  
  openstack-nova-conductor.service \  
  openstack-nova-novncproxy.service  
# systemctl start \  
  openstack-nova-api.service \  
  openstack-nova-scheduler.service \  
  openstack-nova-conductor.service \  
  openstack-nova-novncproxy.service
```

Openstack compute (node): installation and configuration

Install the packages:

```
# yum install openstack-nova-compute
```

Edit the Configuration files:

```
/etc/nova/nova.conf
```

Start the services

```
# systemctl enable libvirtd.service openstack-nova-compute.service  
# systemctl start libvirtd.service openstack-nova-compute.service
```

On Controller: Discover host(s)

```
# su -s /bin/sh -c "nova-manage cell_v2 discover_hosts --verbose" nova  
  
Found 2 cell mappings.  
Skipping cell0 since it does not contain hosts.  
Getting compute nodes from cell 'cell1': ad5a5985-a719-4567-98d8-8d148aaae4bc  
Found 1 computes in cell: ad5a5985-a719-4567-98d8-8d148aaae4bc  
Checking host mapping for compute host 'compute': fe58ddc1-1d65-4f87-9456-bc040dc106b3  
Creating host mapping for compute host 'compute': fe58ddc1-1d65-4f87-9456-bc040dc106b3
```


Openstack compute: installation and configuration

Verify the installation

```
[root@oa101-ac-ctrl ~(keystone_admin)]# . keystone_admin
[root@oa101-ac-ctrl ~(keystone_admin)]#
[root@oa101-ac-ctrl ~(keystone_admin)]#
[root@oa101-ac-ctrl ~(keystone_admin)]#
[root@oa101-ac-ctrl ~(keystone_admin)]# nova-manage cell_v2 list_cells
```

Name	UUID	Transport URL	Database Connection	Disabled
cell0	00000000-0000-0000-0000-000000000000	none:/	mysql+pymysql://nova:****@10.10.0.41/nova_cell0	False
default	c01ef183-c93a-41d4-b655-3ba72399477a	rabbit://guest:****@10.10.0.41:5672/	mysql+pymysql://nova:****@10.10.0.41/nova	False

```
[root@oa101-ac-ctrl ~(keystone_admin)]#
[root@oa101-ac-ctrl ~(keystone_admin)]#
[root@oa101-ac-ctrl ~(keystone_admin)]# openstack compute service list
```

ID	Binary	Host	Zone	Status	State	Updated At
1	nova-conductor	oa101-ac-ctrl.novalocal	internal	enabled	up	2021-11-26T17:09:46.000000
2	nova-scheduler	oa101-ac-ctrl.novalocal	internal	enabled	up	2021-11-26T17:09:43.000000
6	nova-compute	oa101-ac-ctrl.novalocal	nova	enabled	up	2021-11-26T17:09:40.000000
7	nova-compute	oa101-ac-hv.novalocal	nova	enabled	up	2021-11-26T17:09:42.000000

```
[root@oa101-ac-ctrl ~(keystone_admin)]#
[root@oa101-ac-ctrl ~(keystone_admin)]#
[root@oa101-ac-ctrl ~(keystone_admin)]# openstack hypervisor list
```

ID	Hypervisor Hostname	Hypervisor Type	Host IP	State
1	oa101-ac-ctrl.novalocal	QEMU	10.10.0.41	up
2	oa101-ac-hv.novalocal	QEMU	10.10.0.29	up

```
[root@oa101-ac-ctrl ~(keystone_admin)]#
```

Nova troubleshooting: Logs files

Located in `/var/log/nova/` on nodes running Nova services

Controller Nodes:

Logfile

nova-api.log

nova-cert.log

nova-conductor.log

nova-consoleauth.log

nova-manage.log

nova-scheduler.log

Service/Description

- nova-api service log file

- nova-cert service log file

- nova-conductor service log file

- nova-consoleauth service log file

- nova-manage utility log file

- nova-scheduler service log file

Compute Nodes:

Logfile

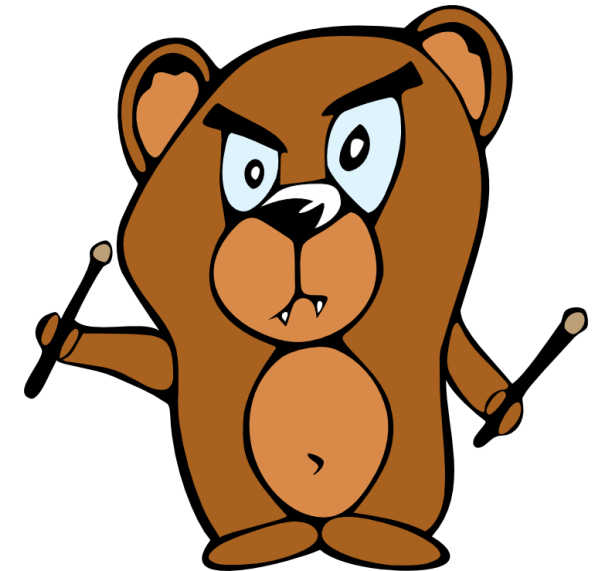
nova-compute.log

Service/Description

- nova-compute log file

Provisioning Bare Metal: Ironic

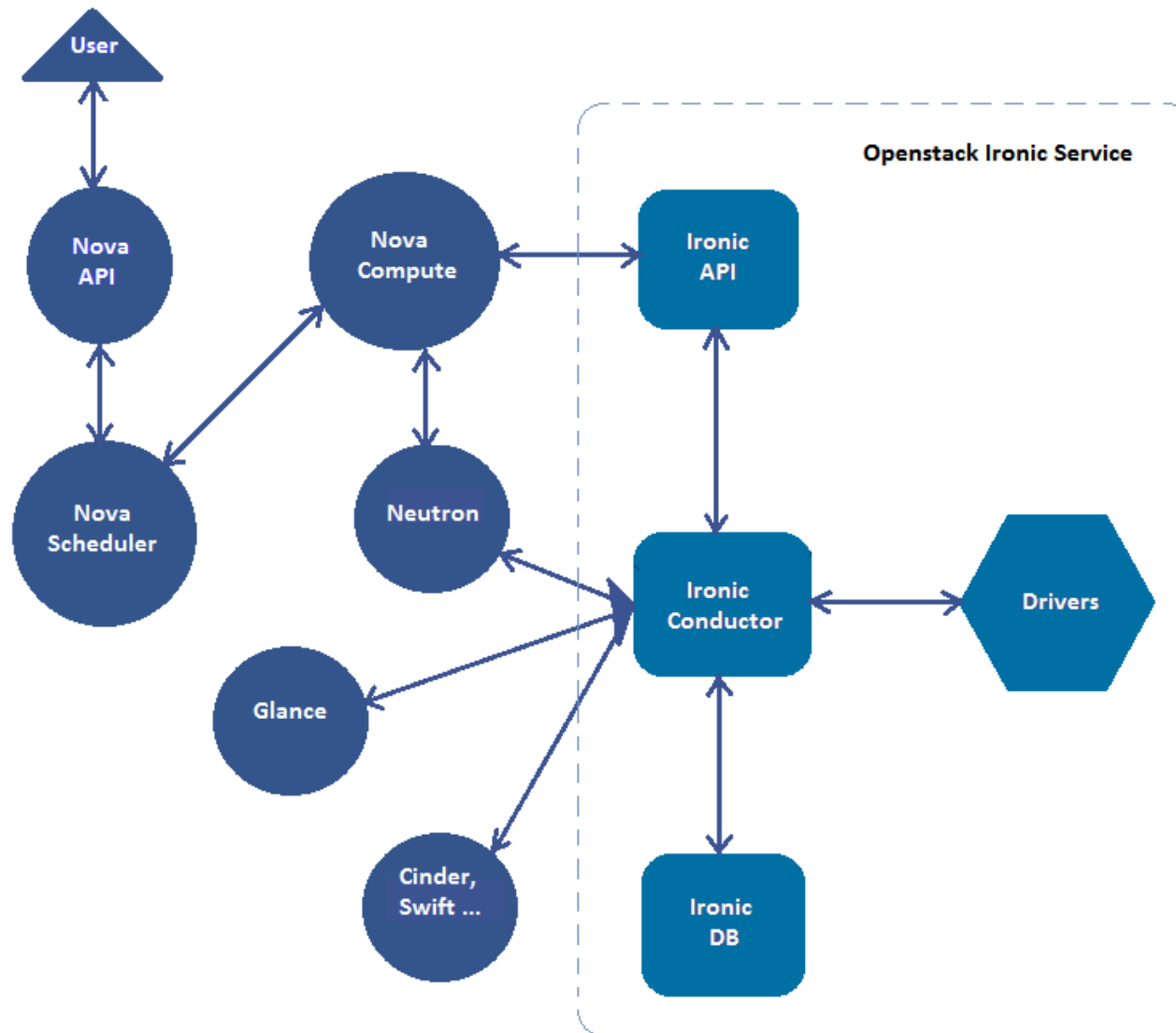
- Ironic: OpenStack Bare Metal Provisioning Program
- Initially developed to provision bare metal servers as part of OpenStack deployment
- Provision servers similarly to Virtual Machine
 - API driven
 - All HW exposed to user including GPUs, FPGAs etc.
- Support multi-tenancy
 - Provides isolation between bare metal on different tenant networks



Ironic Services

- **Ironic API**- An admin-only RESTful for interacting with the managed bare metal servers
- **Ironic Conductor**- Interact with the bare metal node and IPA
- **Ironic Python Agent (IPA)** – provides control over the hardware which is not available remotely to the Ironic Conductor
- **Ironic Inspector** – is used to discover hardware properties for a node managed by Ironic

Ironic Logical Architecture



References

- <https://www.openstack.org/software/releases/wallaby/components/nova>
- <https://object-storage-ca-ymq-1.vexxhost.net/swift/v1/6e4619c416ff4bd19e1c087f27a43eea/www-assets-prod/presentation-media/divideandconquer-2.pdf>
- <https://docs.openstack.org/nova/wallaby/install/controller-install.html>
- <https://docs.openstack.org/nova/wallaby/install/compute-install.html>
- <https://wiki.openstack.org/wiki/Ironic>