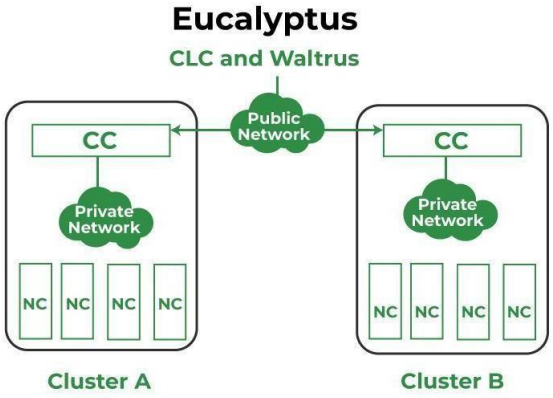
4.4. Cloud Software Environments:

Eucalyptus is a Linux-based open-source software architecture for cloud computing and also a storage platform that implements Infrastructure a Service (IaaS). It provides quick and efficient computing services. Eucalyptus was designed to provide services compatible with Amazon's EC2 cloud and Simple Storage Service(S3).



Eucalyptus Architecture

Eucalyptus CLIs can handle Amazon Web Services and their own private instances. Clients have the independence to transfer cases from Eucalyptus to Amazon Elastic Cloud. The virtual ization layer oversees the Network, storage, and Computing. Occurrences are isolated by hardware virtualization.

Important Features are:-

1. **Images**: A good example is the Eucalyptus Machine Image which is a module software bundled and uploaded to the Cloud.
2. **Instances**: When we run the picture and utilize it, it turns into an instance.
3. **Networking**: It can be further subdivided into three modes: Static mode(allocates IP address to instances), System mode (assigns a MAC address and imputes the instance's network interface to the physical network via NC), and Managed mode (achieves local network of instances).
4. **Access Control:** It is utilized to give limitations to clients.
5. **Elastic Block Storage**: It gives block-level storage volumes to connect to an instance.
6. **Auto-scaling and Load Adjusting**: It is utilized to make or obliterate cases or administrations dependent on necessities.

Components of Architecture

- **Node Controller** is the lifecycle of instances running on each node. Interacts with the operating system, hypervisor, and Cluster Controller. It controls the working of VM instances on the host machine.
- **Cluster Controller** manages one or more Node Controller and Cloud Controller simultaneously. It gathers information and schedules VM execution.
- **Storage Controller (Walrus)** Allows the creation of snapshots of volumes. Persistent block storage over VM instances. Walrus Storage Controller is a simple file storage system. It stores images and snapshots. Stores and serves files using S3(Simple Storage Service) APIs.
- **Cloud Controller** Front-end for the entire architecture. It acts as a Complaint Web Services to client tools on one side and interacts with the rest of the components on the other side.

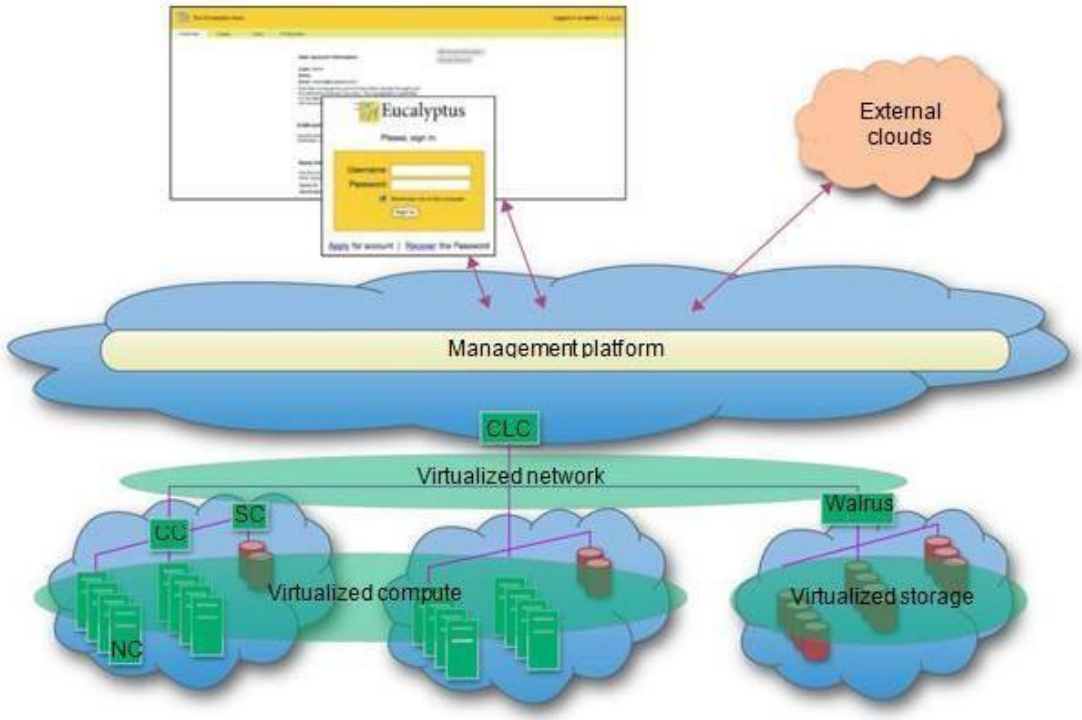
Operation Modes of Eucalyptus

- **Managed Mode**: Numerous security groups to users as the network is large. Each security group is assigned a set or a subset of IP addresses. Ingress rules are applied through the security groups specified by the user. The network is isolated by VLAN between Cluster Controller and Node Controller. Assigns two IP addresses on each virtual machine.
- **Managed (No VLAN) Node:** The root user on the virtual machine can snoop into other virtual machines running on the same network layer. It does not provide VM network isolation.
- **System Mode:** Simplest of all modes, least number of features. A MAC address is assigned to a virtual machine instance and attached to Node Controller's bridge Ethernet device.
- **Static Mode**: Similar to system mode but has more control over the assignment of IP address. MAC address/IP address pair is mapped to static entry within the DHCP server. The next set of MAC/IP addresses is mapped.

Advantages Of the Eucalyptus Cloud

1. Eucalyptus can be utilized to benefit both the eucalyptus private cloud and the eucalyptus public cloud.
2. Examples of Amazon or Eucalyptus machine pictures can be run on both clouds.
3. Its API is completely similar to all the Amazon Web Services.
4. Eucalyptus can be utilized with DevOps apparatuses like Chef and Puppet.
5. Although it isn't as popular yet but has the potential to be an alternative to OpenStack and CloudStack.
6. It is used to gather hybrid, public and private clouds.

7. It allows users to deliver their own data centers into a private cloud and hence, extend the services to other organizations.

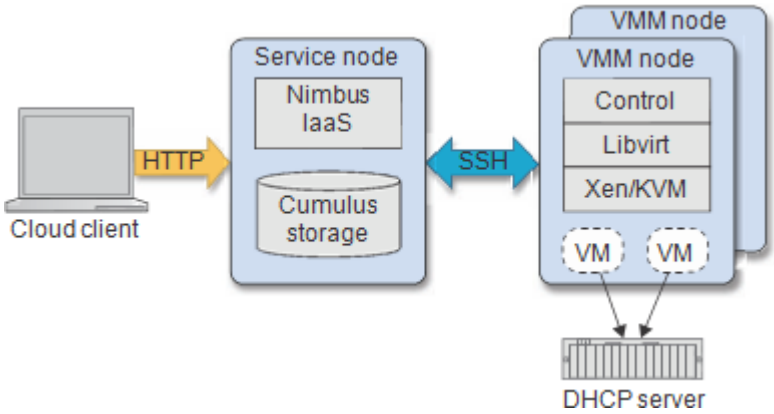


The Eucalyptus architecture for VM image management.

Nimbus:

Nimbus Nimbus is a toolkit that, once included in the collection, provides infrastructure such as a cloud of service to its client through the WSRF-based web service APIs or Amazon EC2 WSDL. Nimbus is a free and open source software, subject to the requirements of the Apache License, version 2.

Nimbus supports both the Xen and KVM hypervisors as well as the portable device organizers Portable Batch System and Oracle Grid Engine. Allows the submission of customized visual clusters for content. It is adjustable in terms of planning, network rental, and accounting usage.



Open Stack

OpenStack is an open - source cloud operating system that is increasingly gaining admiration among data centers. This is because OpenStack provides a cloud computing platform to handle enormous computing, storage, database and networking resources in a data center. In simple way we can say, OpenStack is an open source highly scalable cloud computing platform that provides tools for developing private, public or hybrid clouds, along with a web interface for users to access resources and admins to manage those resources.

Put otherwise, OpenStack is a platform that enables potential cloud providers to create, manage and bill their custom-made VMs to their future customers. OpenStack is free and open, which essentially means that everyone can have access to its source code and can suggest or make changes to it and share it with the OpenStack community. OpenStack is an open-source and freely available cloud computing platform that enables its users to create, manage and deploy virtual machines and other instances. Technically, OpenStack provides Infrastructure-as-a-Service (IaaS) to its users to enable them to manage virtual private servers in their data centers.

OpenStack provides the required software tools and technologies to abstract the underlying infrastructure to a uniform consumption model. Basically, OpenStack allows various organisations to provide cloud services to the user community by leveraging the organization's pre-existing infrastructure. It also provides options for scalability so that resources can be scaled whenever organizations need to add more resources without hindering the ongoing processes.

The main objective of OpenStack is to provide a cloud computing platform that is:

- Global
- Open-source
- Freely available
- Easy to use
- Highly and easily scalable
- Easy to implement
- Interoperable

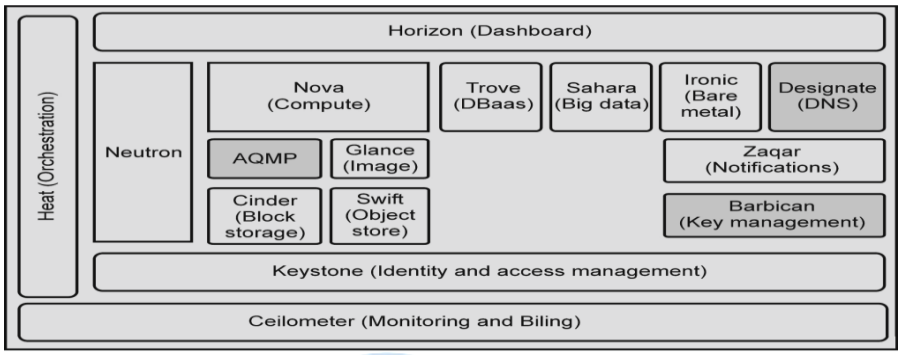
OpenStack is for all. It satisfies the needs of users, administrators and operators of private clouds as well as public clouds. Some examples of open-source cloud platforms already available are Eucalyptus, OpenNebula, Nimbus, CloudStack and OpenStack, which are used for infrastructure control and are usually implemented in private clouds.

Components of OpenStack

OpenStack consists of many different components. Because OpenStack cloud is open - source, developers can add components to benefit the OpenStack community. The following are the core components of OpenStack as identified by the OpenStack community:

- **Nova :** This is one of the primary services of OpenStack, which provides numerous tools for the deployment and management of a large number of virtual machines. Nova is the compute service of OpenStack.
- **Swift :** Swift provides storage services for storing files and objects. Swift can be equated with Amazon's Simple Storage System (S3).
- **Cinder :** This component provides block storage to Nova Virtual Machines. Its working is similar to a traditional computer storage system where the computer is able to access specific locations on a disk drive. Cinder is analogous to AWS's EBS.
- **Glance :** Glace is OpenStack's image service component that provides virtual templates (images) of hard disks. These templates can be used for new VMs. Glance may use either Swift or flat files to store these templates.
- **Neutron (formerly known as Quantum) :** This component of OpenStack provides Networking-as-a- Service, Load-Balancer-as-a-Service and Firewall- as-a-Service. It also ensures communication between other components.
- **Heat :** It is the orchestration component of OpenStack. It allows users to manage infrastructural needs of applications by allowing the storage of requirements in files.
- **Keystone :** This component provides identity management in OpenStack
- **Horizon :** This is a dashboard of OpenStack, which provides a graphical interface.
- **Ceilometer :** This component of OpenStack provisions meters and billing models for users of the cloud services. It also keeps an account of the resources used by each individual user of the OpenStack cloud. Let us also discuss some of the non- core components of OpenStack and their offerings.
- **Trove :** Trove is a component of OpenStack that provides Database-as-a- service. It provisions relational databases and big data engines.
- **Sahara :** This component provisions Hadoop to enable the management of data processors.
- **Zaqar :** This component allows messaging between distributed application components.
- **Ironic :** Ironic provisions bare-metals, which can be used as a substitute to VMs.

The basic architectural components of OpenStack, shown in Fig:4.12, includes its core and optional services/ components. The optional services of OpenStack are also known as Big Tent services, and OpenStack can be used without these components or they can be used as per requirement.



Components of open stack architecture

We have already discussed the core services and the four optional services. Let us now discuss the rest of the services.

- **Designate :** This component offers DNS services analogous to Amazon's Route 53. The following are the subsystems of Designate :
  Mini DNS Server
  Pool Manager
  Central Service and APIs
- **Barbican :** Barbican is the key management service of OpenStack that is comparable to KMS from AWS. This provides secure storage, retrieval, and provisioning and management of various types of secret data, such as keys, certificates, and even binary data.
- **AMQP :** AMQP stands for Advanced Message Queue Protocol and is a messaging mechanism used by OpenStack. The AQMP broker lies between two components of Nova and enables communication in a slackly coupled fashion.

Further, OpenStack uses two architectures - Conceptual and Logical, which are discussed in the next section.

Features and Benefits of OpenStack

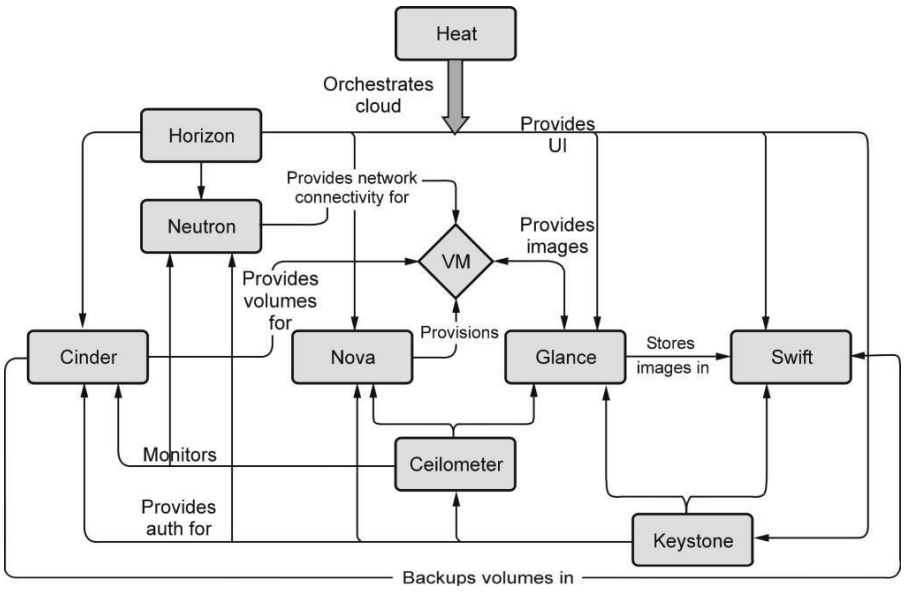
OpenStack helps build cloud environments by providing the ability to integrate various technologies of your choice. Apart from the fact that OpenStack is open-source, there are numerous benefits that make it stand out. Following are some of the features and benefits of OpenStack Cloud :

- **Compatibility :** OpenStack supports both private and public clouds and is very easy to deploy and manage. OpenStack APIs are supported in Amazon Web Services. The compatibility eliminates the need for rewriting applications for AWS, thus enabling easy portability between public and privateclouds.
- **Security :** OpenStack addresses the security concerns, which are the top- most concerns for most organisations, by providing robust and reliable security systems.
- **Real-time Visibility :** OpenStack provides real-time client visibility to administrators, including visibility of resources and instances, thus enabling administrators and providers to track what clients are requesting for.
- **Live Upgrades :** This feature allows upgrading services without any downtime. Earlier, for upgradations, the was a need for shutting-down complete systems, which resulted in loss of performance. Now, OpenStack has enabled upgrading systems while they are running by requiring only individual components to shut- down.

Apart from these, OpenStack offers other remarkable features, such as networking, compute, Identity Access Management, orchestration, etc.

Conceptual OpenStack Architecture

Fig. 4.13, depicting a magnified version of the architecture by showing relationships among different services and between the services and VMs. This expanded representation is also known as the Conceptual architecture of OpenStack.



Conceptual architecture of OpenStack

we can see that every service of OpenStack depends on other services within the systems, and all these services exist in a single ecosystem working together to produce a virtual machine. Any service can be turned on or off depending on the VM required to be produced. These services communicate with each other through APIs and in some cases through privileged admin commands.

Let us now discuss the relationship between various components or services specified in the conceptual architecture of OpenStack. As you can see in Figure 4.2, three components, Keystone, Ceilometer and Horizon, are shown on top of the OpenStack platform.

Here, **Horizon** is providing user interface to the users or administrators to interact with underlying OpenStack components or services, **Keystone** is providing authentication to the user by mapping the central directory of users to the accessible OpenStack services, and **Ceilometer** is monitoring the OpenStack cloud for the purpose of scalability, billing, benchmarking, usage reporting and other telemetry services. Inside the OpenStack platform, you can see that various processes are handled by different OpenStack services; **Glance** is registering Hadoop images, providing image services to OpenStack and allowing retrieval and storage of disk images. Glance stores the images in **Swift**, which is responsible for providing reading service and storing data in the form of objects and files. All other OpenStack components also store data in Swift, which also stores data or job binaries. **Cinder**, which offers permanent block storage or

volumes to VMs, also stores backup volumes in Swift. **Trove** stores backup databases in Swift and boots databases instances via **Nova**, which is the main computing engine that provides and manages virtual machines using disk images.

Neutron enables network connectivity for VMs and facilitates PXE Network for Ironic that fetches images via Glance. **VMs** are used by the users or administrators to avail and provide the benefits of cloud services. All the OpenStack services are used by VMs in order to provide best services to the users. The infrastructure required for running cloud services is managed by **Heat**, which is the orchestration component of OpenStack that orchestrates clusters and stores the necessarys resource requirements of a cloud application. Here, **Sahara** is used to offer a simple means of providing a data processing framework to the cloud users.

Table 4.14 shows the dependencies of these services.

Code Name	Dependent on	Optional
Nova (Compute)	Keystone, Horizon, Glance	Cinder, Neutron
Swift (Object Storage)	Keystone	-
Cinder (Block Storage)	Keystone	-
Glance (Image Service)	Swift, Keystone, Horizon	-
Neutron (Network)	Keystone, Nova	-
Keystone (Identity)	-	-
Horizon (Dashboard)	Keystone	-

Service Dependencies

Modes of Operations of OpenStack

OpenStack majorly operates in two modes - single host and multi host. A single host mode of operation is that in which the network services are based on a central server, whereas a multi host operation mode is that in which each compute node has a duplicate copy of the network running on it and the nodes act like Internet gateways that are running on individual nodes. In addition to this, in a multi host operation mode, the compute nodes also individually host floating IPs and security groups. On the other hand, in a single host mode of operation, floating

IPs and security groups are hosted on the cloud controller to enable communication.

Both single host and multi host modes of operations are widely used and have their own set of advantages and limitations.

A single host mode of operation has a major limitation that if the cloud controller goes down, it results in the failure of the entire system because instances stop communicating.

This is overcome by a multi host operation mode where a copy of the network is provisioned to every node. Whereas, this limitation is overcome by the multi host mode, which requires a unique public IP address for each compute node to enable communication.

In case public IP addresses are not available, using the multi host mode is not possible.