

# MODULE 3

## Working with OpenStack

# Module 4b Objectives

By the end of this module, you should be able to explain:

- What you can use the TWC Jira Help Desk for
- How to deploy a VM using the UI (Horizon)
- How to deploy a VM using the OpenStack CLI tools
- Basic IaaS VM deployment operations
  - Adding a cloud-init initialization file for cloud images
  - Attaching to Network
  - Booting from a Volume
- Using the CLI to manage an application across multiple regions

# TWC OpenStack Portal Onboarding

OpenStack Cloud Service Desk provides a single point of entry for potential and existing customers to get access and help to the TWC OpenStack Cloud.

Getting access to the TWC OpenStack Cloud:

URL: <https://cloud.twc.net/jira/servicedesk/customer/portal/25>


# Service Desk Capabilities and Services

Use the OpenStack Cloud Service Desk for:

- Access
  - New Account Request
  - Account Modification Request
- Network
  - DNS Requests
- Project
  - New project Request
  - Project Modification Request
- User-Help
  - OpenStack Help
  - Report a Bug
  - New Feature Request

# Access

[Help Center](#) > [OpenStack Cloud](#)My requests 1



Cloud  
Engineering  
&  
Operations

## OpenStack Cloud

[Return to: Cloud Operations Portal](#)

Welcome to OpenStack Cloud. Raise a request using one of the options below. Go here for more information on the TWC OpenStack Cloud or enter keyword below to search OpenStack Knowledge Base

**Note:**

Management of access to projects is handled by project owners. If you already have an OpenStack account and just need access to an existing project, please request that the project owner add you as a member to their project. If you do not have already have an OpenStack account, please first create a New Account Request before reaching out to the project owner to request access.

Find a solution

**Access**

[Network](#)

[Project](#)

[User-Help](#)



### New Account Request

Use this request type when you need to have a new user or system account created for you in the TWC OpenStack Cloud. All new account requests will be validated with the requestor's supervisor.



### Account Modification Request

Use this request type when you need to modify an existing user or system account in the TWC OpenStack Cloud. All account modification requests will be validated with the account owner and/or account owner's supervisor.



# New Account Request



## New Account Request

[Return to: Cloud Operations Portal](#)

### Note:

All project modification requests will be validated with the project owner and/or account owner's supervisor.

Raise this request on behalf of

Kirkpatrick, Kevin

### Summary

Brief Summary of this request

### Description

Detailed description of this request. Explain how this account will be used? What is the justification for this request? What type of workloads?

Estimated Completion Date (optional)



### Note:

Date to be entered and maintained Cloud Operation Teams

Account Access Type

System

User account is associated with the 'real' person and uses an EID for the OpenStack username. System accounts are desirable when you need to programmatically interact with OpenStack.

Corporate EID

Search for a user

Employee ID

System Account

Globally unique system username

Manager/Supervisor

Search for a user

Requestor's Manager/Supervisor

Customer Contact

Customer Contact Information (i.e. e-Mail, Phone)

Project

OpenStack project name associated with this request

Project Request Type

Existing

Project Request Type (New or Existing)

Project Owner

Search for a user

New projects please provide the Project Owner Username

Create Cancel

# Account Modification Request



## Account Modification Request

[Return to: Cloud Operations Portal](#)

### Note:

All project modification requests will be validated with the project owner and/or account owner's supervisor.

Raise this request on behalf of

Kirkpatrick, Kevin

### Summary

Brief Summary of Request

### Description

Detailed description of this request. Explain how it will be used? What is the justification for this request type of workloads?

Estimated Completion Date (optional)



### Note:

Date to be entered and maintained Cloud Operations Portal

Corporate EID

Employee ID

System Account

Globally unique system username

### Account Change Type

Remove Account

Remove Account

Enable

Disable

Username Change

Ownership Change

### Customer Contact

Create Cancel

User account is associated with the 'real' person and uses an EID for the OpenStack username. System accounts are desirable when you need to programmatically interact with OpenStack.

User's Supervisor/Manager

Customer Contact Information (Phone, e-Mail)


### Note:

Project Ownership changes e-Mail address of new owner must be provided.

JIRA Service Desk (2.1.1) · Atlassian

# Project

[Help Center](#) > [OpenStack Cloud](#)My requests 1



Cloud  
Engineering  
&  
Operations

## OpenStack Cloud


[Return to: Cloud Operations Portal](#)

Welcome to OpenStack Cloud. Raise a request using one of the options below. Go [here](#) for more information on the TWC OpenStack Cloud or enter keyword below to search OpenStack Knowledge Base

**Note:**


Management of access to projects is handled by project owners. If you already have an OpenStack account and just need access to an existing project, please request that the project owner add you as a member to their project. If you do not have already have an OpenStack account, please first create a New Account Request before reaching out to the project owner to request access.

[Access](#)  
[Network](#)  
**[Project](#)**  
[User-Help](#)



### New Project Request

Use this request type when you need to have a new TWC OpenStack Cloud project created. All new project requests will be validated with the requestor's owner's supervisor.



### Project Modification Request

Use this request type when you need to modify and existing TWC OpenStack Cloud project. All project modification requests will be validated with the project owner and/or requestor's supervisor.



# New Project Request




## New Project Request

[Return to: Cloud Operations Portal](#)

Note:

All project modification requests will be validated with the project owner and/or account owner's supervisor.

Raise this request on behalf of

 Kirkpatrick, Kevin

Summary

Brief Summary of Request

Description

Detailed description of this request. Explain how this account will be used? What is the justification for this request? What type of workloads?

Estimated Completion Date (optional)



Note:

Date to be entered and maintained Cloud Operation Teams

Project

OpenStack Project Name

Username

Existing system Username or EID

Customer Contact

Contact Information (i.e. Phone, e-Mail)

Requestor's Manager/Supervisor

User's Supervisor/Manager

Project Roles

OpenStack Project Role(s)

Create

Cancel



# Project Modification Request



## Project Modification Request

[Return to: Cloud Operations Portal](#)

### Note:

All project modification requests will be validated with the project owner and/or account owner's supervisor.

Raise this request on behalf of

Kirkpatrick, Kevin

### Summary

Brief Summary of Request

### Description

Detailed description of this request. Explain how this will be used? What is the justification for this request type of workloads?

Estimated Completion Date *(optional)*



### Note:

Date to be entered and maintained Cloud Operation

Project

OpenStack Project Name

Customer Contact

Contact Information (i.e. Phone, e-Mail)

Requestor's Manager/Supervisor

Search for a user

User's Supervisor/Manager

### Project Change Type

Remove Project

Remove Project

Enable Project

Disable Project

Project Name Change

Ownership Change

Quota Change

Instances *(optional)*

Injected Files *(optional)*

Injected Content Bytes *(optional)*

Volumes *(optional)*

Snapshots *(optional)*

Storage Space Requirements *(optional)*

Security Groups

Security Group Rules

Floating IP's

Networks

Ports

Routers

### Note:

When selecting a change type ensure that the proper corresponding fields have appropriate information (i.e. "Project Name Change" information populated in the "OpenStack Project Name" field.

Default Metadata Items value is 128

Default Instances value is 10

Default Volumes value is 10

Default Snapshots value is 10

Default Volumes value is 1000 GB

Default Security Group value is 10

Default Security Group Rules value is 100

Default Floating IP's value is 50


Default Networks value is 10

Default Ports value is 50

Default Routers value is 10

# Network

[Help Center](#) > [OpenStack Cloud](#)My requests 1



Cloud  
Engineering  
&  
Operations

## OpenStack Cloud


[Return to: Cloud Operations Portal](#)


Welcome to OpenStack Cloud. Raise a request using one of the options below. Go here for more information on the TWC OpenStack Cloud or enter keyword below to search OpenStack Knowledge Base

**Note:**

Management of access to projects is handled by project owners. If you already have an OpenStack account and just need access to an existing project, please request that the project owner add you as a member to their project. If you do not have already have an OpenStack account, please first create a New Account Request before reaching out to the project owner to request access.

[Access](#)  
**Network**  
[Project](#)  
[User-Help](#)

**DNS Request**  
OpenStack DNS Request



# DNS Request



## DNS Request

[Return to: Cloud Operations Portal](#)

### Note:

All Network requests will be validated with the project owner and/or account owner's supervisor.

Raise this request on behalf of

Kirkpatrick, Kevin

### Summary

### Description

### Project

### Project Owner

Search for a user

### Requestor's Manager/Supervisor

Search for a user

### Customer Contact

### Priority (optional)

Normal

### Request Date



### DNS Record Name

### DNS Record Type

### IP Address

### Note:

All requests will be part of os.cloud.twc.net, or a customer sub-domain of os.cloud.twc.net, and fully qualified forward entries for anything <name>.twcable.com etc. need to follow the typical / standard DNS process.

The floating IP allocated by OpenStack

### Additional Information (optional)

### Estimated Completion Date (optional)



### Note:

Date to be entered and maintained Cloud Operation Teams

### Attachment (optional)

Choose file(s)

### Note:

Attachment limit 10 MB


Create

Cancel

Will be replaced by a OpenStack DNSaaS function in 2015

# User-help

[Help Center](#) > [OpenStack Cloud](#)My requests 1



Cloud  
Engineering  
&  
Operations

## OpenStack Cloud

[Return to: Cloud Operations Portal](#)

Welcome to OpenStack Cloud. Raise a request using one of the options below. Go here for more information on the TWC OpenStack Cloud or enter keyword below to search OpenStack Knowledge Base

**Note:**

Management of access to projects is handled by project owners. If you already have an OpenStack account and just need access to an existing project, please request that the project owner add you as a member to their project. If you do not have already have an OpenStack account, please first create a New Account Request before reaching out to the project owner to request access.

Find a solution

[Access](#)

[Network](#)

[Project](#)

**[User-Help](#)**



**OpenStack Help**

OpenStack Q&A



**Report Bug**

Report potential bug or deficiency



**New Feature Request**

New Feature Request



# OpenStack Help



## OpenStack Help

[Return to: Cloud Operations Portal](#)

### Note:

Submit a questions, defect or new feature requests for OpenStack Environments. Existing [OpenStack Knowledge Database](#) location.


Raise this request on behalf of

 Kirkpatrick, Kevin

Summary

Description

Attachment (optional)

 Choose file(s)

Create

Cancel

# Report a Bug



## Report Bug

[Return to: Cloud Operations Portal](#)

### Note:

All project modification requests will be validated with the project owner and/or account owner's supervisor.

Raise this request on behalf of

 Kirkpatrick, Kevin

Bug Summary

Brief Summary of Bug/Issue

OpenStack Application

Cinder

Select the applications you reporting a bug

Bug Detailed Description

Detailed Description of Bug/Issue to include steps to replicate

Attachment (optional)

 Choose file(s)

Create

Cancel

# New Feature Request




## New Feature Request

[Return to: Cloud Operations Portal](#)

### Note:

All project modification requests will be validated with the project owner and/or account owner's supervisor.

Raise this request on behalf of

 Kirkpatrick, Kevin


### Feature Summary

Provide a brief description of the requested feature

### Feature Detailed Description

Provide a detailed description of the requesting feature

Attachment *(optional)*

 Choose file(s)

Create

Cancel



# MODULE 4

## Deploying Virtual Machines in OpenStack

# Setting up CLI Tools

- There are multiple ways of interacting with OpenStack, but one of the most straight forward is to use the CLI tools.
  - Supported on any platform that supports Python 2.7 (no 3.0 support yet):
    - Linux
    - Windows
    - OS-X
  - Installation is simplified with “pip” but easy\_install can also be used.
- Microsoft use case is slightly more difficult because the environment is set via system level resources, but otherwise works fine
- CLI tools can be used either directly or in a script, which can dramatically simplify re-use, especially for parametric scripts.
- [http://docs.openstack.org/user-guide/content/install\\_clients.html](http://docs.openstack.org/user-guide/content/install_clients.html)

# BASIC VM DEPLOYMENT

## Understanding Nova

# Nova - OpenStack Compute

- OpenStack Compute (Nova) is a cloud computing fabric controller (orchestrator)
- Provides a highly scalable management framework for virtual machines
- The original OpenStack orchestrator of orchestrators: Interacts with Neutron, Cinder, Glance to create and provision a VM
- Called by other services to implement compute aspects (Heat, Trove, Sahara)

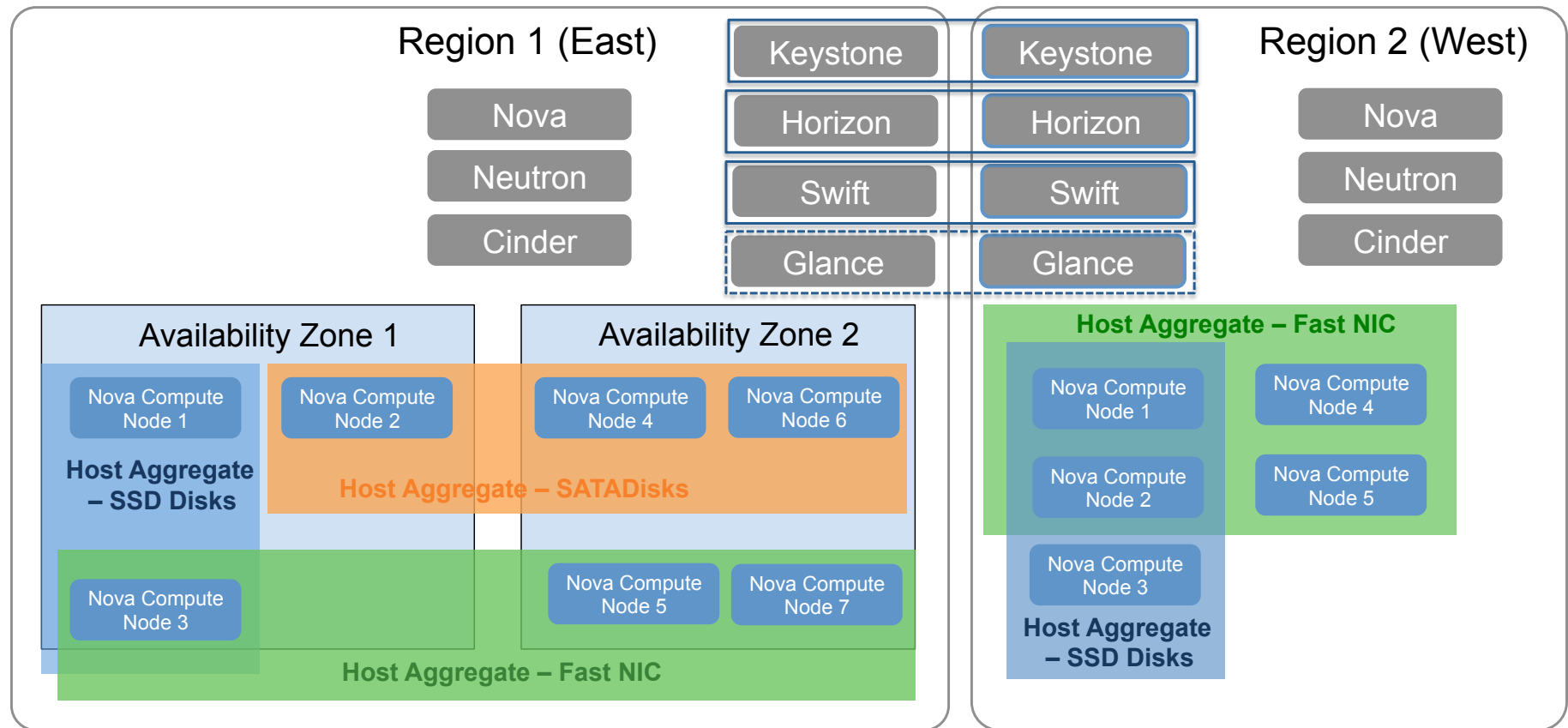
# Key Features

- Compute Entity management
  - Create/Delete operations
  - Power management (shutdown, power on, sleep)
  - Root Disk management (snapshots with Glance)
  - Migration (note: not available on all deployments)
  - Network attachment
  - Network edge security via ACL
  - Storage attachment (root or additional volume data)
- Scheduling Capabilities
  - Scheduling as in resource targeting
  - Flavor/"type" of system
  - Tag mechanism for affinity/anti-affinity
- VM access
  - Log access
  - "KVM" keyboard/video/mouse console access
  - Serial interface access (not available on all systems/target compute)

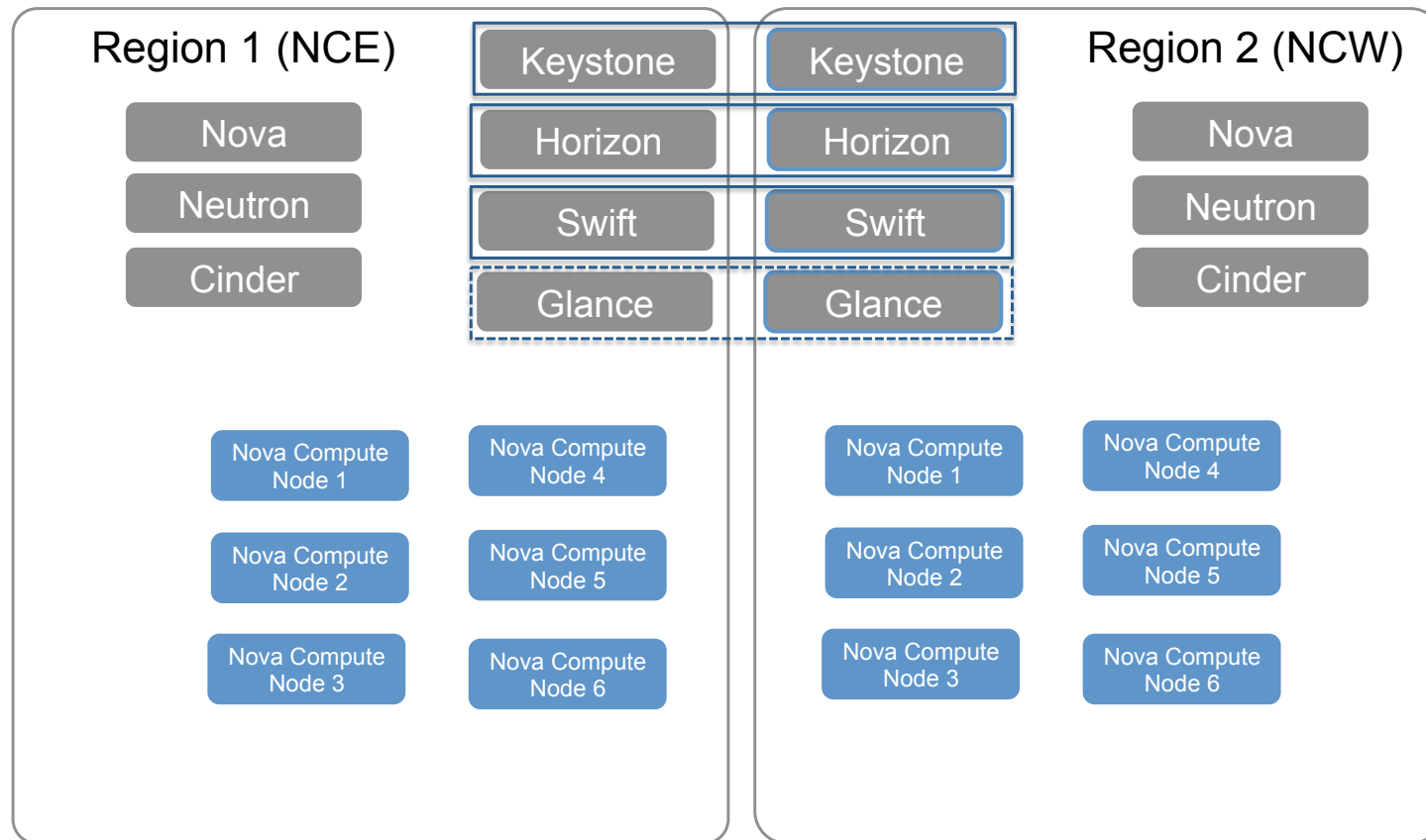
# Nova Concept - Flavors

- A flavor is an available hardware configuration for a server.
- Flavors define the configuration of the instance, including:
  - Number of Virtual CPUs
  - Amount of RAM
  - Size of ephemeral disks
- OpenStack provides number of predefined flavors which cloud administrators may edit or add to
- Users **must** pass a flavor from the set of available flavors defined
- Flavors are how a host aggregate are presented to a user

# OpenStack Segregation Methods (Generic)



# OpenStack Segregation Methods (TWC)





# Nova-Scheduler

- The Filter Scheduler supports filtering and weighting to make informed decisions on where a new instance should be created base on infrastructure status at time request
- All compute hosts periodically publish their status, available resources and hardware capabilities to nova-scheduler through the queue
- Physical placement of VM in the environment influenced or dictated based on required flavor, segregation methods, host capabilities and capacity, and other environmental filters.
- E.g. affinity or anti-affinity can assure close physical placement of VMs (reduce communication's latency) in a datacenter, or anti-affinity filters can disperse VMs (e.g. provide backup functionality)

# Nova Affinity/Anti-affinity

- Create a group per “domain” e.g. per servers to be deployed that need either affinity or anti-affinity:
  - `nova server-group-create anti-group anti-affinity{/affinity}`
- To use:
  - `nova boot --flavor standard.small --image Ubuntu-Server-14.04 --hint group=anti-group --key-name default node`
- Affinity is only available via CLI/API calls (e.g. HEAT)

# Nova-Compute - Storage Management

## Ephemeral Storage

- Exists only for the life of an instance.
- Persistent across reboots of the guest Operating System.
- Will be deleted when instance is deleted.

## Persistent Volume Storage

- Volumes are persistent virtualized block.
- Independent from instance.
- Volume can be attached to a single instance at a time.
- May be detached or reattached to a different instance while retaining all data, much like an USB drive.

# Provisioning Instances - CLI Ephemeral Disk

## Command to boot your instance from CLI

```
nova boot --image imageid --flavor flavorid --key-name name --  
security-groups name --nic net-id=networkid instancename
```

- **Image id:** list of images available in the image repository (nova image-list or glance image-list)
- **Flavor id:** list of flavors available (nova flavor-list)
- **Key-name:** key pair name from the list of key pairs (nova key pair-list)
- **Security-groups:** security group name with appropriate security group rules assigned. (Nova secgroup-list.)
- **Net-id:** network id from the list of networks. (Neutron net-list)

# Nova Boot From Volume

- Why?
  - Persistent storage (delete the VM, keep the Volume)
  - Performance?
  - IT managed (e.g. storage and storage replication)
- Data needed:
  - Image source type and id (or name)
  - Destination type (volume or local disk)
  - Size (disk in GB)
  - Shutdown goal (preserve/delete)

```
nova boot --flavor standard.small --block-device source=image,id=Ubuntu-  
Server-14.04,dest=volume,size=10,shutdown=preserve,bootindex=0 \  
myInstanceFromVolume
```

# Nova - Meta-data / Cloud-Init

- How do you configure a virtual machine in a cloud after it has been booted from a 'generic' image?
- Web service, managed by Nova with nova-network, managed by Neutron with Neutron based network service
- Can pass: user scripts, ssh keys, configuration data (YAML formatted data)
- Requires that 'generic' images include the cloud-init package
- Alternate model exists as well: "config-drive" (Cloud-Init)
- Same capability for configuration, and same OS package, but provides an auto-generated ISO that is attached to the instance at boot time

# Provisioning Instances - Continued

Inject the cloud-init scripts to the instance

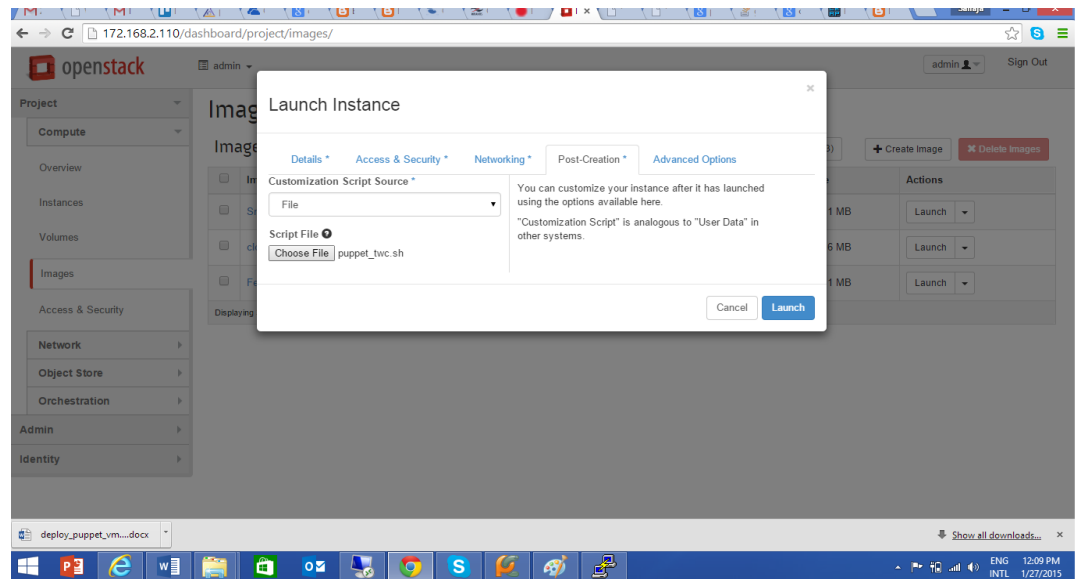
- Userdata flag is used to customize the instance provisioning via cloud-init package. Provide the location of the script to `--user-data` while provisioning
- `nova boot --user-data ./user-data.sh --image imageid --flavor flavorid --key-name name --security-groups name --nic net-id=networkid instancename`
- `user-data.sh` contains a script, e.g. to install puppet, git and python-pip

An example `user-data.sh` script might be:

```
#!/bin/bash
apt-get update -y
apt-get install git
git clone https://github.com/onecloud/todo-app
```

# App Deployment via Dashboard

We can use Post creation tab and add the user-data as a shell script attachment or copy paste the code to the text area available and launch the instance.





# Security Groups

- What is it?
  - Mini ACL based firewall mapped to the VM interface port.
  - Implemented as IPTables on a per-VM linuxbridge bridge.
- Can define Source, Dest, type (ICMP, TCP, UDP)
  - Source (for ingress rules) or Dest (for egress rules) can be either CIDR addr or a security group (even the same one)
  - Can define ports or port ranges for each ACL
- Can define and apply security groups per VM, either at time of creation or after VM boot
- Default rule:
  - Ingress: ICMP (ping), ssh (22), RDP (3389) {note: 22 blocked from greater internet}
  - Egress: Open

# Create Security Groups

- Create a security group called “redis-db”, and allow 6379 TCP

```
nova secgroup-create redis-db "allow ingress Redis"  
nova secgroup-add-rule redis-db tcp 6379 6379 0.0.0.0/0
```

- Now apply the “redis-db” rule to your vm(s)

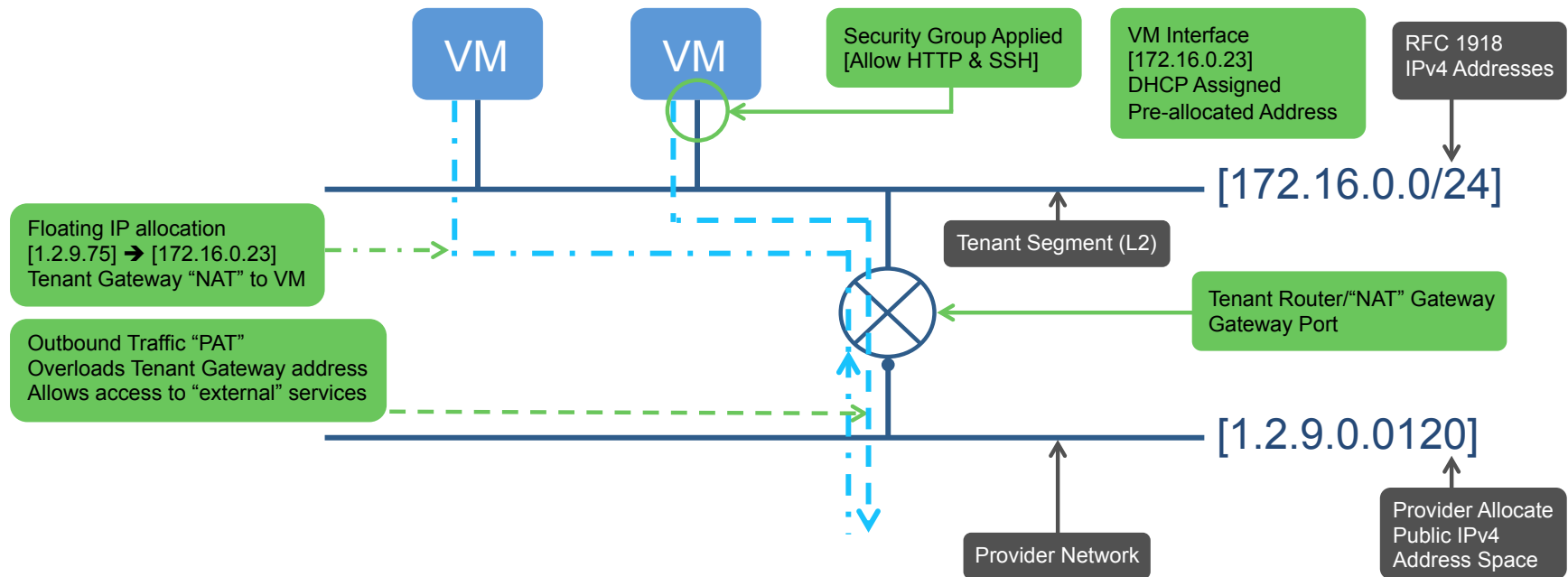
```
nova add-secgroup web-101 redis-db
```

# Network Service Interaction

- Neutron network model
- Network as deployed at TWC
- Attaching a port to a VM
- Floating IPs
- Load Balancing - How to interact (external capability via TWC portal)

# Cloud Network Model

## OpenStack Generic Network



# Dealing with multiple possible Networks

- If a tenant has only a single network, it is not necessary to define it to the nova boot command.
- If however, there are more than one networks `_visible_` to the tenant (perhaps two private networks), then it is necessary to define the network in the boot command
- This is true in Horizon or the GUI as well

# Find a network and associate to a VM

- Find a network:

```
# neutron net-list
```

id	name	subnets
e9bee325-a6f4-49b6-a59d-d26709cfc56f	flat	45b4192c-6e33-4278-8b2b-c387f5ac4fc1 10.1.10.0/24
4aa9927b-4a77-43f0-88de-dcb0f1c8d9a9	private	656f5c91-fe81-440c-abcc-2c31b2a78023 10.0.10.0/24

- Select the network\_id from the network of choice and apply to a nic:

```
nova boot --flavor 3 --image trusty --nic net-id=4aa9927b-4a77-43f0-88de-dcb0f1c8d9a9 net_vm
```

# Find and associate a Floating-IP

- Floating IPs are addresses that are a part of an “external” network that can be associated with a private (tenant-network attached) virtual machine.
- The network function that enables this is call NAT for Network Address Translation
- First find the external network:

```
neutron net-list
```

- If the network isn't obvious (name includes something like Ext or Public) then you can sort through the list looking for a network with “router:external” set to True:

```
neutron net-show Ext-Net | grep router:external | grep True
```

# Create and associate the floating IP

- Create a Floating IP

`neutron floatingip-create Ext-Net`

- Determine the port you want to associate with the floating IP

`neutron port-list | grep IP.OF.TENANT.VM`

- Then associate them.

`neutron floatingip-associate ID_FROM_NEW_FLOATING_IP ID_FROM_PORT`



# BASIC VM DEPLOYMENT

**Create a VM, deploy your app**

# Compute Deployment - “OpenStack” Methods

- 2 Basic Methods
  - CLI
    - nova boot ...
  - Horizon UI
- More integrated methods
  - SDK Based (python leveraging the same capabilities as the CLI tools)
    - Fog.io - Ruby
    - Jclouds - Java
  - Environment based
    - Vagrant - Ruby based

**Launch Instance**

Details \*   Access & Security \*   Networking \*   Post-Creation   Advanced Options

**Availability Zone:**  
nova

**Instance Name: \***  
cloudinit

**Flavor: \***  
m1.tiny

**Instance Count: \***  
1

**Instance Boot Source: \***  
Boot from image

**Image Name:**  
F20-x86\_64 (201.1 MB)  
Select Image  
F20-x86\_64 (201.1 MB)  
UbuntuTrusty (244.6 MB)  
cirros (12.5 MB)

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

**Flavor Details**

Name	m1.tiny
VCPUs	1
Root Disk	1 GB
Ephemeral Disk	0 GB
Total Disk	1 GB
RAM	512 MB




**Project Limits**

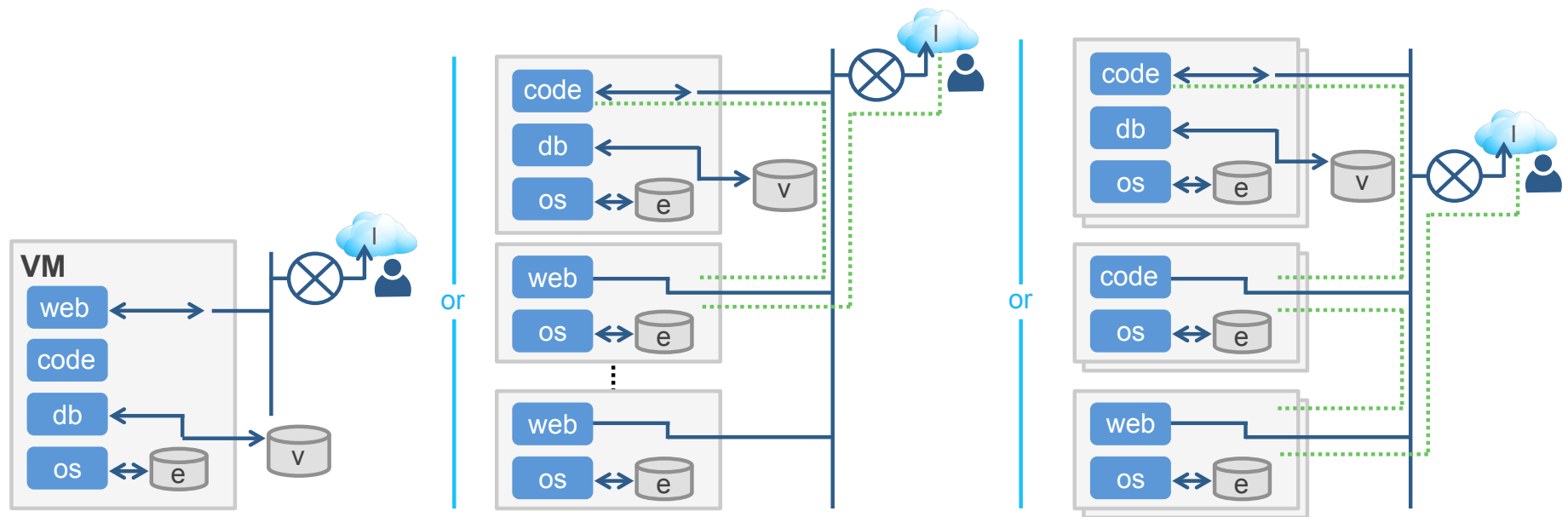
Number of Instances	5 of 10 Used
Number of VCPUs	5 of 20 Used
Total RAM	10,240 of 51,200 MB Used

Cancel   Launch

# Model for Application to be deployed

- Datastore on one node (how to configure, cloud-init?)
- App/interface on another node (configure, etc.)
- Network model
- Security model

Model → need data store =   
View → provides interface =   
Controller → transforms = 

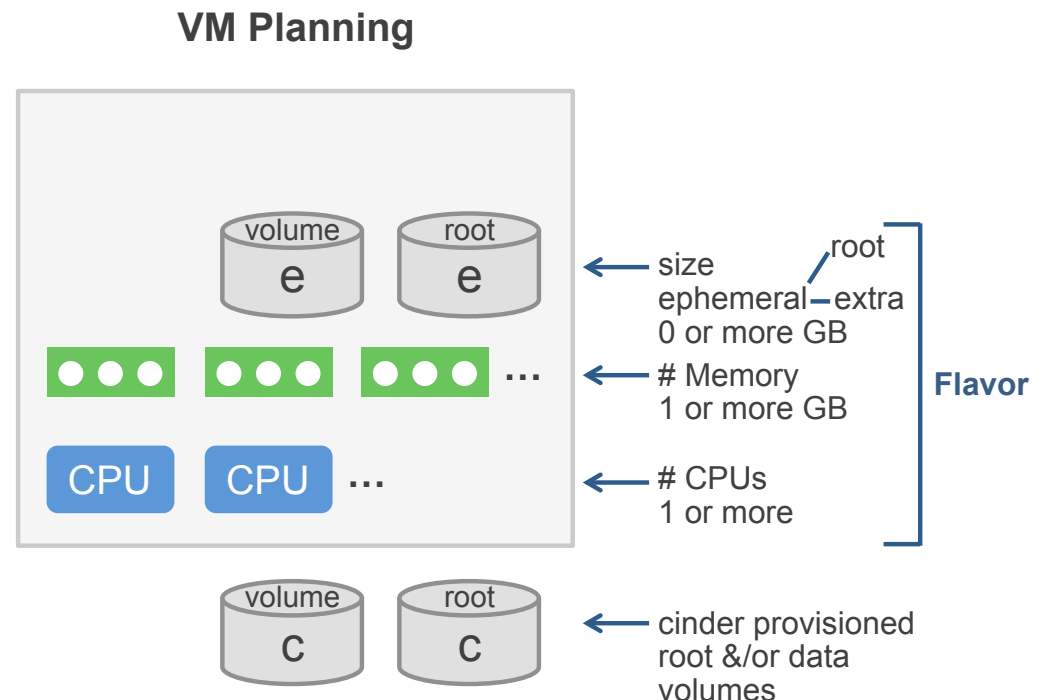


# What would you do with Physical Infra?

- Plan your application based on deployment constraints
  - Number of servers available
  - Type and quantity of storage available (perhaps also performance)
  - Network services needed (security, load balancing, DNS)
- Procure... wait... rack/stack/base-os... deploy!
- How is this different in OpenStack with Nova?

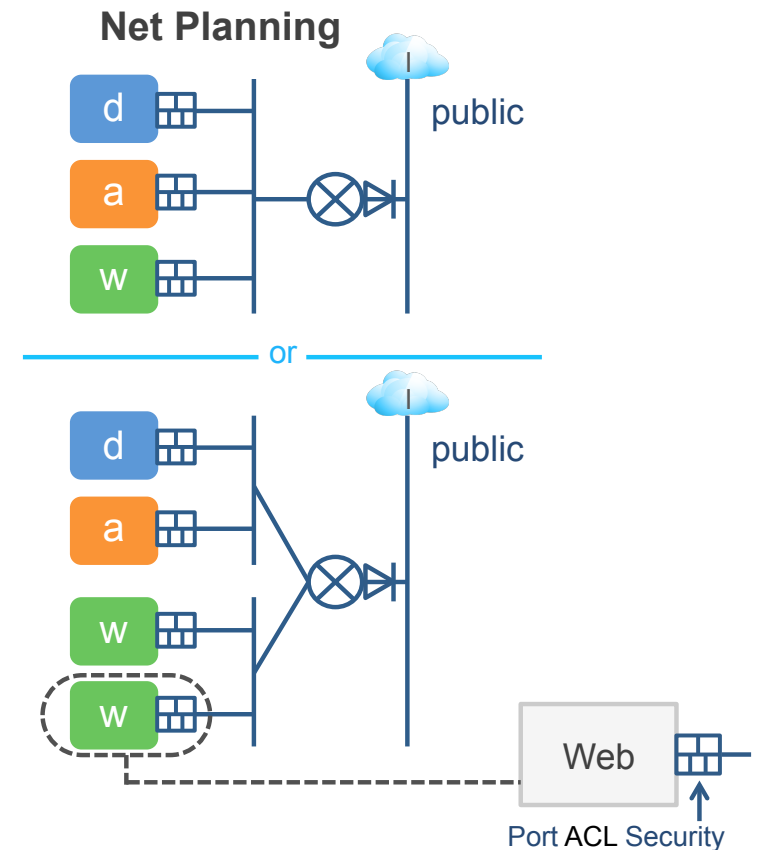
# Planning a VM deployment

- How big a VM do I need for each different service
  - Drives a decision on initial Flavor
    - changing can be done but is not automatic (today)
    - One big VM, many smaller instances
    - Small instance, and add as load increases
  - Affinity/anti-affinity
  - Do I need persistent storage
    - can I separate OS and Data
    - What is my recovery model if the VM dies



# Planning your VM network

- What Network segregation models should I use
  - Do I need to segregate services onto different networks
  - Is FloatingIP for public services enough segregation
- Do I need server load balancing
- What Security groups might have
  - Will I have more than one service on the tenant network
  - Are there groups of devices that should be allowed to talk to each other
  - Will I have separate backend networks that share a router



# How to deploy?

- Horizon UI
  - Graphical interface
  - Wizard to capture critical elements (name, image, keypair, network, etc.)
  - Can automatically create N identical instances
  - No way to re-do a deployment other than re-entering all the data
- CLI
  - Python based command utilities
  - Need to know parameter names (there is online help in the cli tools)
  - Can do an auto-repeat, and can poll for completion
  - Can be scripted to allow for repeating the same operation multiple times
- Heat
  - We will cover in detail later

# Multi-Site Deployments

- Same considerations for basic platform (flavor, disk, etc.)
- Database replication? Old models still apply, or, leverage new database model:
  - Cassandra, provides multi-site auto-replication
  - CouchDB, can be configured in a replicated model
- Don't move golden images if possible (use generic pre-vetted base images and build out)
- Leverage Object Storage, both for application content, and possibly even for application code (auto-replicated)



# Multi-Site – Continued

- Application sync/replication, if possible over the public network, potentially provides the ability to scale out of a private environment to a public, or vice versa
- Consistent naming of objects (tenant networks, VM images, flavors, affinity groups) will simplify scripting multi-site deployments
- For Multi-site deployments, automation via CLI scripts (or other methods to talk to OpenStack APIs), is critical