Introduction to Openstack, an Open Cloud Computing Platform

Libre Software Meeting

10 July 2012

David Butler

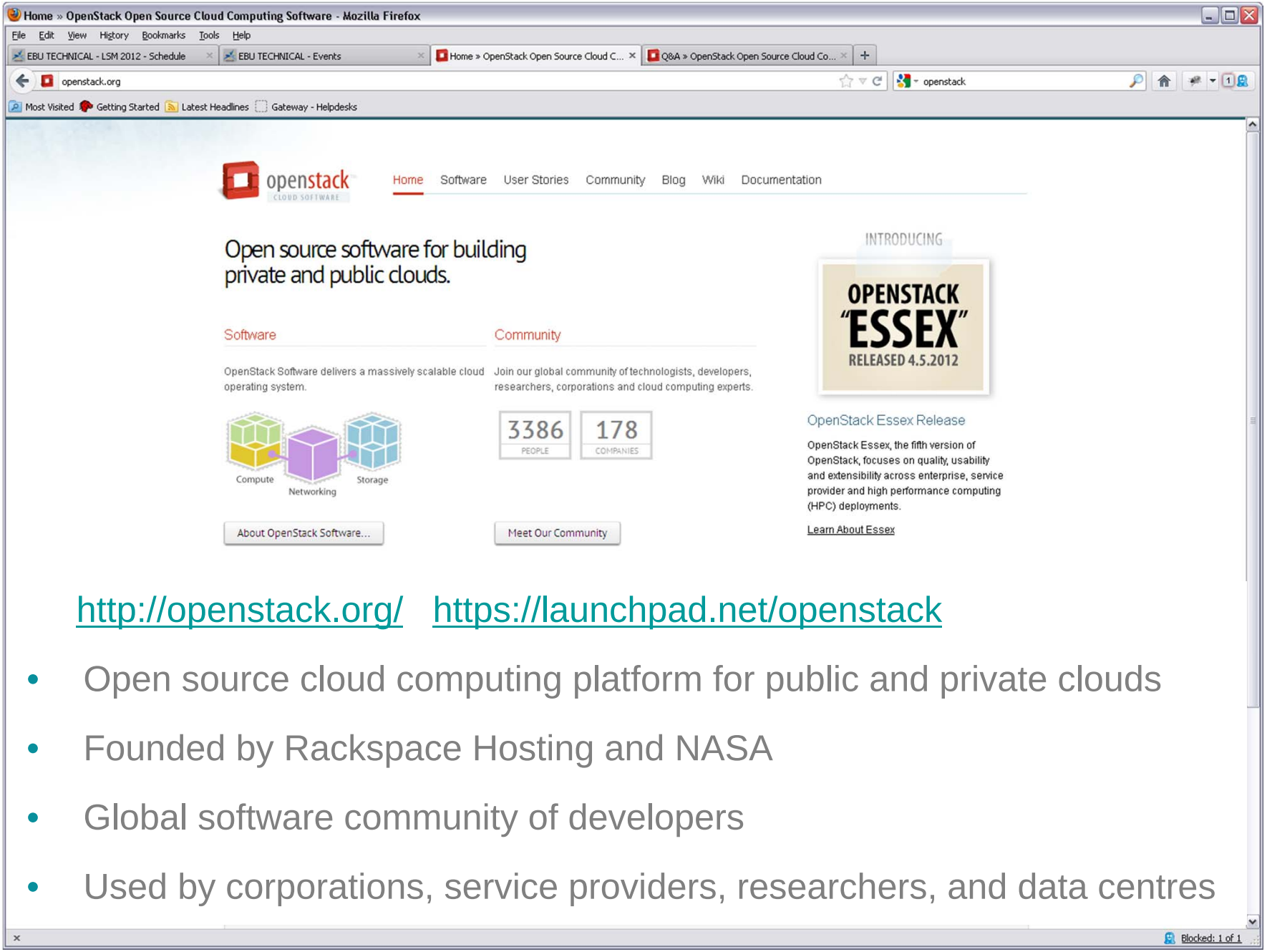
BBC Research & Development

david.butler@rd.bbc.co.uk

Introduction: Libre Software Meeting 2012

- What is covered in this presentation
  - What is Openstack?
  - Why is the BBC interested in cloud?
  - BBC cloud proof of concept
  - Openstack timeline
  - Openstack components
  - Openstack DIABLO
  - Openstack ESSEX
  - Openstack install process

What is OpenStack



http://openstack.org/ https://launchpad.net/openstack

- Open source cloud computing platform for public and private clouds
- Founded by Rackspace Hosting and NASA
- Global software community of developers
- Used by corporations, service providers, researchers, and data centres

Why Cloud in the BBC?

Flexibility

- Easier to change and maintain operations & services
  - Separation of services from hardware

Efficiency

- Better utilisation of our servers and resources
  - Re-use of resources for multiple services

Scalability

- Scale services with demand
  - Re-deploy services and access external resources

Cost

- Lower capital and operations cost ???
  - Lower support cost for business services?

1. Simple File and Document Management

   Automated file transfer capability enabling BBC staff to “send” a file to a cloud location to be securely accessed by other individuals.

2. Streaming Bulk Storage and Manipulation

   Very large file movement service, ideally supporting scripted file manipulation between private and public clouds.

3. On-Demand IaaS/PaaS for Test and Development

   A deployable test and development environment for build, integration and test.

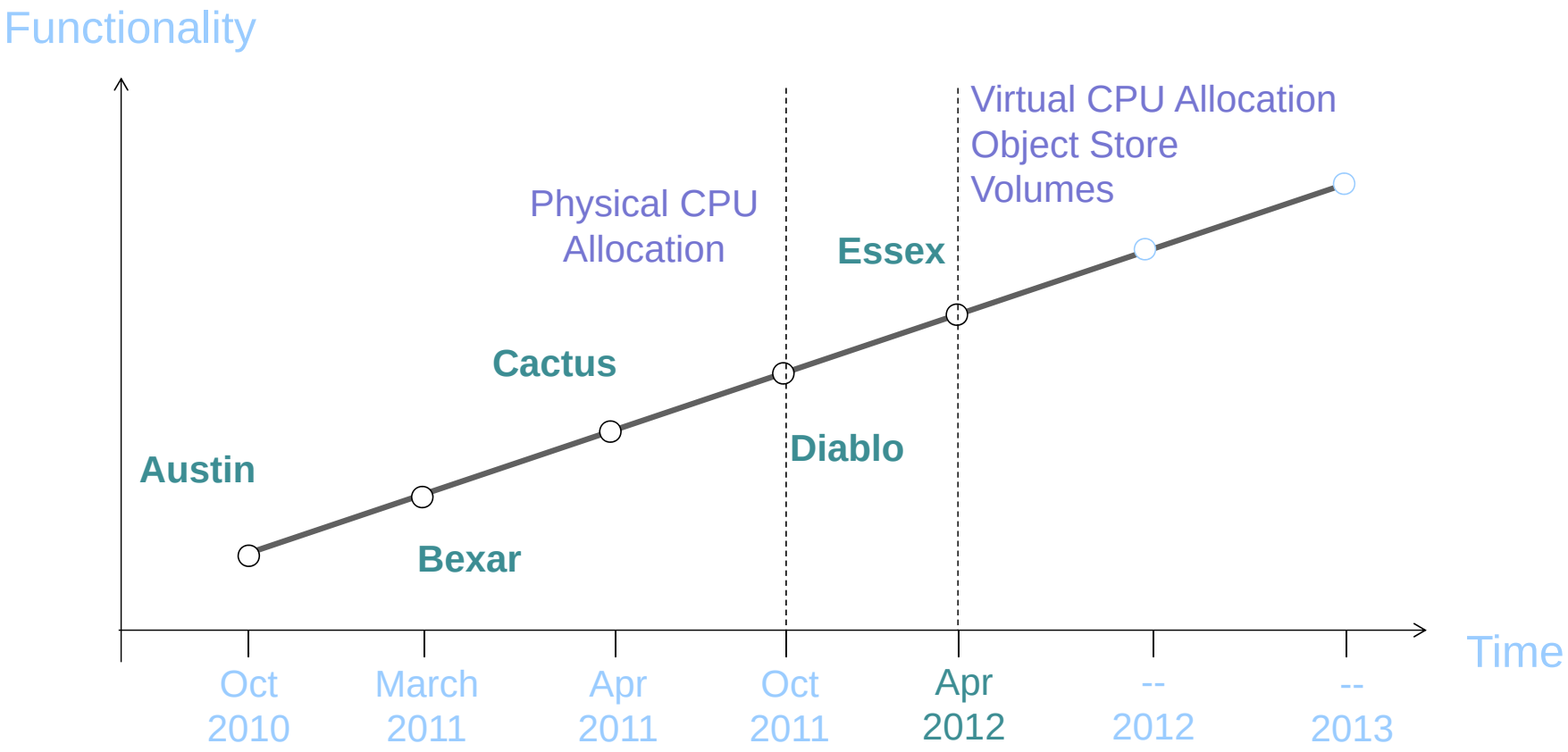
4. Simple Encode/Transcode/Render as a service

   Simple media transcode and/or render workflow “as a service”, to send media and metadata to the cloud through a simple user interface.

5. Automated Processing of Archive Imagery and Video

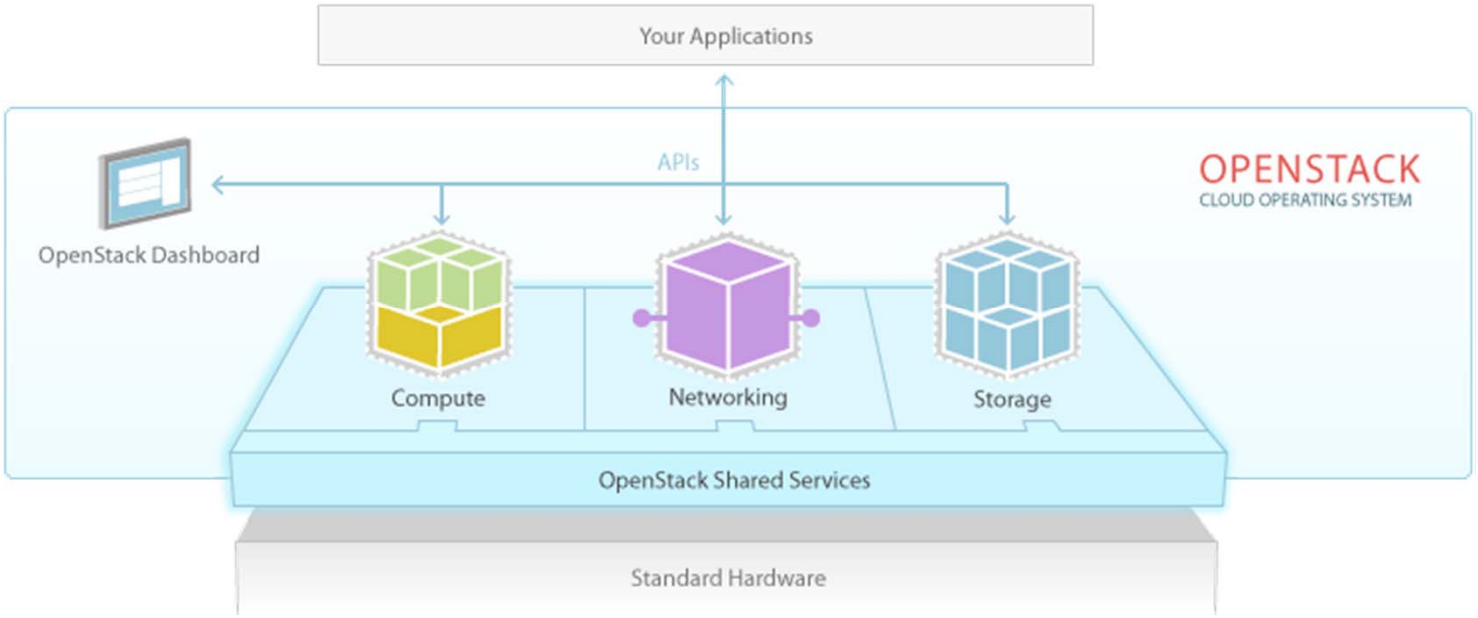
   A secure distributed image processing service with appropriate handling of the master material, transient data and final results.

Openstack: Time Line



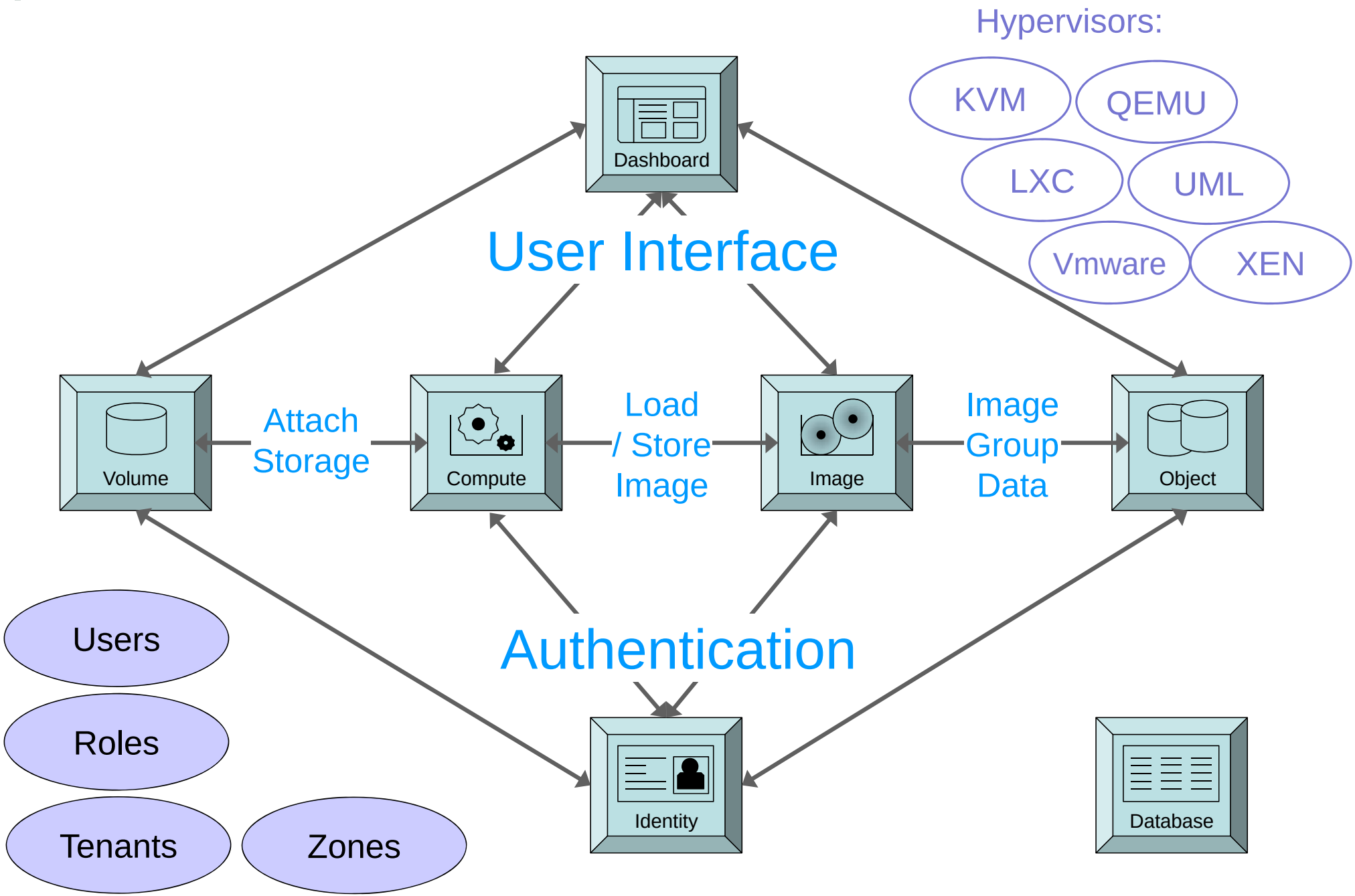
- Openstack is used for real commercial services, e.g. Rackspace etc
- Openstack is used in commercial public cloud service platforms
- Openstack is catching up with commercial private cloud service platforms

OpenStack Components

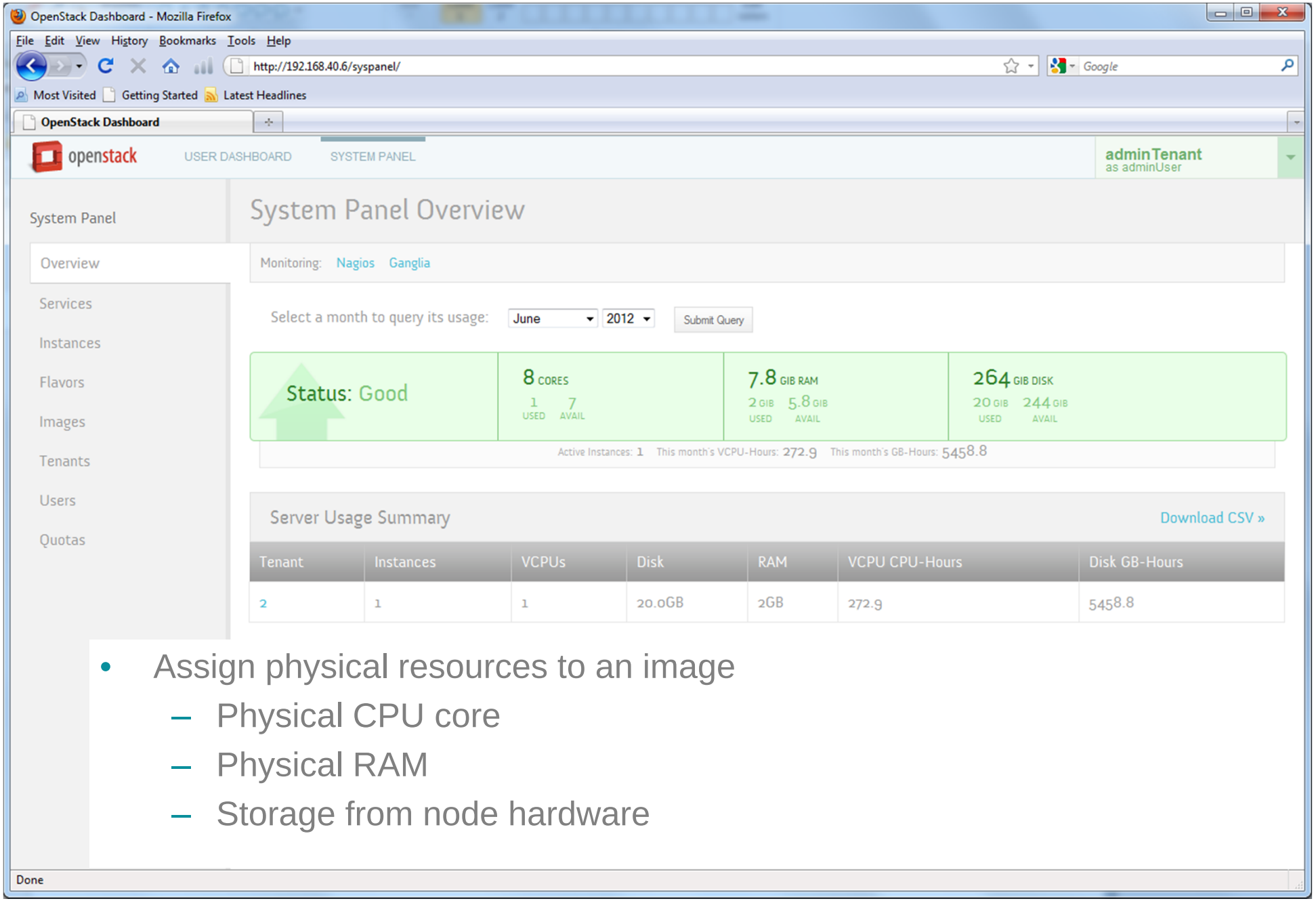


- Compute Components
  - API Server (nova-api)
  - Message Queue (rabbit-mq server)
  - Compute Workers (nova-compute)
  - Network Controller (nova-network)
  - Volume Worker (nova-volume)
  - Scheduler (nova-scheduler)
  - Image Service (glance)
- Dashboard
  - User Interface (horizon)
- Service Object Storage
  - Storage Infrastructure (swift)
- Security and Users
  - Identity Service (keystone)
  - Database (mysql)

How the components work together

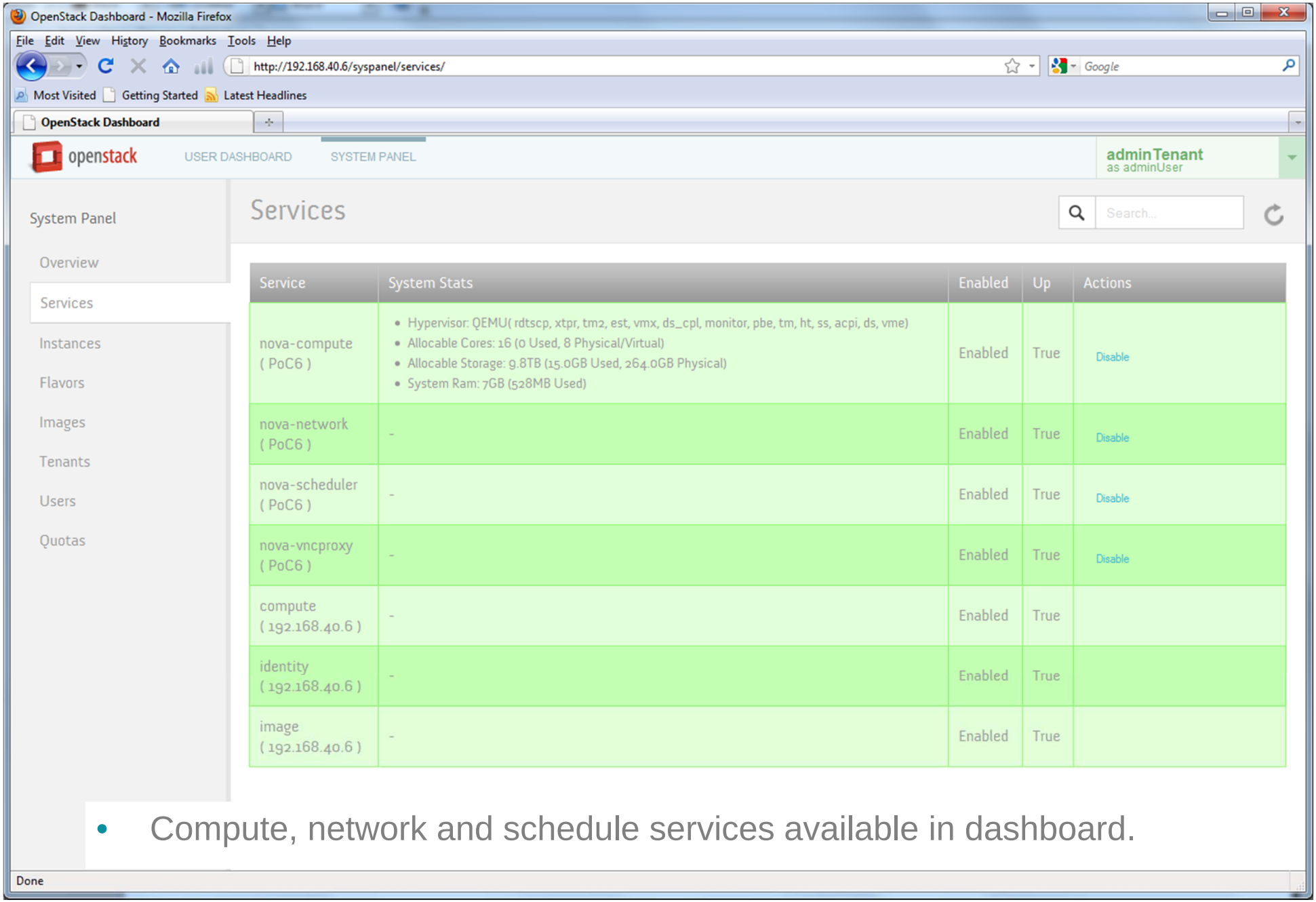


Openstack DIABLO: Resources



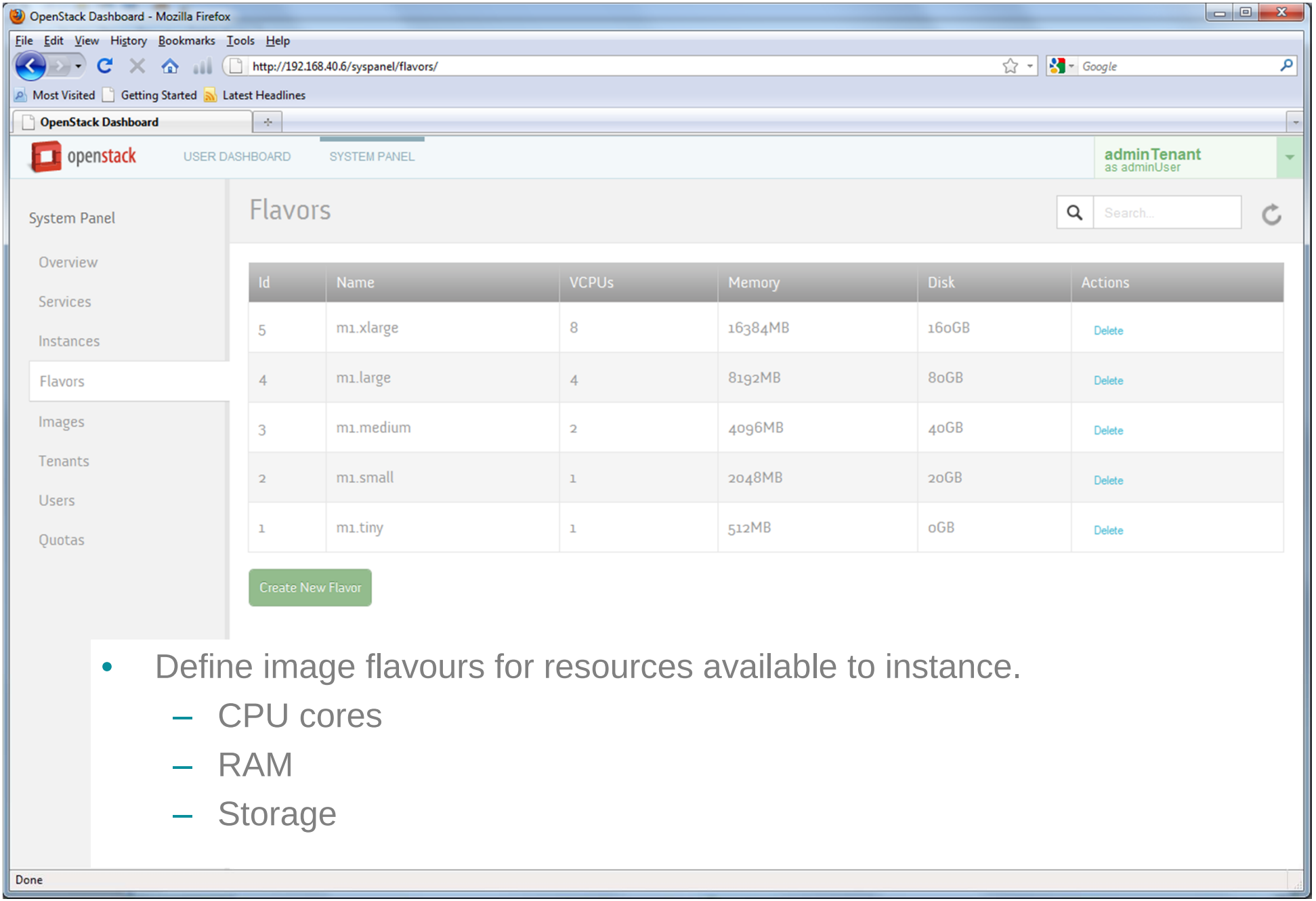
- Assign physical resources to an image
  - Physical CPU core
  - Physical RAM
  - Storage from node hardware

Openstack DIABLO: Services



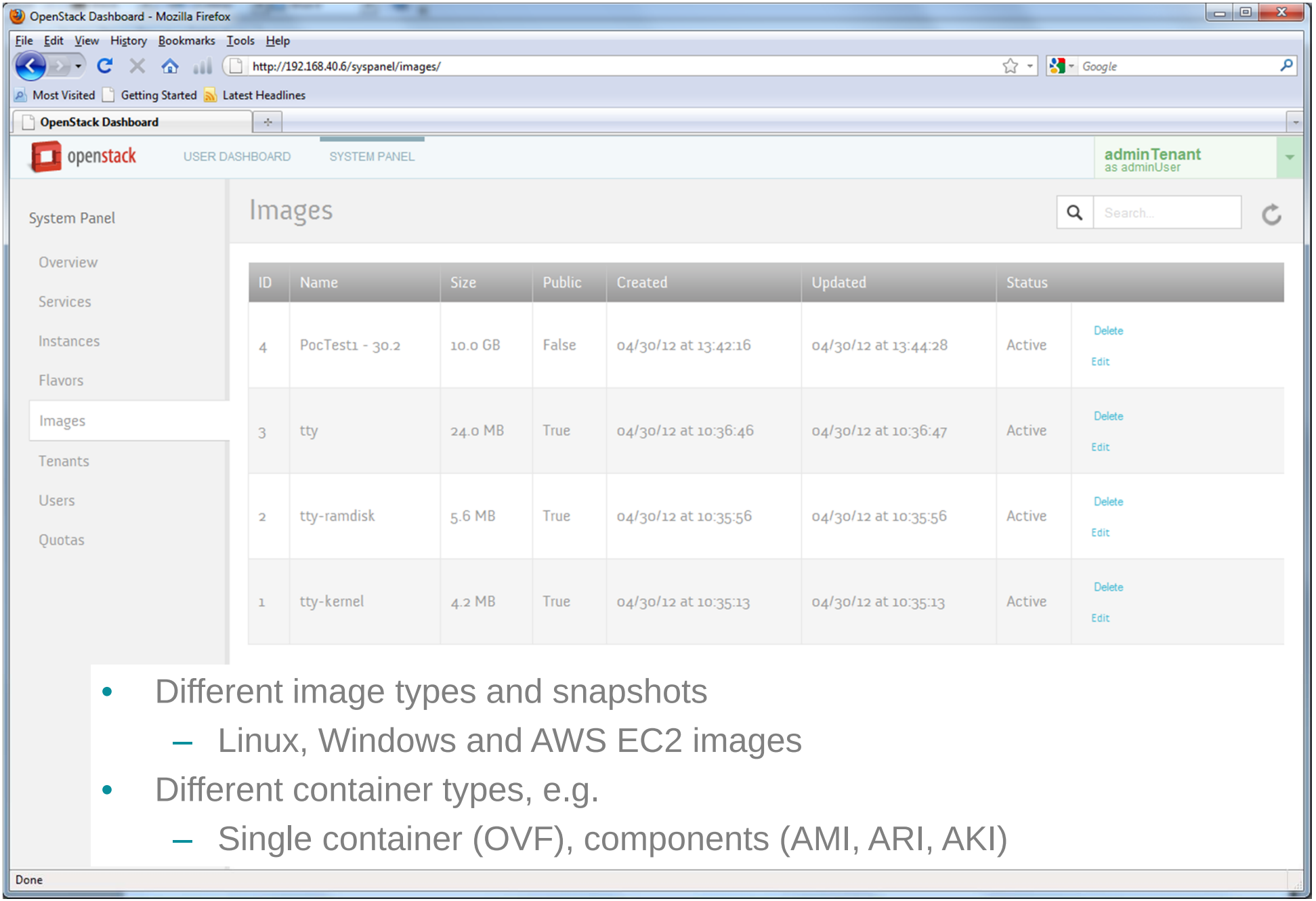
- Compute, network and schedule services available in dashboard.

Openstack DIABLO: Flavours (Assigned Resources)



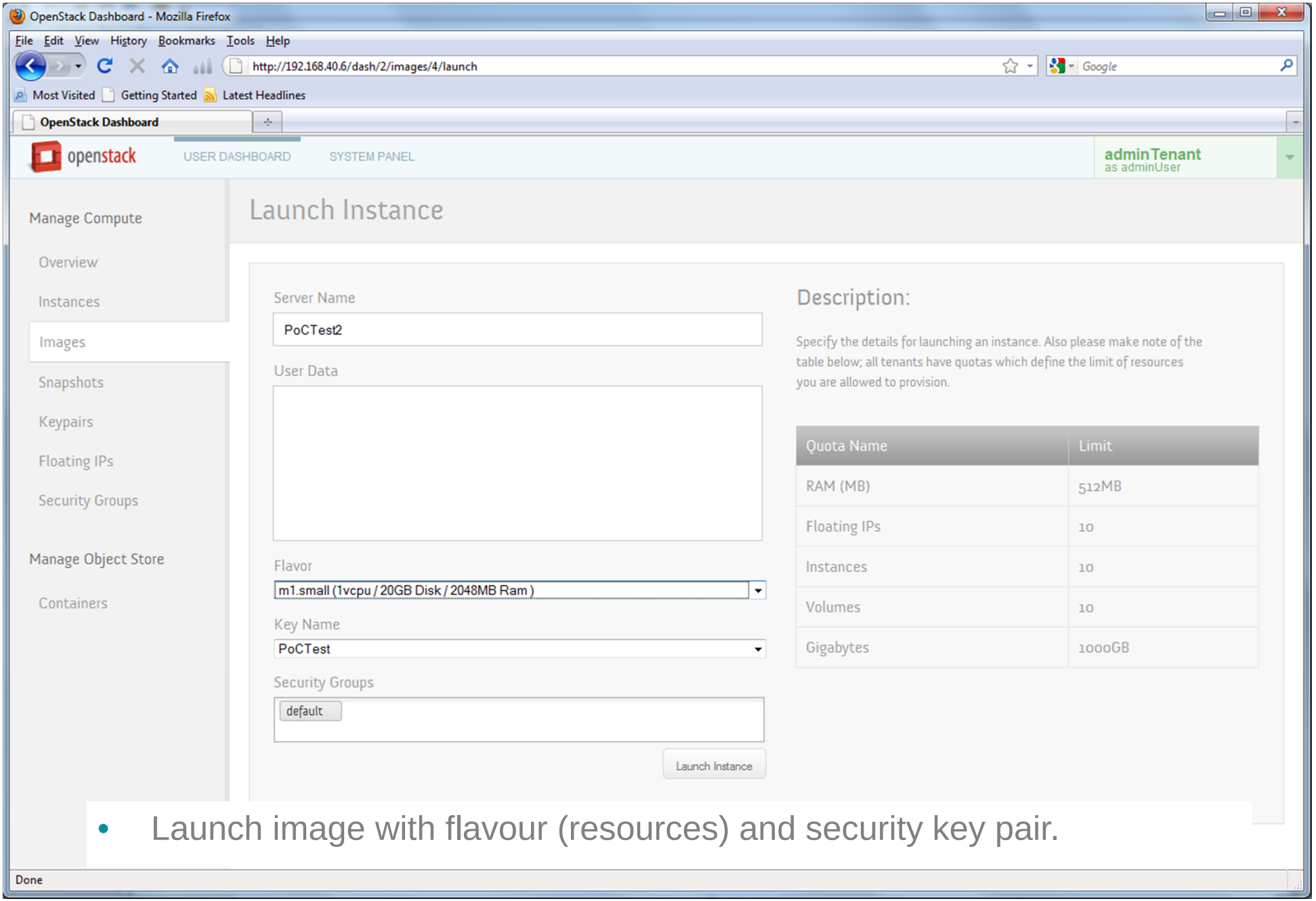
- Define image flavours for resources available to instance.
  - CPU cores
  - RAM
  - Storage

Openstack DIABLO: Images and Snapshots



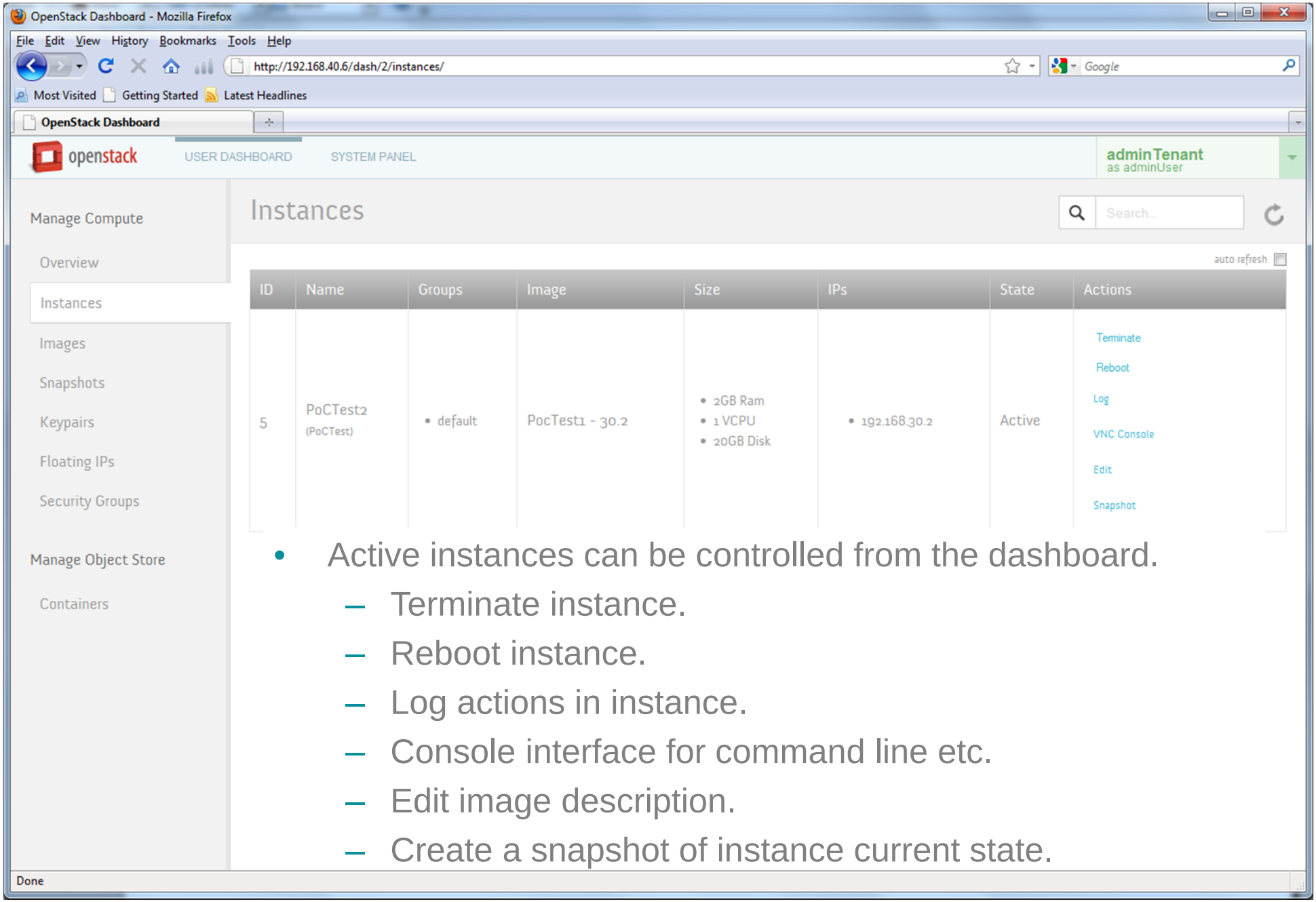
- Different image types and snapshots
  - Linux, Windows and AWS EC2 images
- Different container types, e.g.
  - Single container (OVF), components (AMI, ARI, AKI)

Openstack DIABLO: Launching an Instance



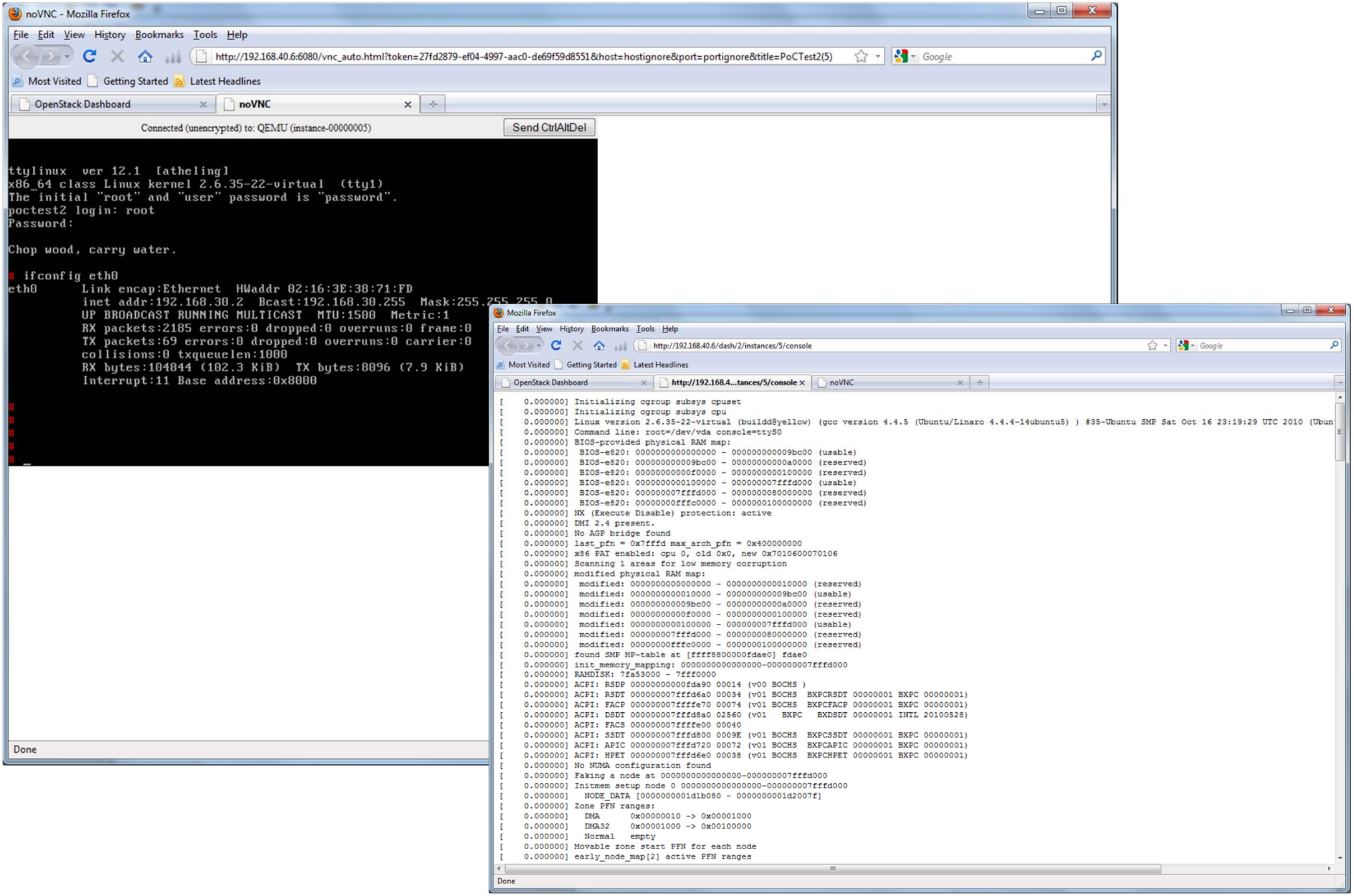
- Launch image with flavour (resources) and security key pair.

Openstack DIABLO: Active Instance

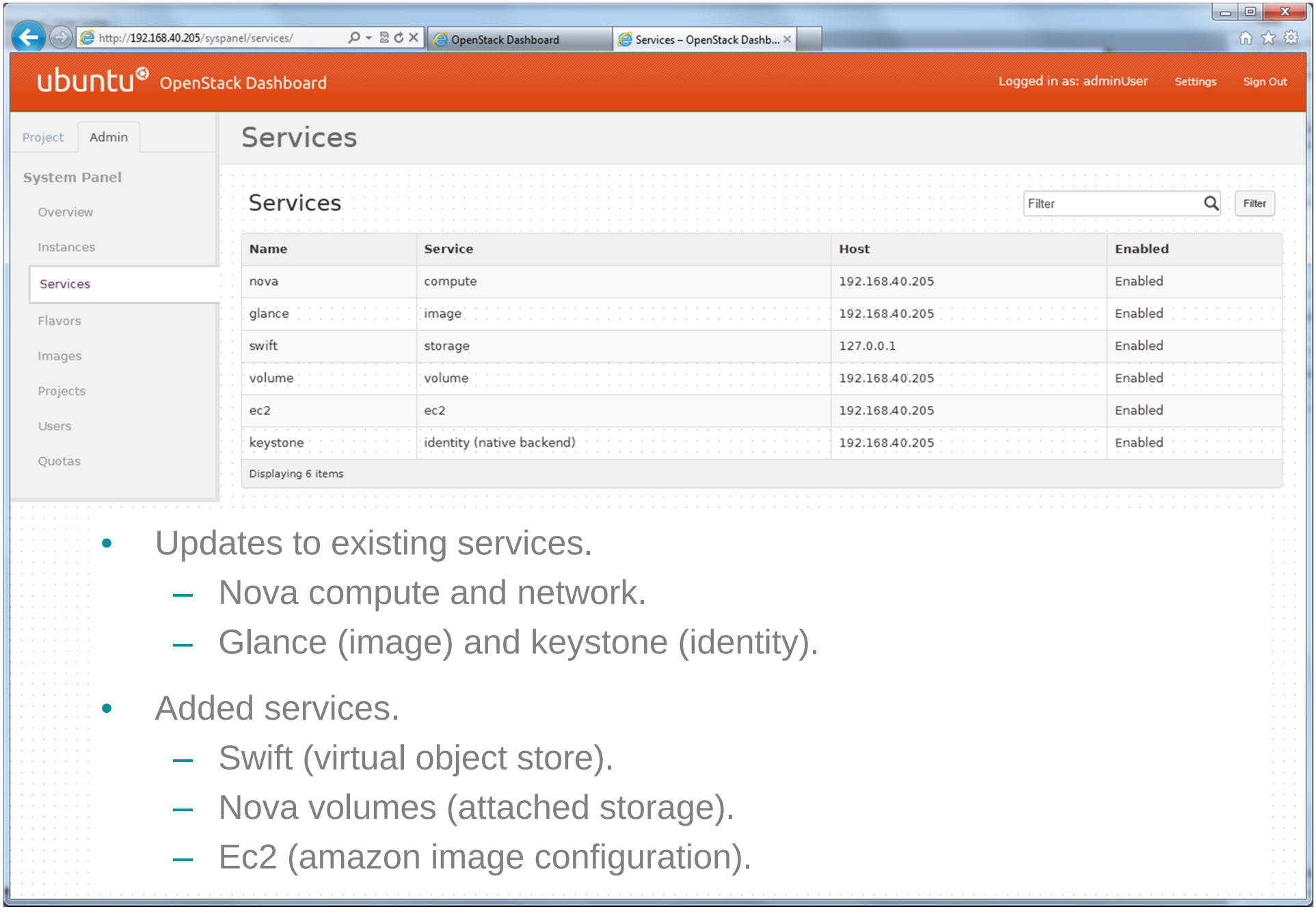


- Active instances can be controlled from the dashboard.
  - Terminate instance.
  - Reboot instance.
  - Log actions in instance.
  - Console interface for command line etc.
  - Edit image description.
  - Create a snapshot of instance current state.

Openstack DIABLO: Console and logging

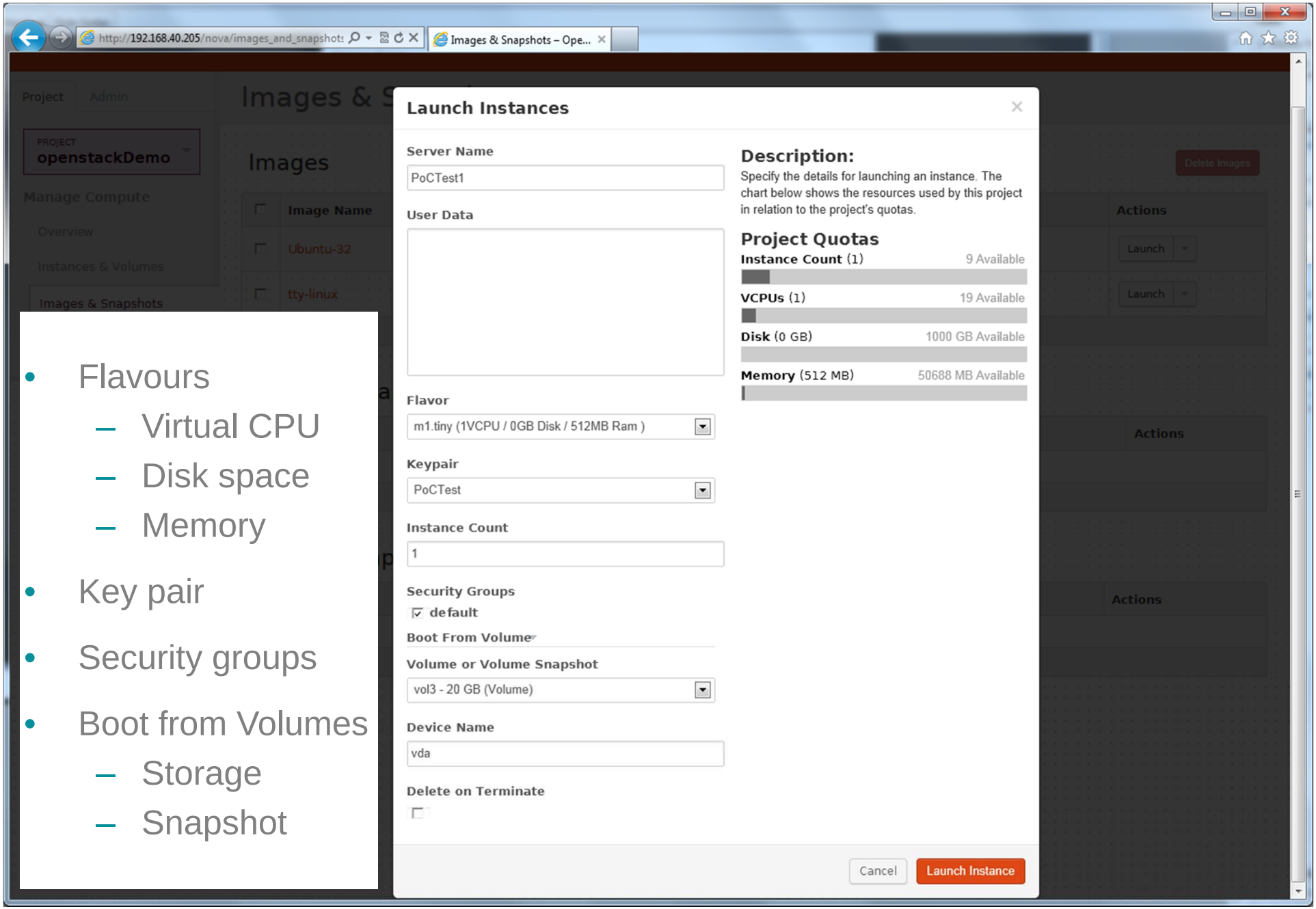


Openstack ESSEX: Services



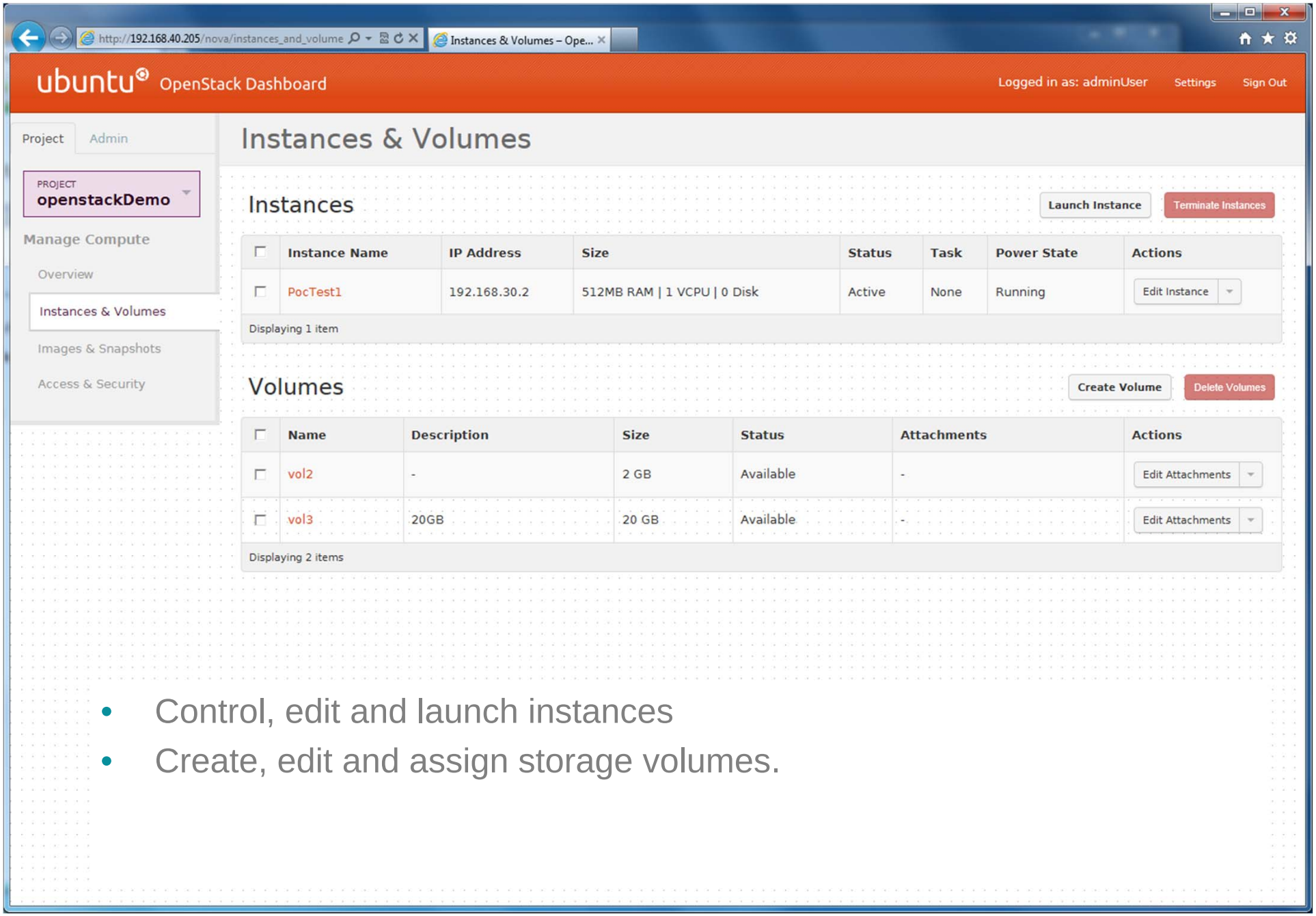
- Updates to existing services.
  - Nova compute and network.
  - Glance (image) and keystone (identity).
- Added services.
  - Swift (virtual object store).
  - Nova volumes (attached storage).
  - Ec2 (amazon image configuration).

Openstack ESSEX: Launching an Image



- Flavours
  - Virtual CPU
  - Disk space
  - Memory
- Key pair
- Security groups
- Boot from Volumes
  - Storage
  - Snapshot

Openstack ESSEX: Instance and volumes



- Control, edit and launch instances
- Create, edit and assign storage volumes.

Openstack: Setup

http://docs.openstack.org/essex/openstack-compute/install/apt/content/

1. Update all packages
2. Enable Virtualisation Technology in BIOS
3. Install the network time package (ntp).
4. Install the database (MySQL).
5. Install the Identity Service (Keystone).
6. Configure the Identity Service and Endpoints.
7. Install the Image Service (Glance).
8. Configure the Image Service.
9. Install Compute (Nova).
10. Configure Compute networking
11. Create and initialize the Compute database with MySQL.
12. Create and add VM images.
13. (Optional) Install Openstack Object Storage (Swift).
14. Install the Openstack Dashboard.

Complexity: Identify service

Create Tenants, Users and Roles

```
# Create a default tenant, openstackDemo
$ keystone --token SECRET1234TOKEN --endpoint http://192.168.0.210:35357/v2.0 tenant-create --name openstackDemo -
-description "Default Tenant" --enabled true

# Create a default user named adminUser
$ keystone --token SECRET1234TOKEN --endpoint http://192.168.0.210:35357/v2.0 user-create --tenant_id
dd4a5cd309cf4ae2be0560e6968f88c5 --name adminUser --pass password --enabled true

# Create the default role, admin
$ keystone --token SECRET1234TOKEN --endpoint http://192.168.0.210:35357/v2.0 role-create --name admin
```

Create Services and Endpoints

```
$ keystone --token SECRET1234TOKEN --endpoint http://192.168.0.210:35357/v2.0/ service-create --name=keystone --
type=identity --description="Keystone Identity Service"

$ keystone --token SECRET1234TOKEN --endpoint http://192.168.0.210:35357/v2.0/ endpoint-create --region RegionOne
--service_id=fc2a447e542d4f209c5b34a707a641ef --publicurl=http://192.168.0.210:5000/v2.0 --
internalurl=http://192.168.0.210:5000/v2.0 --adminurl=http://192.168.0.210:35357/v2.0
```

Create Tokens

```
$ curl -d '{"auth": {"tenantName": "adminTenant", "passwordCredentials":{"username": "adminUser", "password":
"password"}}}' -H "Content-type: application/json" http://192.168.0.210:35357/v2.0/tokens | python -mjson.tool
{
    "access": {
        "serviceCatalog": {},
        "token": {
            "expires": "2012-05-11T11:22:38Z",
            "id": "d8510902060f497a92f52aec2777c5d1"
        },
}
```

Complexity: Starting multiple services

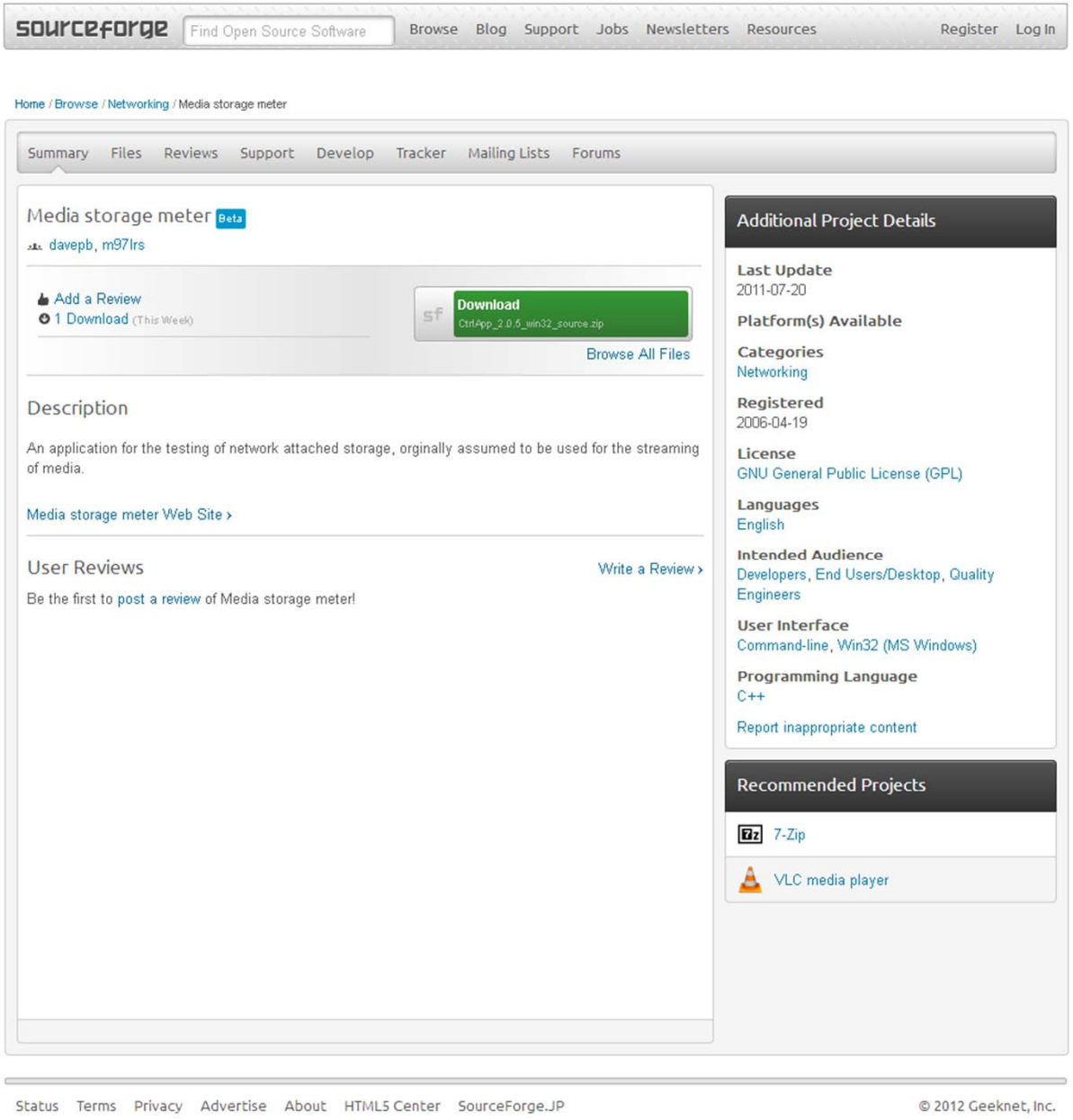
```
sudo start nova-network
source ~/creds/openrc
sudo service mysql restart
sudo service keystone restart
sudo service glance-registry restart
sudo service glance-api restart
sudo service rabbitmq-server restart
sudo service iscsitarget start
sudo service open-iscsi start
sudo restart nova-api
sudo restart nova-compute
sudo restart nova-network
sudo restart nova-scheduler
sudo restart nova-consoleauth
sudo restart nova-volume
sudo restart nova-cert
sudo restart libvirt-bin
sudo /etc/init.d/rabbitmq-server restart
sudo nova-manage service list
Binary           Host                                 Zone             Status     State Updated_At
nova-consoleauth PoC5                                 nova             enabled    :-)   2012-05-29 15:40:54
nova-compute     PoC5                                 nova             enabled    :-)   2012-05-29 15:40:59
nova-network     PoC5                                 nova             enabled    :-)   2012-05-29 15:40:56
nova-scheduler   PoC5                                 nova             enabled    :-)   2012-05-29 15:40:54
nova-volume      PoC5                                 nova             enabled    :-)   2012-05-29 15:40:56
nova-cert        PoC5                                 nova             enabled    :-)   2012-05-29 15:40:59
sudo service rsync restart
sudo swift-init main restart
sudo swift-init rest restart
sudo /etc/init.d/apache2 restart
sudo restart nova-api
```

Summary

- Openstack is in use now for real commercial services.
- Functionality is increasing with each new version.
- Supports different hypervisors and containers.
- Compatible with AWS ec2 images.
- Complex to setup.
- Most public cloud offerings use openstack components.
- Some private clouds support or plan to support openstack images.
- Openstack is becoming an open standard?

Questions

Testing your own storage using Media Storage Meter



http://sourceforge.net/projects/msmeter/