

AN INTRODUCTION TO

OpenNebula



Agenda

- Overview
- History
- Architecture overview
- Components
- Architecture
- Comparison
- Conclusion

The logo for 'one' consists of the word 'one' in a lowercase, sans-serif font. The letter 'o' is black, while the letters 'n' and 'e' are a vibrant blue.

AN INTRODUCTION TO

OpenNebula

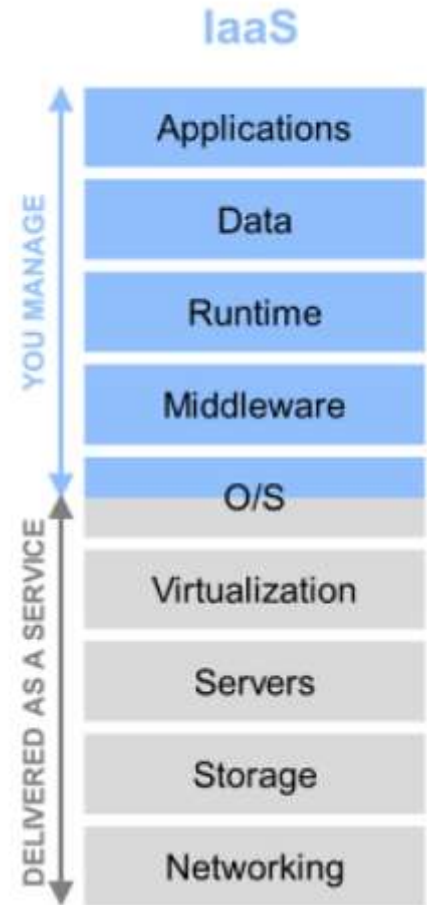
By Saeed Amiri 23 May 2017 Slide 2 of 26

Overview

OpenNebula is an open-source Cloud management platform (CMP) to build **IaaS**

- Private clouds
- public clouds
- hybrid clouds

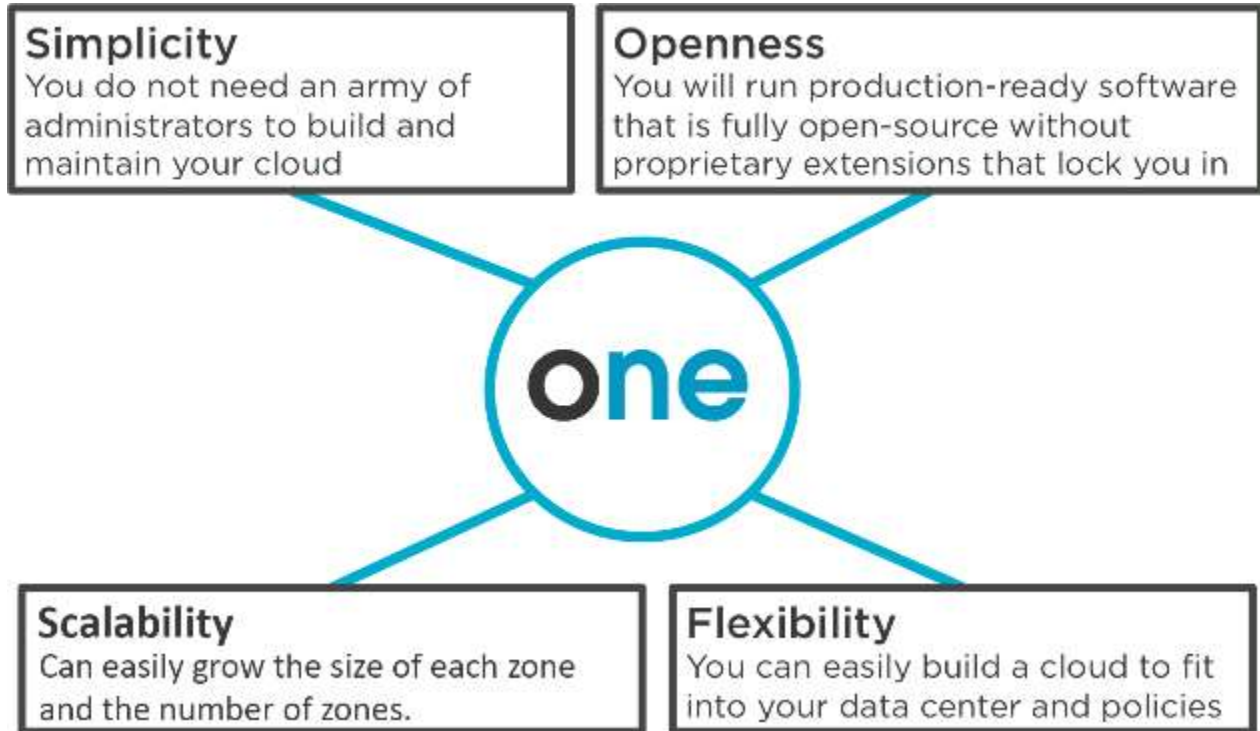
Fully open-source software released under Apache license



OpenNebula main features

- ✓ **Cloud Bursting.** Extension of the local private infrastructure with resources from remote clouds.
- ✓ **On-demand Provision of Virtual Data Centers.** A Virtual Data Centers (VDC) is a fully-isolated virtual infrastructure environment where a group of users, under the control of the VDC administrator, can create and manage compute, storage and networking capacity.
- ✓ **Multiple Zones.** Centralized management of multiple instances of OpenNebula (zones) for scalability, isolation or multiple-site support.
- ✓ **Multi-VM Application Management.** Automatic execution of multi-tiered applications with auto-scaling.

Commitment

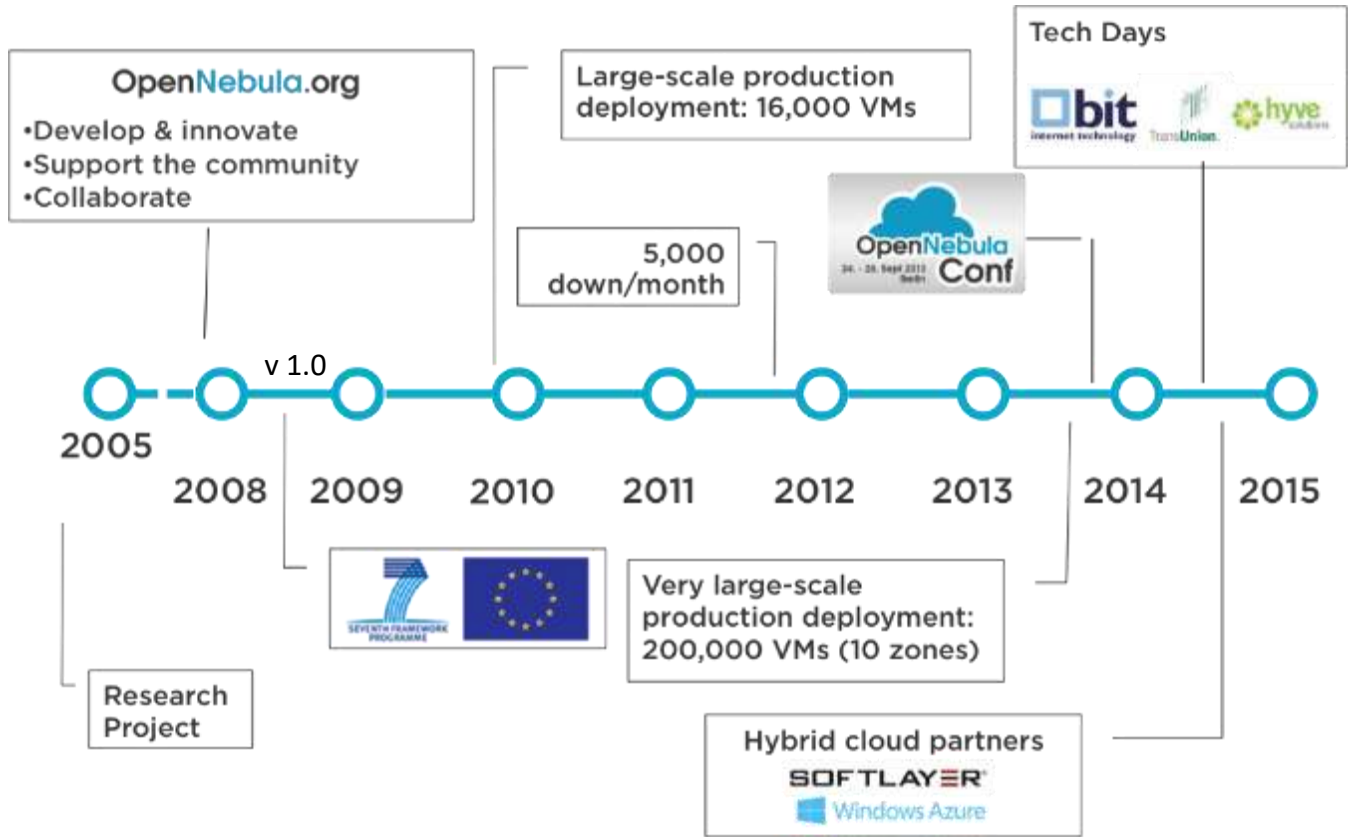


Sponsors

Organizations who have funded the project.



History



History

JUNE 19-20, 2017

FIFTH OPENNEBULA CONFERENCE IN BOSTON

OpenNebula Community Conferences

OpenNebula Conferences are education events that serve as a meeting point of cloud users, developers, administrators, integrators and researchers, featuring talks with experiences and use cases.

Previous speakers include:

Santander Bank, CentOS, European Space Agency, Akamai, FermiLab, Puppet, Red Hat, BlackBerry, Telefonica, Akamai, Runtastic, Unity, Citrix...

private, public and hybrid clouds

Public

- **public** over the Internet
- Simple web interface
- Raw *Infrastructure* Resources
- Pay-as-you-go (On-demand)
- Elastic & “infinite” Capacity

Private

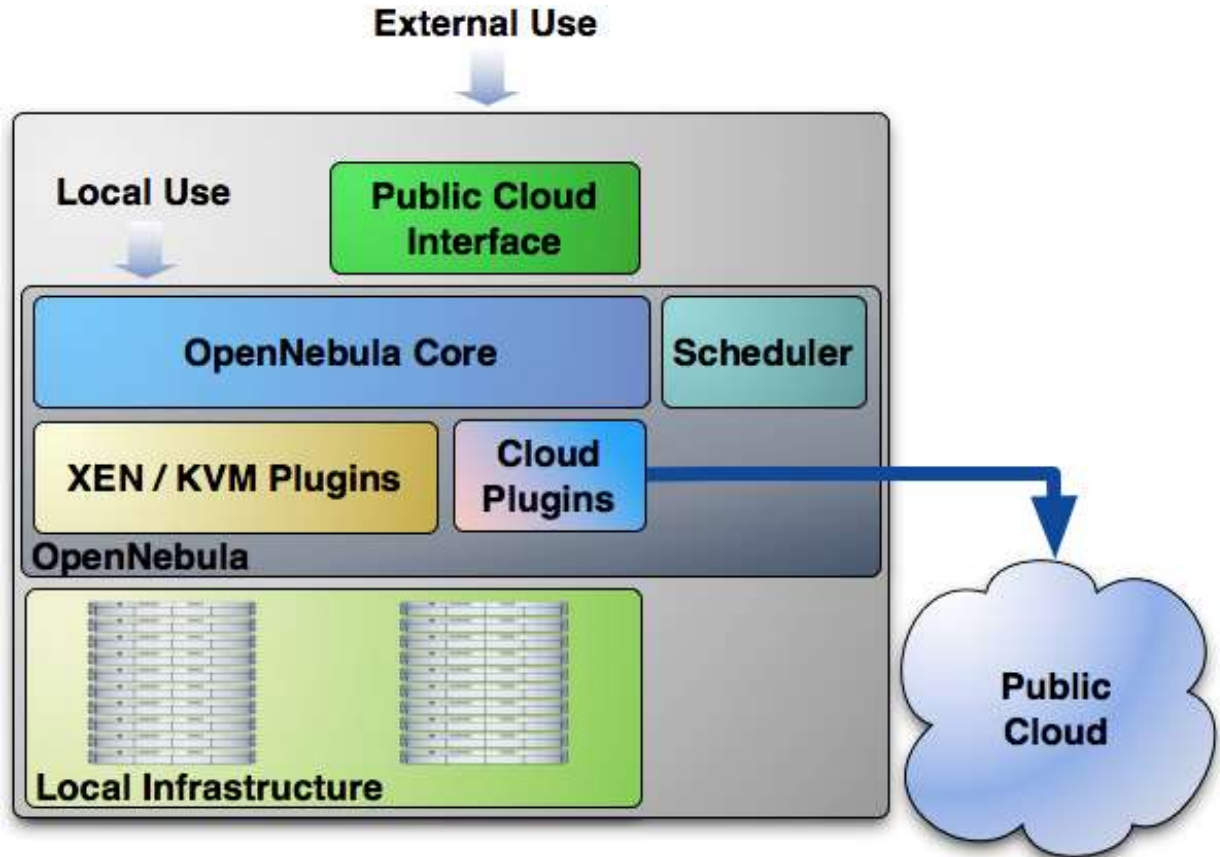
“Public Cloud behind firewall”

- used in the datacenter or cluster.
- Simplify internal operations
- Higher utilization & operational savings
- Security concerns.

Hybrid

Supplement the capacity
of the Private Cloud
maintained by both internal
and external providers

private, public and hybrid clouds



Architecture overview

OpenNebula follows a bottom-up approach.

A cloud architecture is defined by three components:

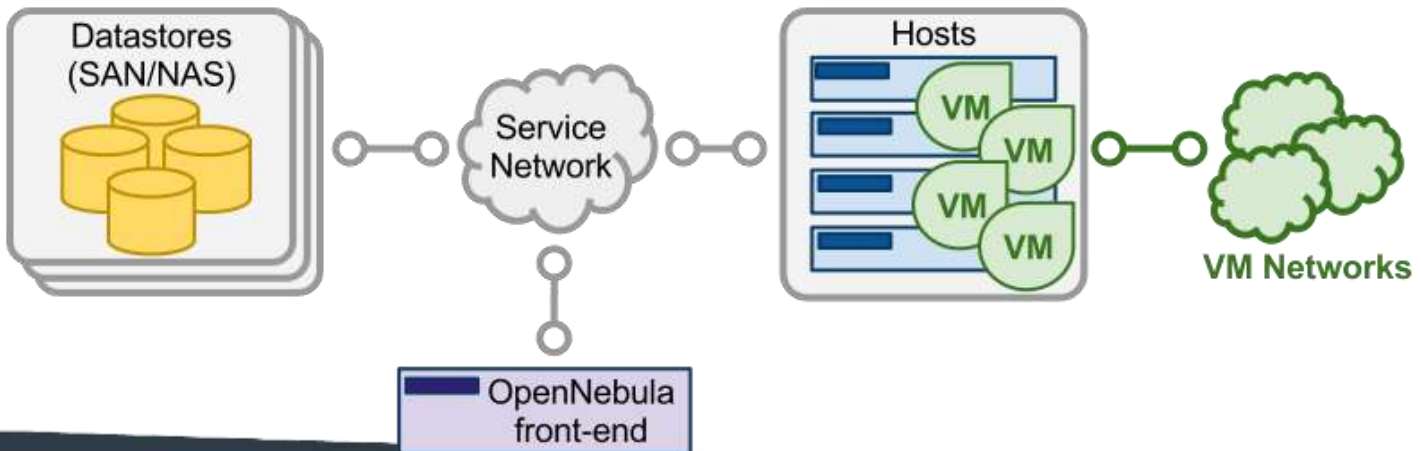
- storage
- Networking
- virtualization

Therefore, the basic components of an OpenNebula system are:

- Front-end
- hosts
- Datastores
- Networks

Architecture overview

- **Front-end** that executes the OpenNebula services.
- Hypervisor-enabled **hosts** that provide the resources needed by the VMs.
- **Datstores** that hold the base images of the VMs.
- Physical **networks** used to support basic services such as interconnection of the storage servers and OpenNebula control operations, and VLANs for the VMs.



Front-End

The machine that holds the OpenNebula installation is called the front-end.

This machine needs network connectivity to all the hosts, and possibly access to the storage Datastores

takes less than 150MB.

OpenNebula services include:

- Management daemon (`oned`) and scheduler (`mm_sched`)
- Web interface server (`sunstone-server`)
- Advanced components: OneFlow, OneGate, econe, ..

Front-End

There are several certified platforms to act as front-end

Component	Version
RedHat Enterprise Linux	7.0
Ubuntu Server	14.04 (LTS), 16.04 (LTS)
CentOS	7.0
Debian	8
Ruby Gems	Versions installed by packages and install_gems utilit

OpenNebula's default database uses **SQLite**.
And in large scale using MySQL.

Monitoring

The monitoring subsystem gathers information relative to the hosts and the virtual machines, such as the host status, basic performance indicators, as well as VM status and capacity consumption.

This information is collected by executing a set of static probes provided by OpenNebula.

The information is sent according to the following process: (push mechanism) each host periodically sends monitoring data to the front-end which collects it and processes it in a dedicated module.

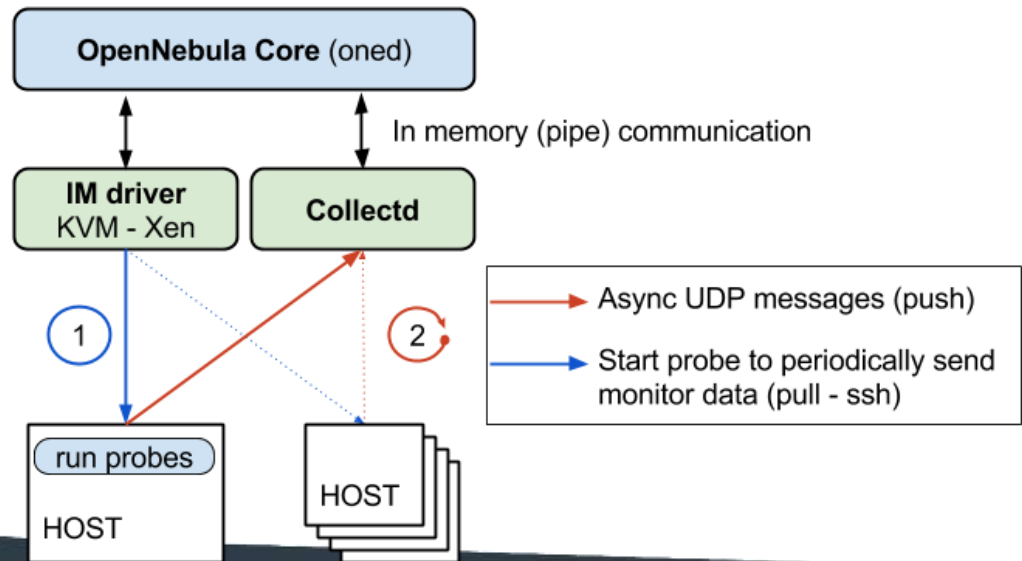
This model is highly scalable and **its limit** (in terms of number of VMs monitored per second) is bounded to the performance of the server running *oned* and the database server.

Monitoring

This distributed monitoring use:

push model is a style of communication where the request for a given transaction is initiated by the publisher or central server.

A **lightweight protocol** is one of communication protocols that are characterized by a relatively small overhead in transmitted on top of the functional data



Virtualization Hosts

The hosts are the physical machines that will run the VMs.
also called OpenNebula nodes.

The Virtualization Subsystem is the component in charge of talking with the hypervisor installed in the hosts and taking the actions needed for each step in the VM life-cycle.

OpenNebula natively supports the [KVM](#) hypervisor

OpenNebula is configured by default to interact with hosts running KVM.

Storage

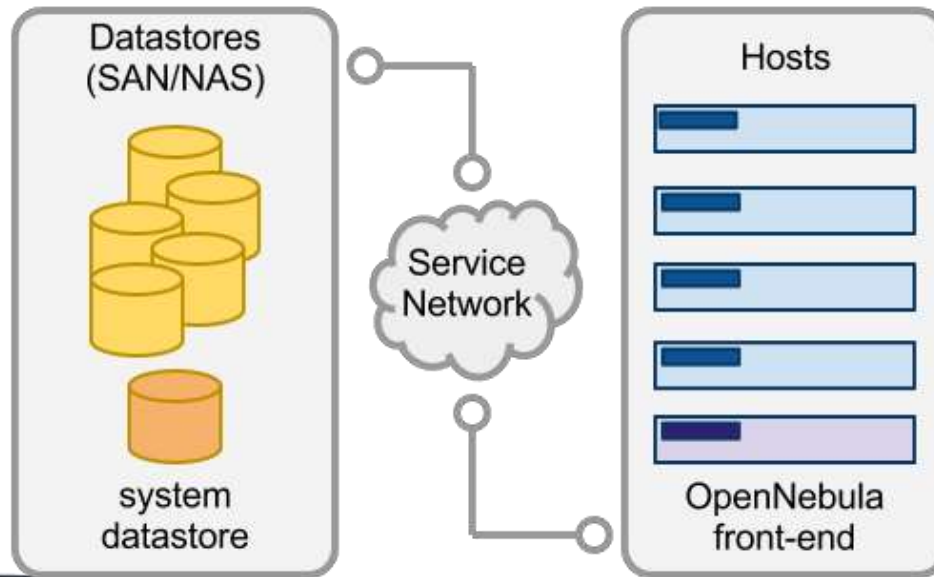
OpenNebula is shipped with 3 different datastore classes:

- **System Datastores:** to hold images for running VMs.
- **Image Datastores:** to store the disk images repository. Disk images are moved, or cloned to/from the System Datastore when the VMs are deployed or shutdown.
- **File Datastore:** a special datastore used to store plain files, not disk images. plain files can be used as kernels, ram-disks or context files.

Each datastore has to be accessible through the front-end using any suitable technology *NAS, SAN or direct attached storage*. (In Next Slide)

Storage

- **Storage-Area Network (SAN)** is a specialized, high-speed network that provides block-level and presents shared pools of storage.
- **Network-attached storage (NAS)** is a file-level computer data storage server that connected to a network and providing data access.



Networking

At least two different physical networks are needed:

Service Network:

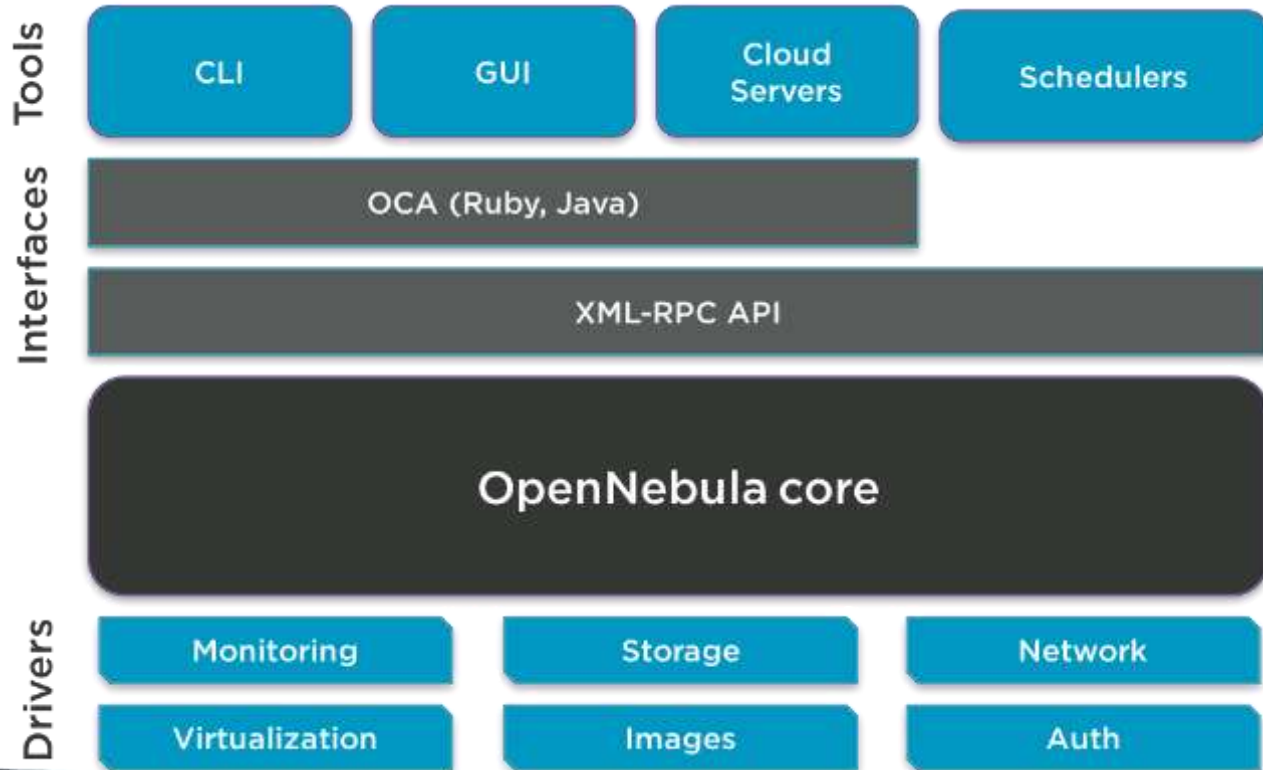
used by the OpenNebula front-end daemons to access the hosts in order to manage and monitor the hypervisors, and move image files. It is highly recommended to install a dedicated network for this purpose.

Instance Network:

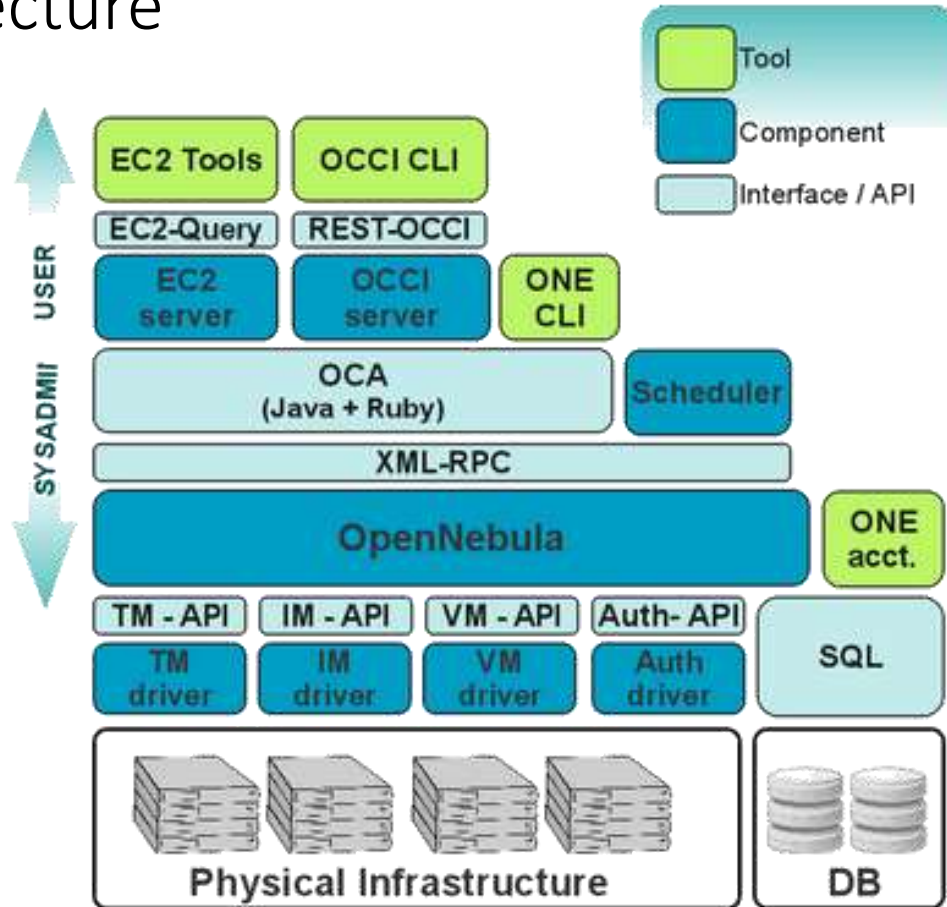
offers network connectivity to the VMs across the different hosts. To make an effective use of your VM deployments, you will probably need to make one or more physical networks accessible to them.

Architecture

Integrate with other Components



Architecture



Comparison

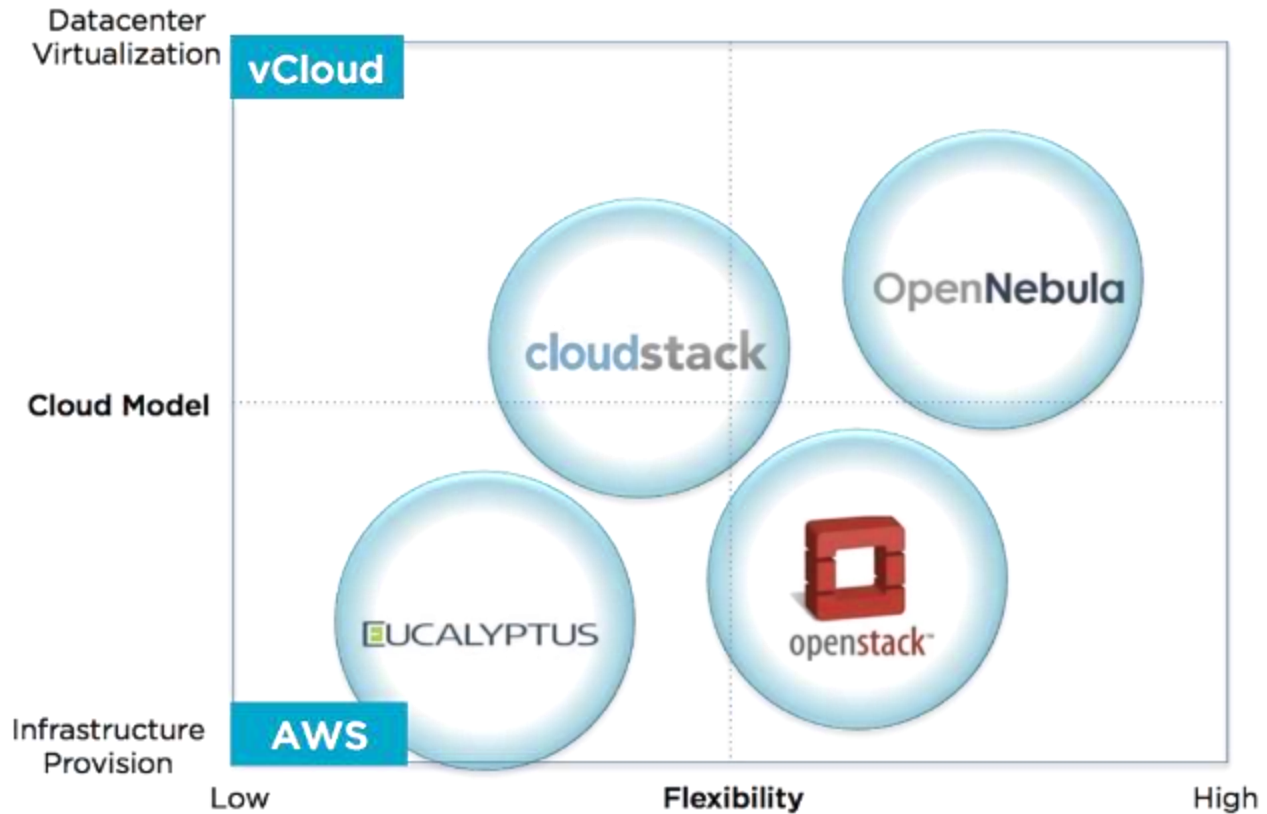
Two Different Cloud Models

- **Datacenter Virtualization:** On one side, there are businesses that understand cloud as an extension of virtualization in the datacenter; hence looking for a vCloud-like infrastructure automation tool to orchestrate and simplify the management of the virtualized resources.
- **Infrastructure Provision:** On the other side, there are businesses that understand cloud as an AWS-like cloud on-premise; hence looking for a provisioning tool to supply virtualized resources on-demand.

Comparison

	DATACENTER VIRTUALIZATION	INFRASTRUCTURE PROVISION
Interfaces	Feature-rich API and administration portal	Simple cloud APIs and self-service portal
Management Capabilities	Complete life-cycle management of virtual and physical resources	Simplified life-cycle management of virtual resources with abstraction of underlying infrastructure
Cloud Deployment	Mostly private	Mostly public
Internal Design	Bottom-up design dictated by the management of datacenter complexity	Top-down design dictated by the efficient implementation of cloud interfaces
Datacenter Integration	Easy to adapt to fit into any existing infrastructure environment to leverage IT investments	Built on new, homogeneous commodity infrastructure

Comparison



Conclusion

Example of using OpenNebula with other cloud platforms:

- ❖ Some corporations are mixing an Enterprise Cloud with an in-house Cloud Service. They are implementing a cloudbursting architecture where an OpenNebula enterprise cloud bursts to an OpenStack- or Eucalyptus-based cloud when the demand for computing capacity spikes.
- ❖ Other corporations are using components from different projects to build their cloud. The integration capabilities of OpenNebula are allowing its integration with *OpenStack Swift* or *OpenStack Quantum* for object/block store and networking management respectively in the data center.

Tanks For Attention

