

*NOVA (Compute)*

Nova compute or the king service **provides a platform** on which we are going to **run our guest machines**; It’s the virtual machine **provisioning and management module** that defines drivers that interact with underlying virtualization. It provides a **Control plane** for an underlying hypervisor.  Each hypervisor requires a separate Nova Instance. Nova **supports almost all hypervisors** known to man.

*SWIFT (Object Storage)*

Swift Offers **cloud storage software**, Look at it as **Dropbox or Google drive**, as they are **not attached to servers**, they are individual addressable objects. It’s built for scale and optimized for durability, availability, and concurrency across the entire data set. Swift is ideal for storing unstructured data that can grow without bound. Swift provides redundancy checksum for files, Files are stored as segments and a manifest file tracks them.

*HORIZON (Dashboard)*

Horizon is the **Dashboard to Openstack, your eyes and ears**. It provides a web based user interface to OpenStack services including Nova, Swift, Keystone etc.

*GLANCE (Image Service)*

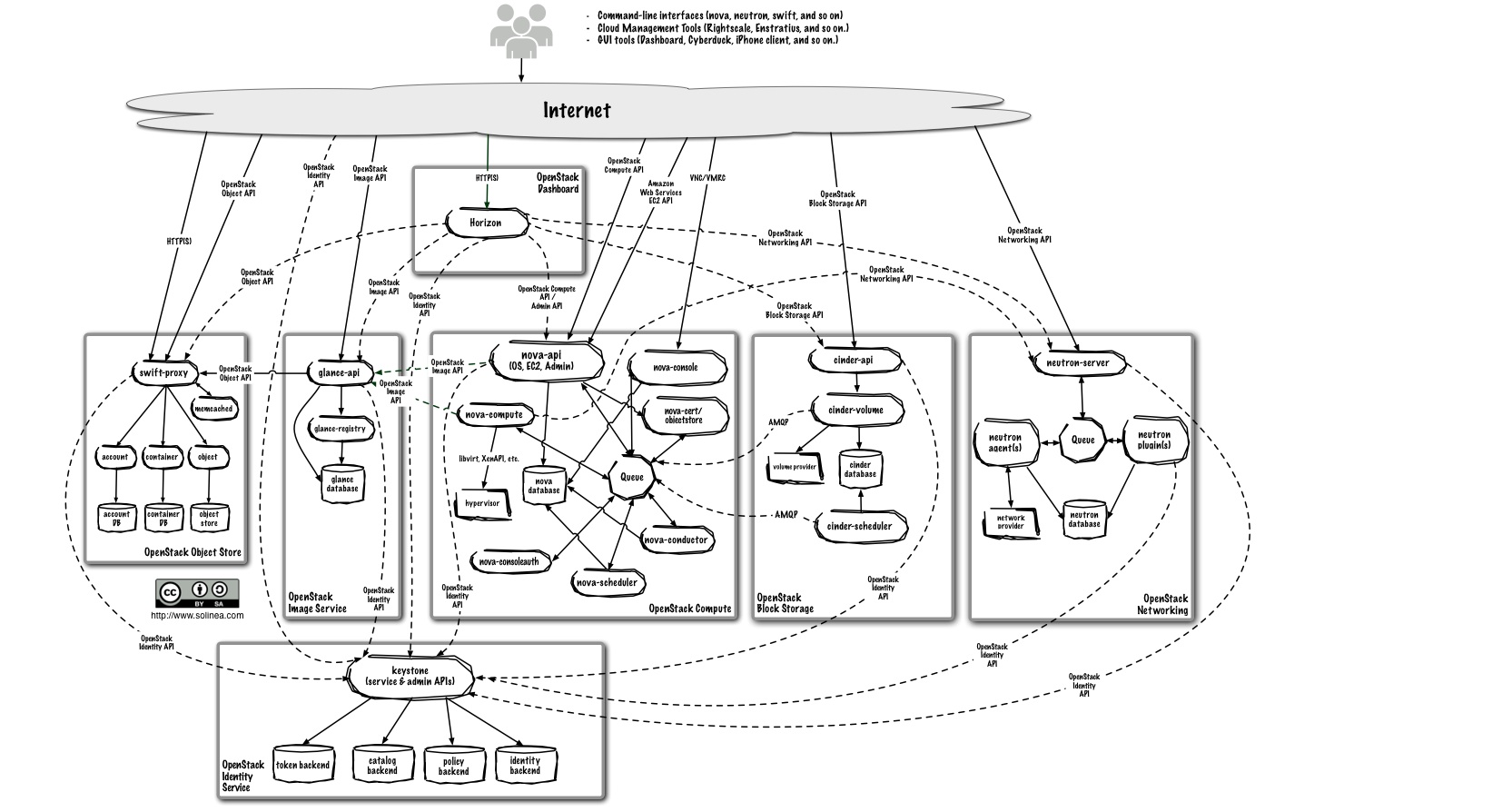
In simple words glance is the **Image Registry**, it stores and Manage our guest (VM) images, Disk Images, snap shots etc. It also **contains prebuilt VM templates** so that you can try it on the fly. Instances are booted from our glance image registry. **User can create custom images and upload them** to Glance later reuse. A feature of Glance is to **store images remotely so to save local disk space**.

*KEYSTONE (Identity)*

It’s the **main authentication and authorization service** as I heard from someone, it’s the a who are you and what do you want service. It authorizes

* Users
* Services
* Endpoints

Keystone uses **tokens** for authorization and **maintains Session state**



We can Access Vietual Machines through  
API, Command Line Interface and Horizon

Authentication Process

Username and Password is authenticated by Key Stone.

Key Stone Authenticate the User and generates a Token.

That Token sticks with the user and allows him to access the services.

Rabbit Message Queue Server   
- Queue is generated to pile up the generated requests.

Nova contains the following Services:

1. Nova ConsoleAuth - This Authorizes the user to allow him to use the Nova Services.
2. Nova-Cert – This service Generates the Security certificate for the user.
3. Nova-Compute – This service is used to do all the computation requested by the users.
4. Nova-Conductor – This service is used to pass the request and response.
5. Nova-Scheduler – This service is used to schedule the requests in a queue given by the users.
6. Nova-API – It is a Programming Interface which enables any program or user to access all the services given by NOVA.

Q and A Session

Q1. Cloud VS Virtualization

Is cloud can be independent or data center

Difference between full and Para Virtualization

Difference between NAS, SAS and DAS

What openstack is IAAS not TAAS

What Kind of parti

Why cloud called dynamically scalable?

<http://www.slideshare.net/gpaterno1/openstack-explained>

**Outline**

* Motivation
* Problem Statement
* Storage: Google File System
* Processing: Map Reduce
* Benchmarks
* Conclusions

**Motivation**

* Buy superfast, ultrareliable hardware?
  + Ultra-expensive
  + Controlled by third party
  + Internals can be hidden and proprietary
  + Hard to predict scalability
  + Fails less often, but still fails!
  + Not suitable solution on the market

**Define your own case study initially having requirement of ext4 file system and now requirement of having own file system.**

* Use commodity hardware? Benefits:

**The need of the File System for retrieving the data and how really the file system affects the performance of the data retrieval than the cloud?**

**The performance of the running cloud with smaller chunk sizes as well as the consequences for the same.**

**Security of Cloud Computing**

Security Services: Confidentiality, Availability, Integrity

**Give me the example for the loss of control as well as lack of trust as well as multi tenancy related to the security in the distributed environment. (with Respect to MySQL)**

* In the distributed environment, user loses the control of the files uploaded on the server. All the files can be seen and managed by the vendor. Manipulation can be done from the management side as well.
* User is restricted to the functions allowed by the vendor.
* All the database access is there with the vendor as well. He can easily get into the database without the user information.
* Vendor can check the log and can keep a track of all the activities done on the distributed environment and this can breech the privacy of the user.
* SSH allows the other user to run malicious Shell Scripts on our machines.
* Using SSH, vendor can restrict our access to other resources on the web using IPTABLES
* Using SSH, vendor can open any port for backdoors to get access to our machines.
* Multiple users are there on the same hardware, anyone can be the attacker and can put other projects down.