OpenStack

Introduction to OpenStack

Kh. Rashedul Arefin

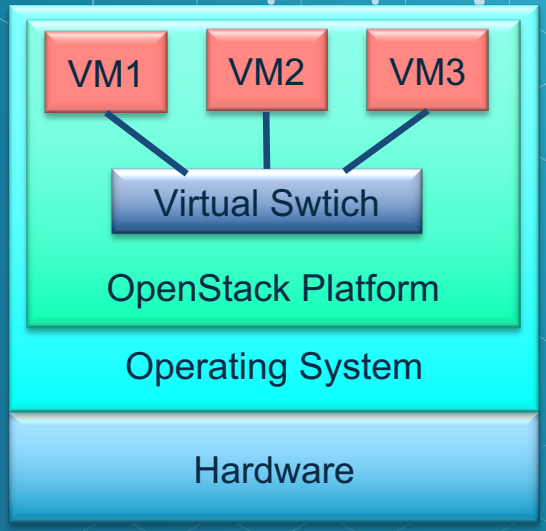
Agenda

- Overview of OpenStack
- Components of OpenStack
- Operation Flow
- Architecture of OpenStack
- Deployment Model
- Virtual Data Center

Overview of OpenStack

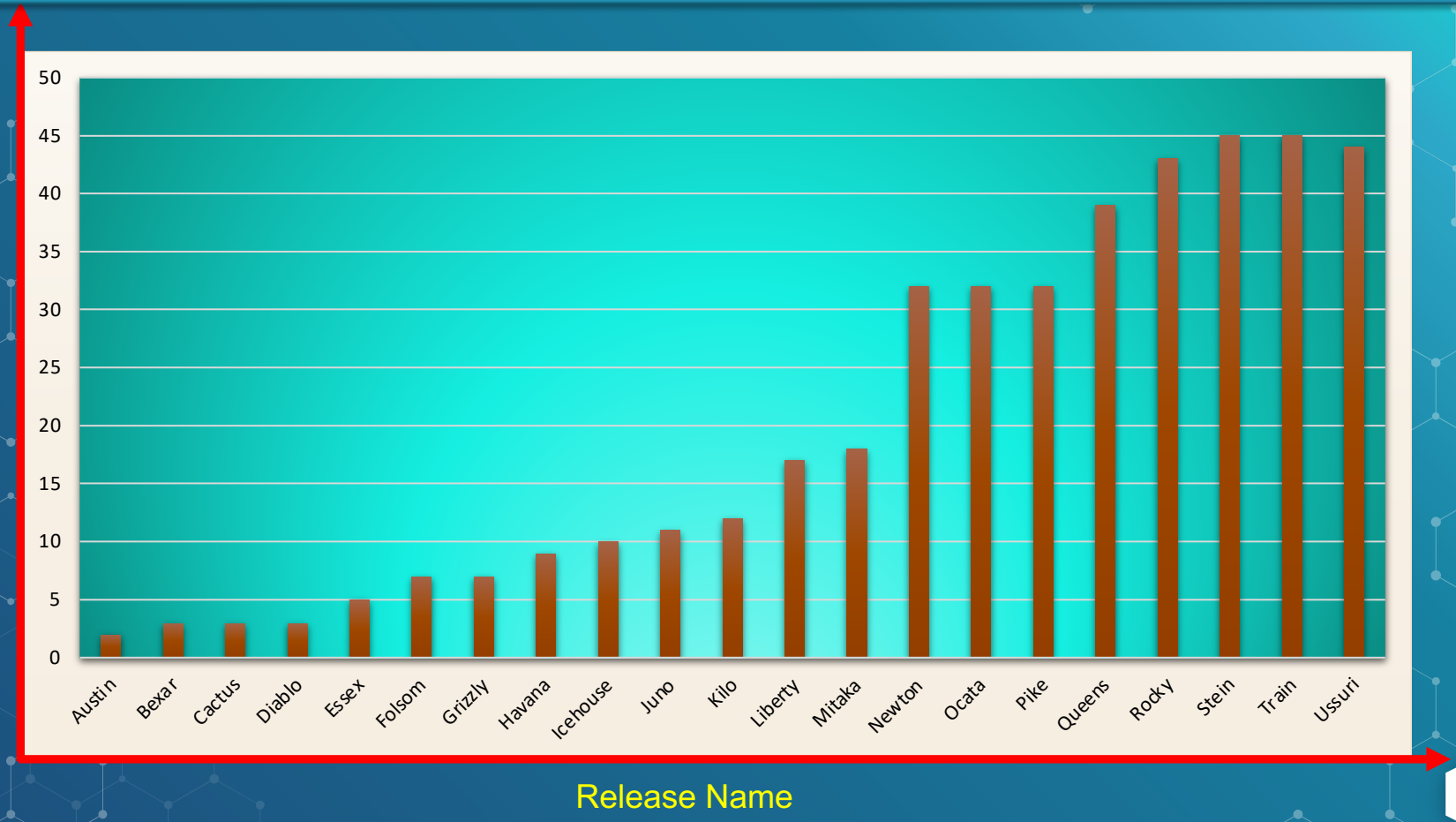
OpenStack is made up of a number of largely independent but related projects, each of which provides the abstraction layer for a particular area of infrastructure

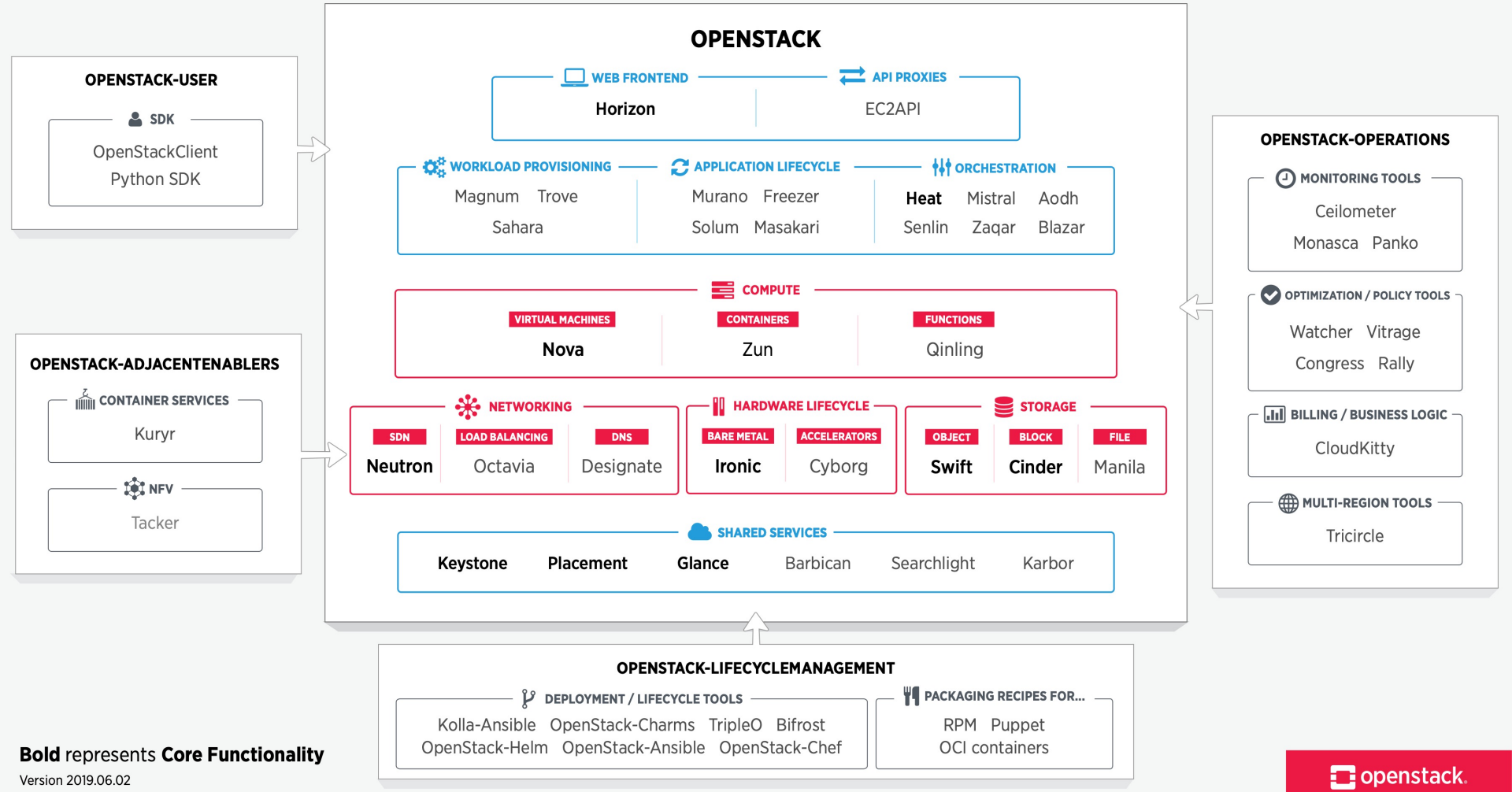
- Founded by NASA and Rackspace in 2010
- Provides IaaS
- Mainly Written in Python
- Command-Line Tools
- RESTful Web Service
- Open Source
- Modular https://www.openstack.org/software/sample-configs#web-applications
- Managed by OpenStack Foundation



OpenStack is treated as an Environment rather than just a Product

Development History of OpenStack





Bold represents **Core Functionality**

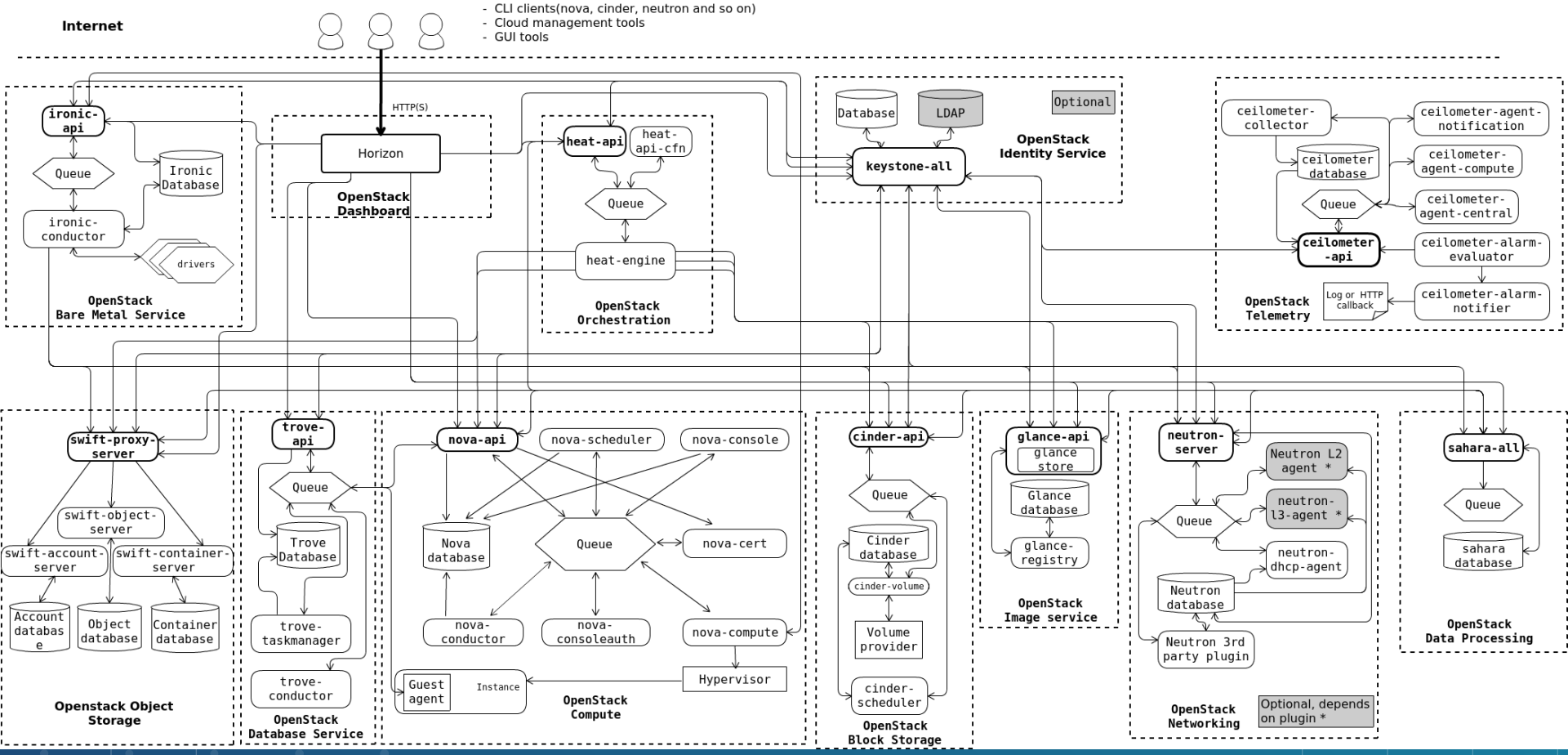
Version 2019.06.02

openstack.

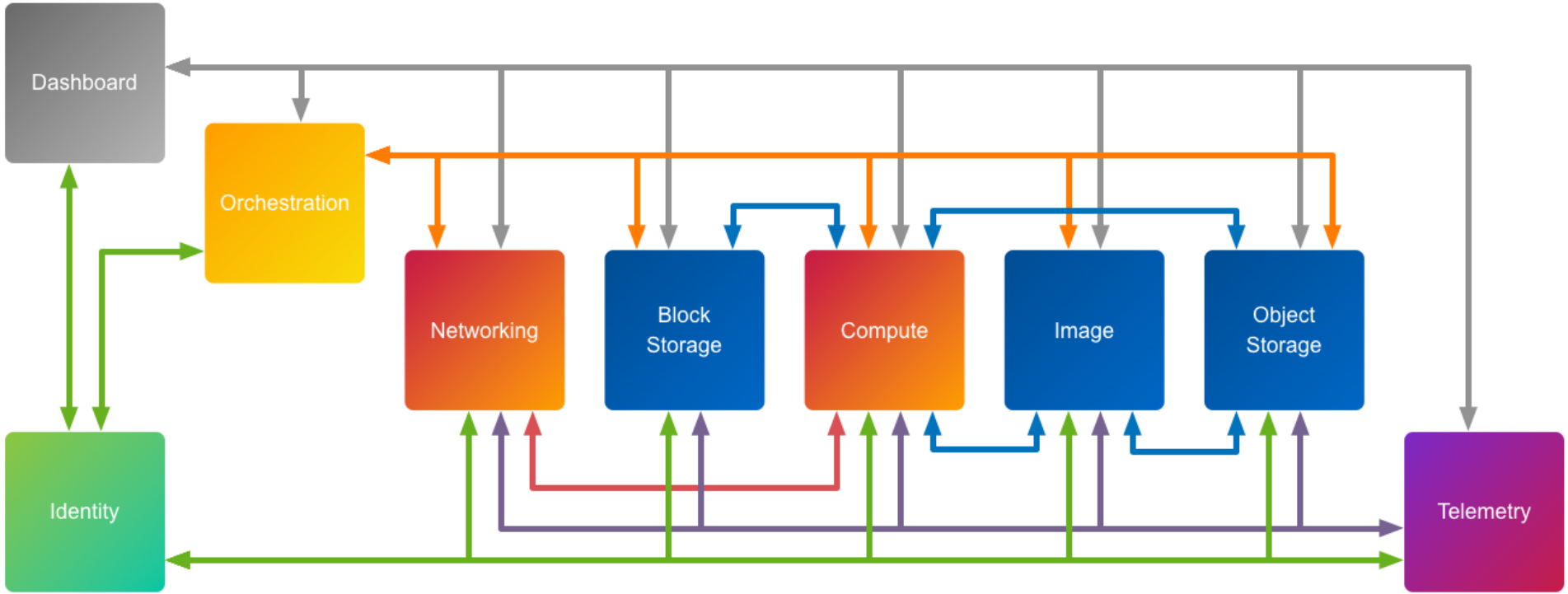
Major Components' Dependency

Component	Functional Area	Dependancy
Keystone	Identity Service	None
Nova	Compute Service	Keystone, Horizon, Glance, Cinder (Optional), Neutron (Optional)
Ironic	Bare Metal Service	Nova, Keystone, Horizon, Glance, Cinder (Optional), Neutron (Optional)
Neutron	Networking Service	Keystone, Nova
Glance	Image Service	Swift, Keystone, Horizon
Cinder	Block Storage Service	Keystone
Swift	Object Storage Service	Keystone
Manila	File Sharing Service	Keystone
Horizon	Dashboard Service	Keystone
Sahara	Data Processing Service	Keystone, Nova, Glance, Swift,
Trove	Database Service	Keystone, Nova, Glance
Ceilometer	Telemetry Service	Keystone
Heat	Orchestration Service	Keystone

OpenStack Logical Architecture



Conceptual Diagram

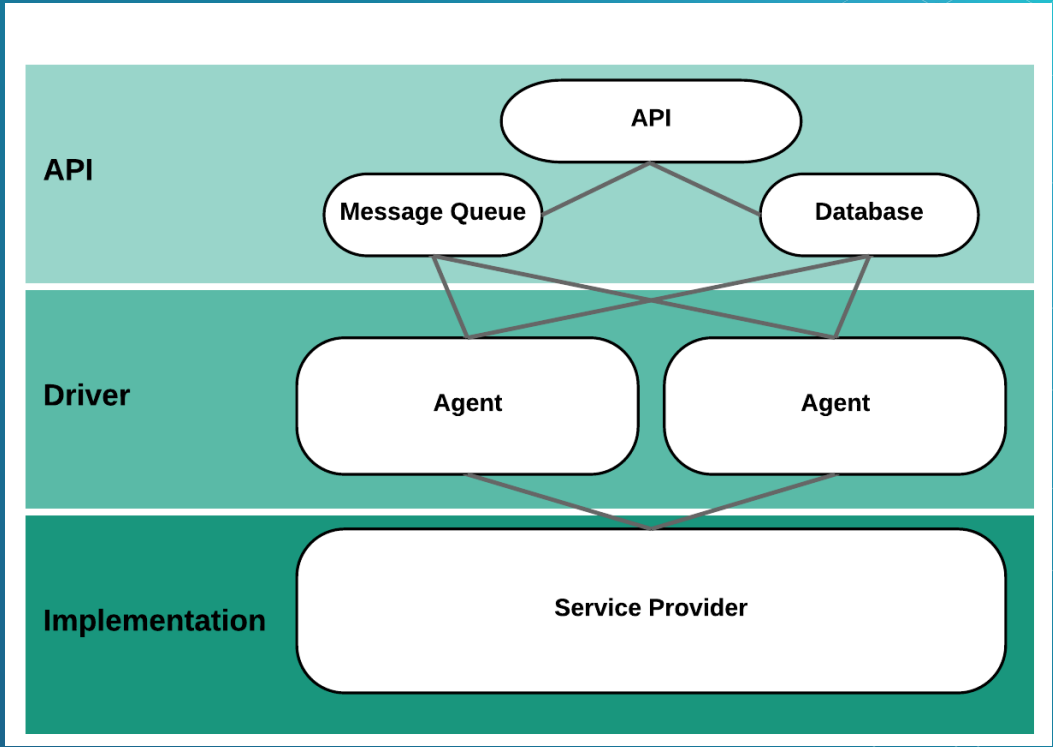


Individual Component Architecture

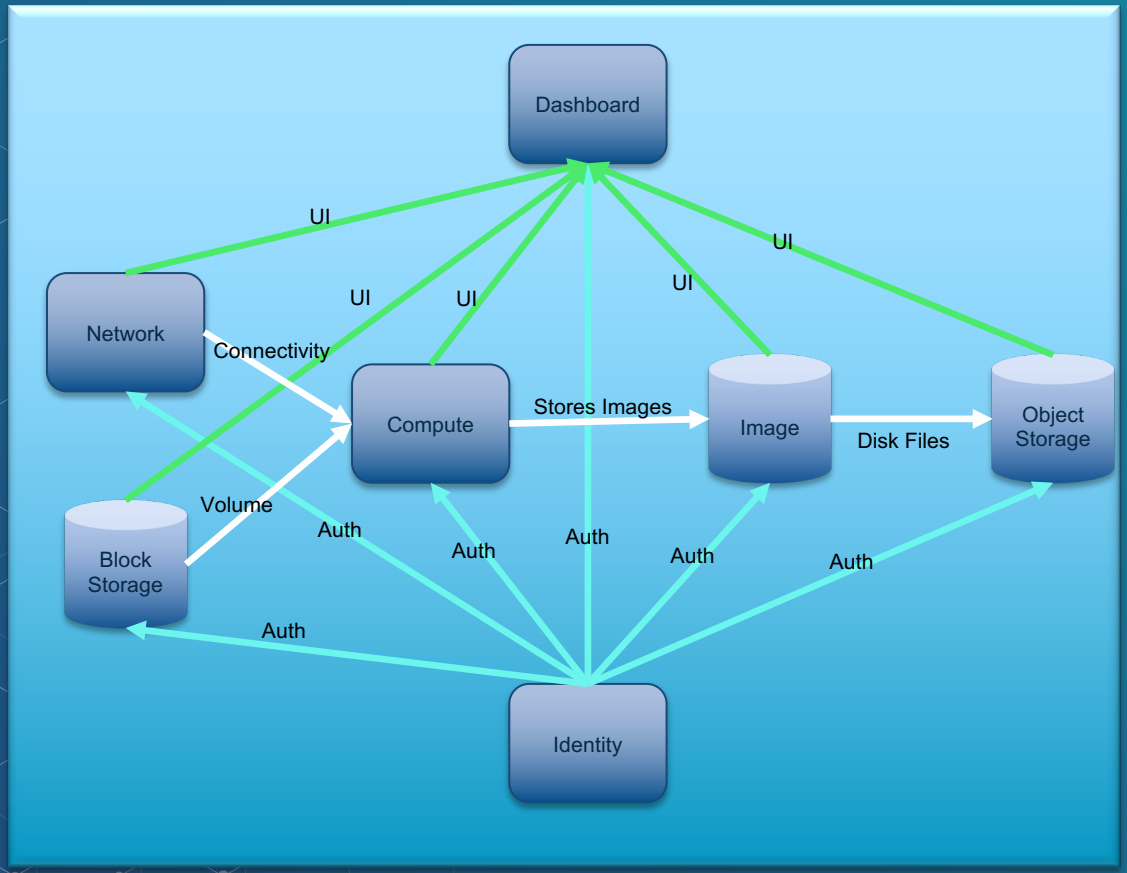
API Layer: A Software application that provides API describing a way of interacting and expects respective response

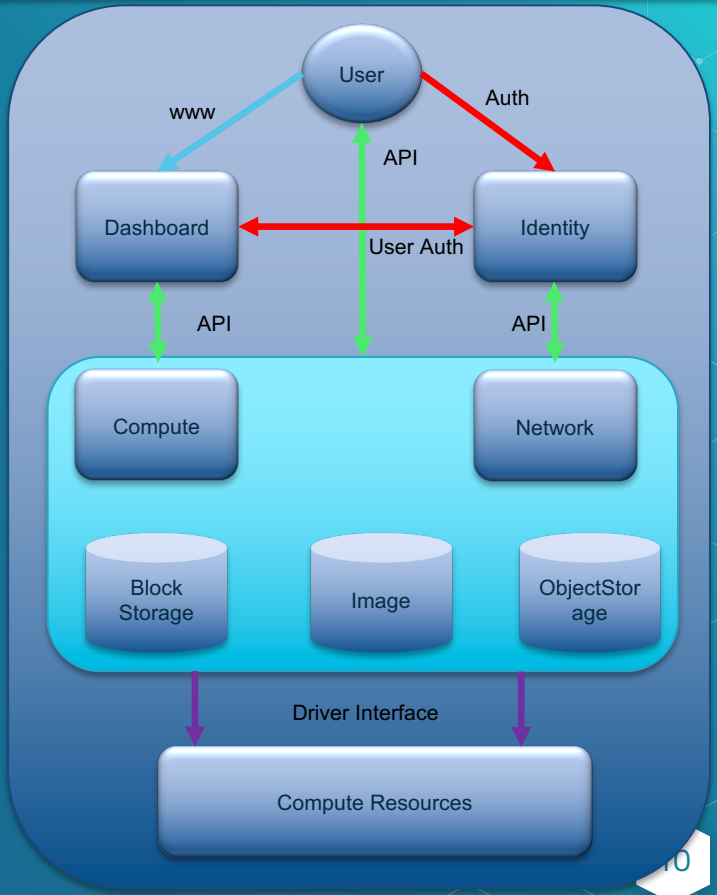
Driver Layer: A translator that takes API instruction set and converts it into appropriate commands for the base infrastructure

Implementation Layer: Actual infrastructure that provides the resources for consumption for the Applications.



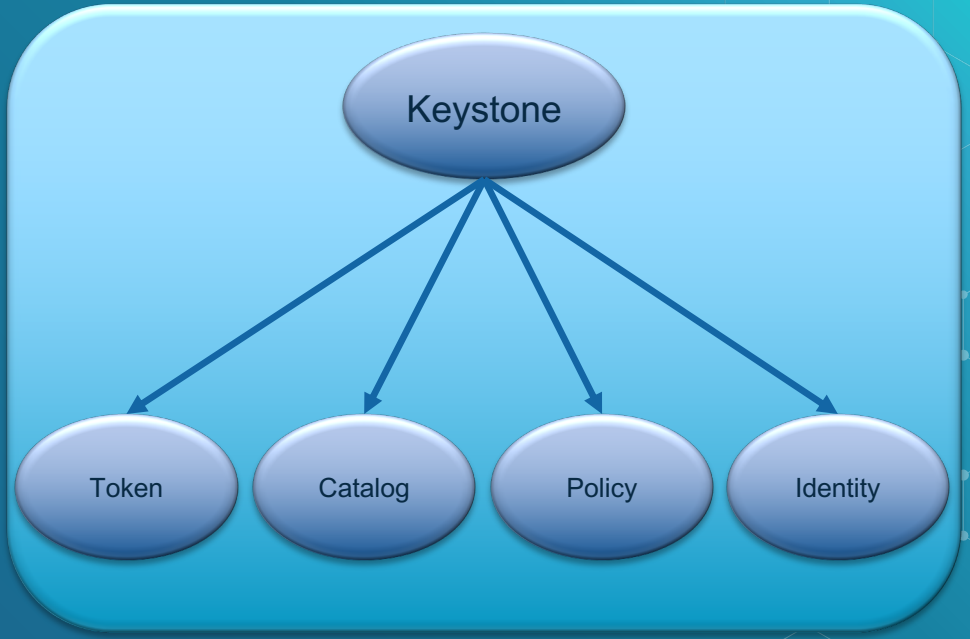
Operation Flow of OpenStack





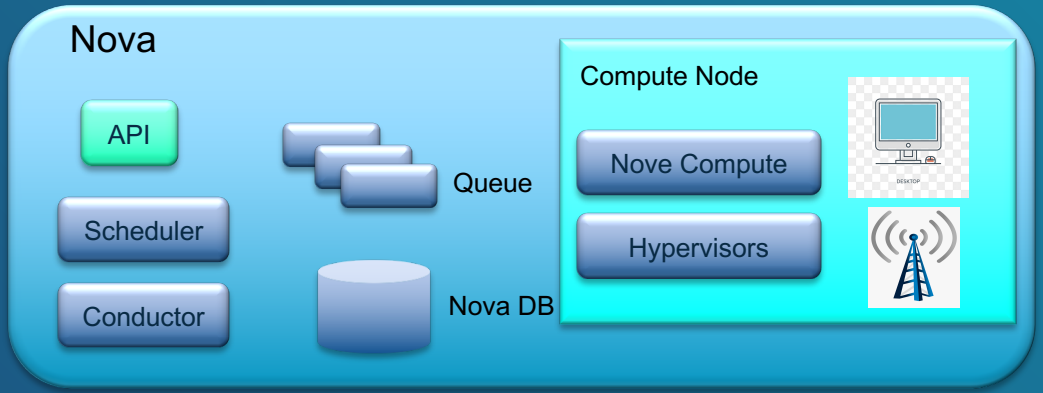
Keystone - Identity Service

- Basically contains a central list of all users and mapped to the accessible services
- Openstack Identity Servcie
- Authentication and Authorization
- Two Functions: User Management and Service Catalog
- Can be integrated with existing backend directory services like LDAP

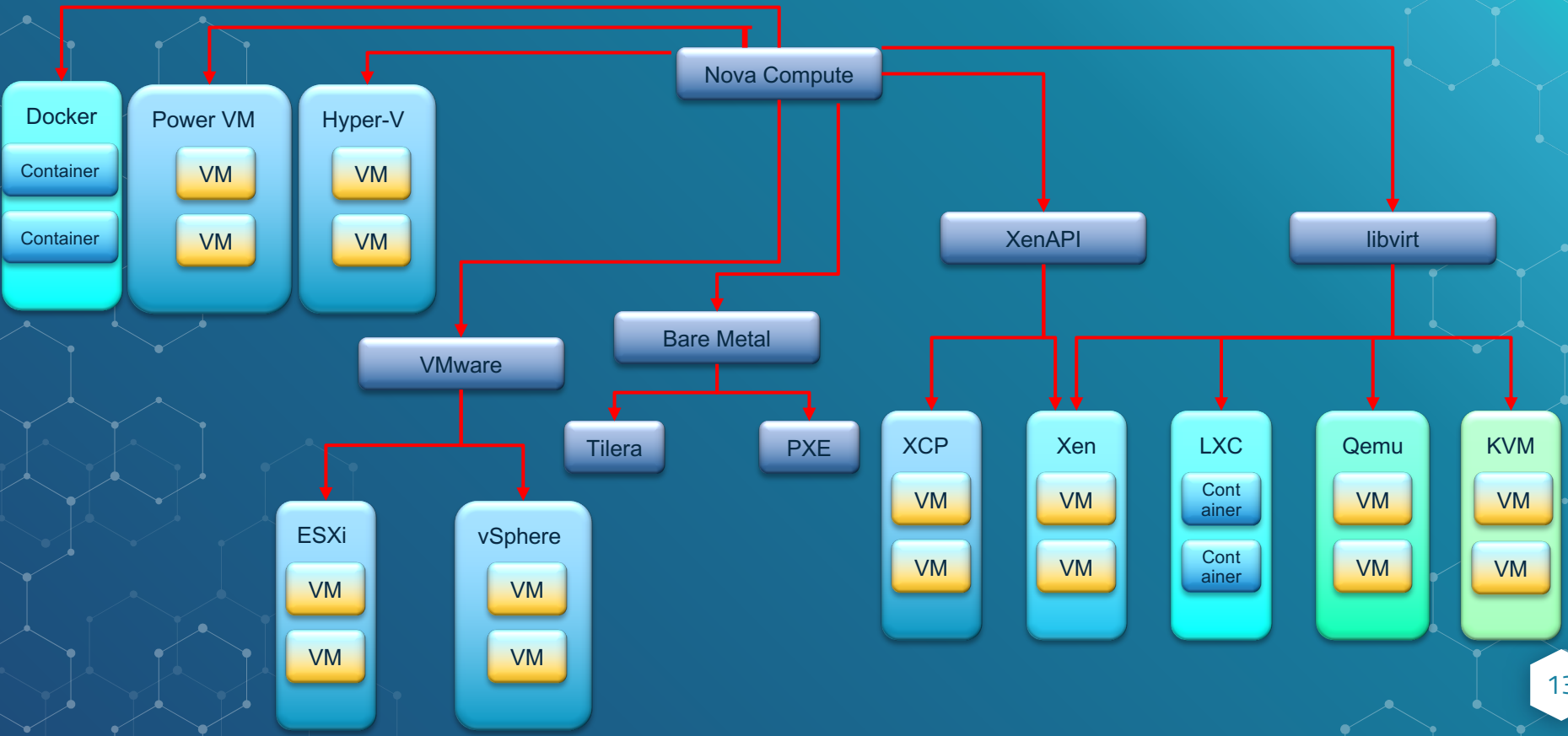


Nova – Compute Service

- Most important and mandatory computing component
- Cloud computing fabric controller
- Provide on-demand access to compute resources
- Manage and automate pools of computing resources
- Provision and Manage large network for VMs
- Works with VMware, Xen, KVM, Hyper-V, Docker and LXC

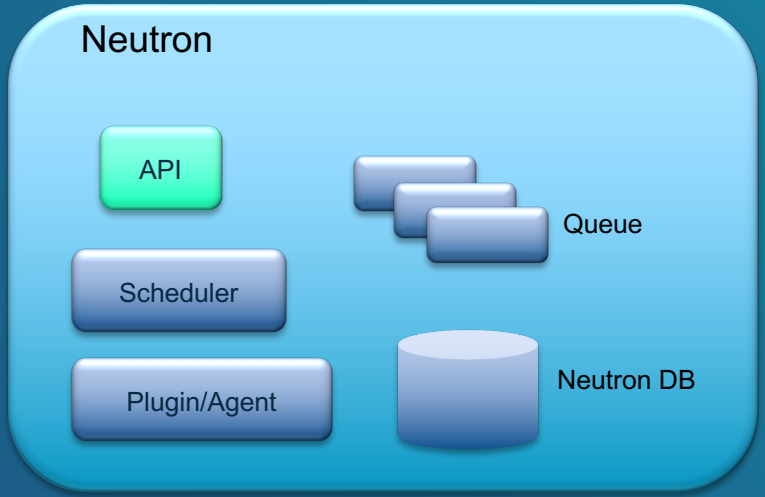


OpenStack Compute Service

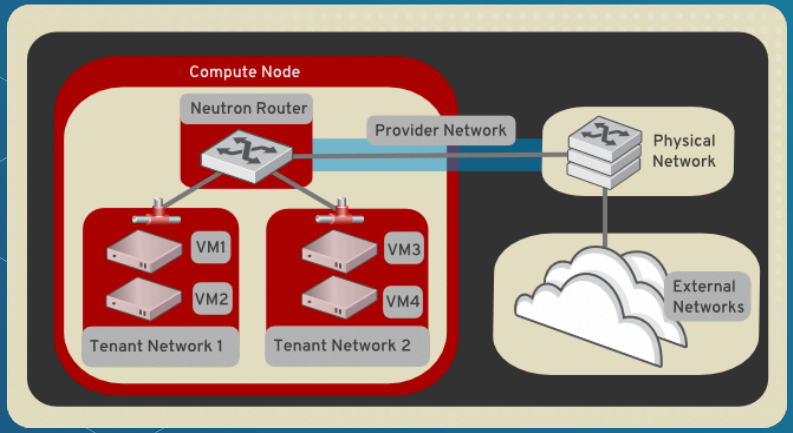


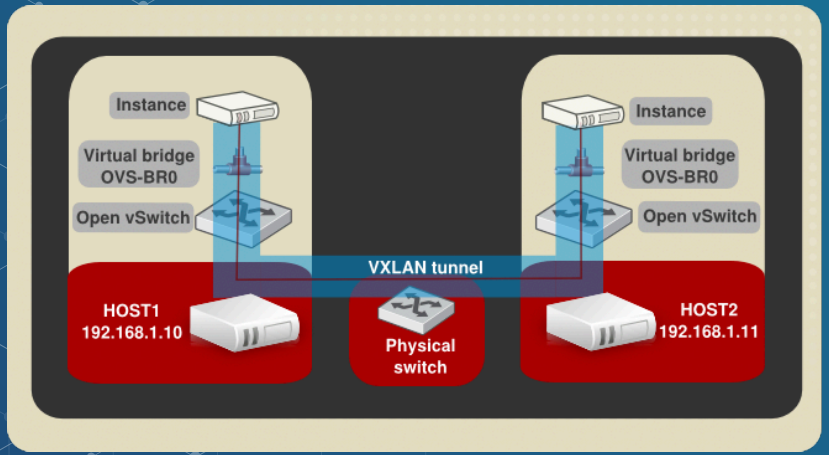
Neutron – Networking Service

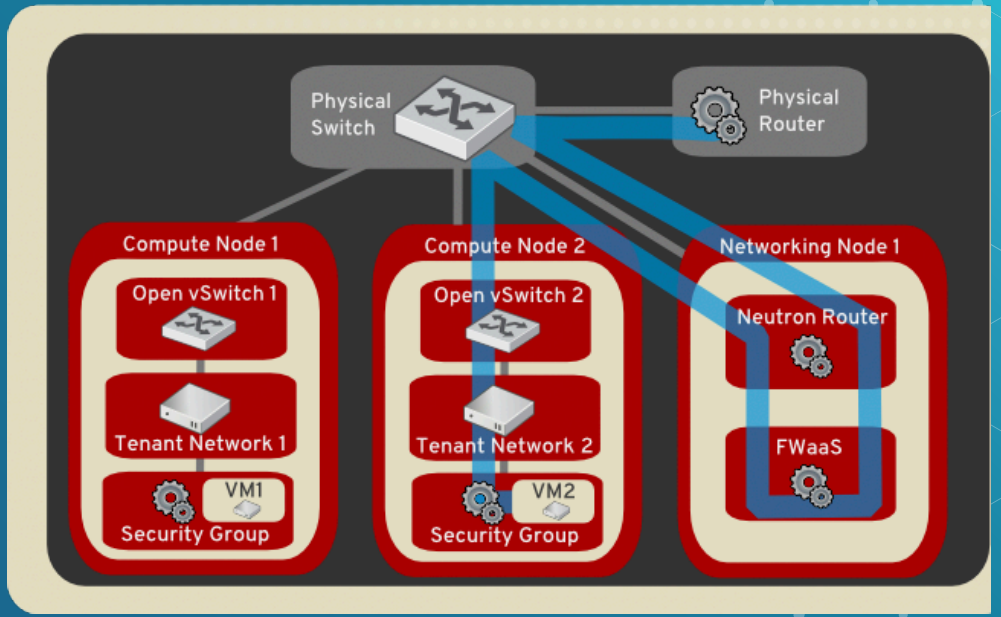
- Responsible for neuron structure between components for better connectivity
- Accepts API request and routes them to appropriate plugins
- Provide Network as a Service (NaaS)
- Ships with plugins and agents for Cisco virtual and physical switches, Linux bridging and Midokua
- Common agents are L3, DHCP and specific plugin agent
- Advanced Networking services like SDN, LBaaS, FWaaS, VPN, IDS etc.



Neutron – Networking Service

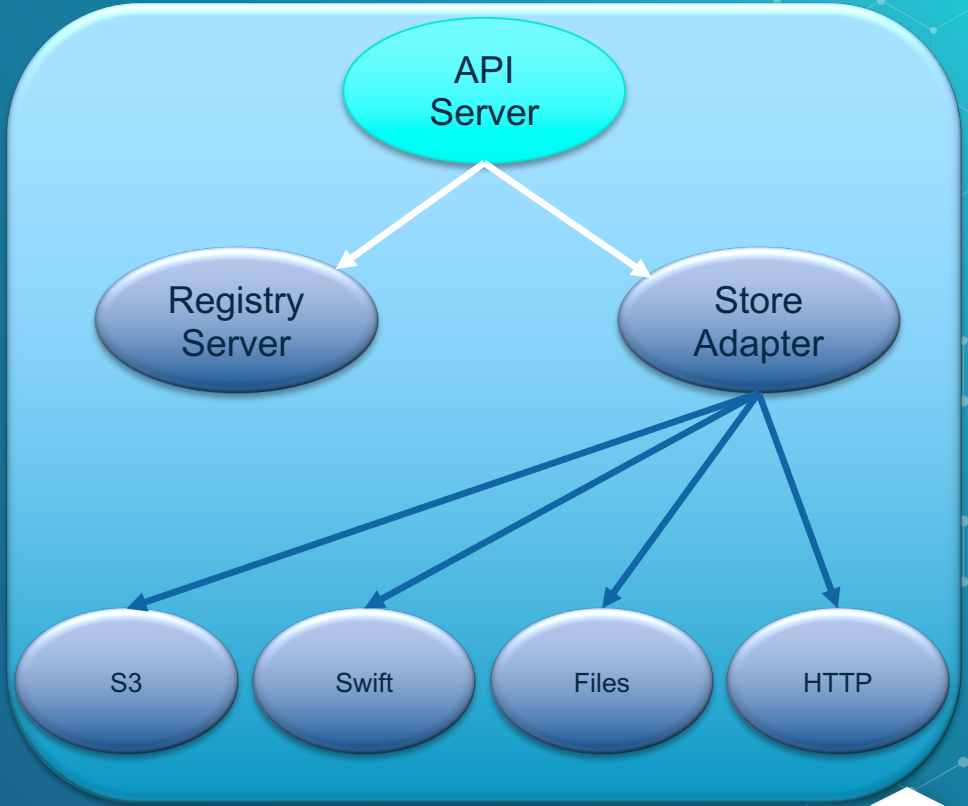






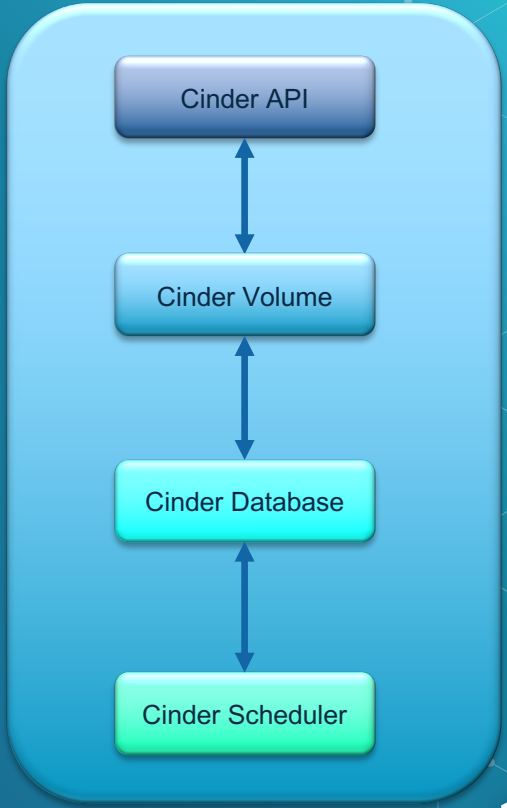
Glance – Image Service

- Users can upload and discover data assets
- Contents Images and Metadata Definitions
- Service includes discovering, registering and retrieving virtual machine images
- Stored images can be used as template
- Can store and caalog an unlimited number of backups
- Can store disk and server images in a variety of backends including Swift
- If integrated with Vmware, Glance introduces advanced features to the vSphere family – vMotion, DRS etc.



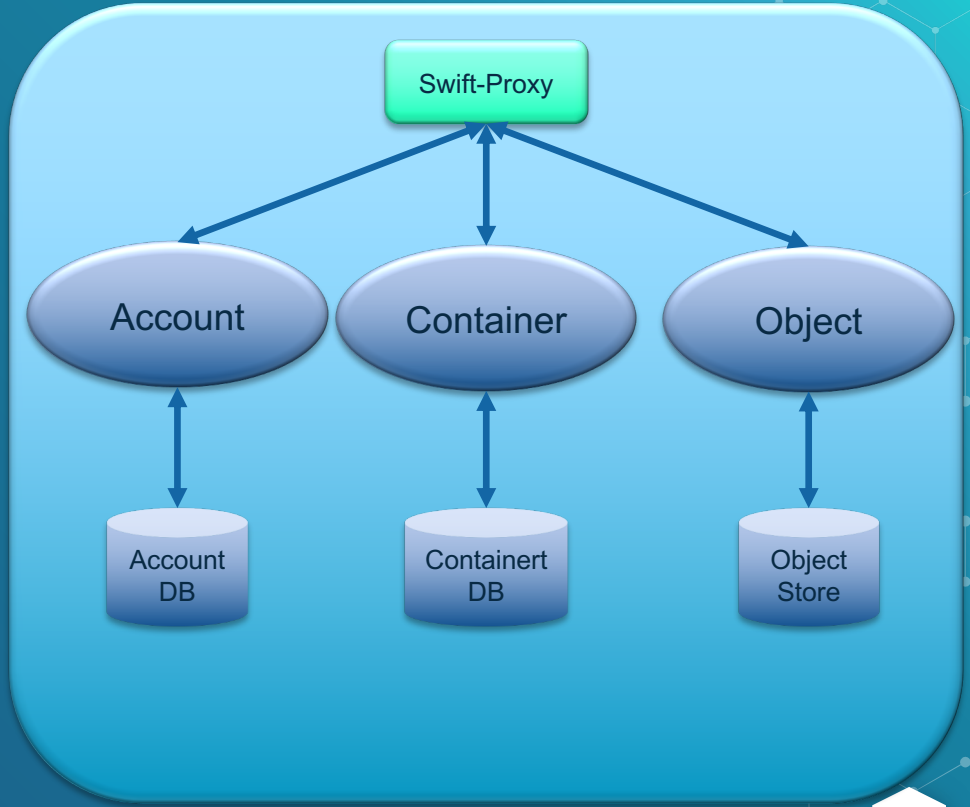
Cinder – Block Storage Service

- Provide persistend block storage resources to compute
- This functions as secondary storage (analogous to the traditional ways of locating and accessing specific locations on a disk or a drive
- Helps in creating, attaching and detaching of new block devices in the server
- Able to write images for compute to use as a bootable persistend image
- In addition to Linux Server StorageCan be integrated with Cloudbyte, EMC, Coraid, SAN Storage etc.
- Block storage is appropriate for performance sensitive scenarios like Database Storage, Expandable Filesystems or providing a server with access to raw block level storage
- Snapshot management provides powerful functionality for backing up data stored on block storage volume
- Snapshot can be restored or used to create new block storage volume.



Swift – Object Storage Service

- Provide Object Storage Service to store objects and files
- Objects can be referred using Unique Identifier instead of path
- This is the mountable storage unit
- Helps data replication across the data center
- Object storage units are replicated with every new server addition
- Swift-proxy is responsible for tying together the rest of swift architecture
- For each request, it looks up the account, container or object
- Inexpensive commodity hard drives and servers can be used
- It can be used independently without Nova

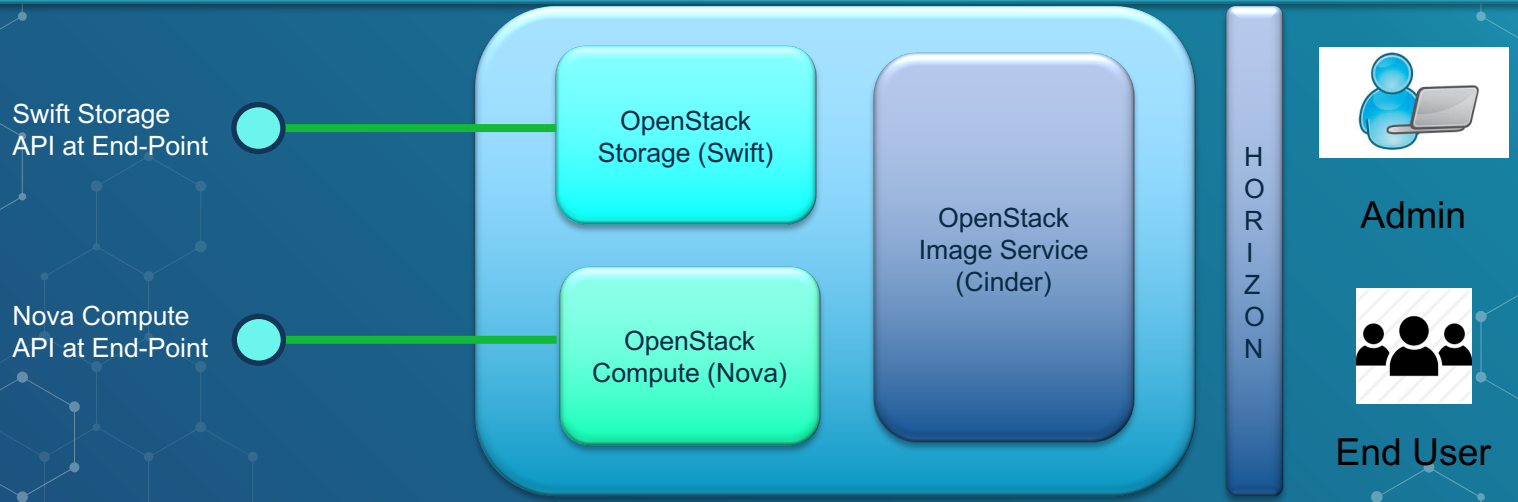


OpenStack Storage Service

On-Instance / Ephemeral	Block Storage (Cinder)	Object Storage (Swift)	File Storage (Manila)
Runs Operating Systems and provides scratch space	Used for adding additional persistent storage to a Virtual Machine (VM)	Used for storing Virtual Machine images and data	Used for providing file shares to a Virtual Machine (VM)
Persists until VM is terminated	Persists until deleted	Persist until deleted	Persists until deleted
Access associated with a VM	Access associated with a VM	Available from anywhere	Access can be provided to a VM
Implemented as a filesystem underlying OpenStack Compute	Mounted via OpenStack Block Storage controlled protocol (for example: iSCSI)	REST API	Provides Shared File System service via NFS, CIFS, GlusterFS or HDFS protocol
Encryption is available	Encryption is available	Work in progress	Encryption is not available yet
Administrator configures sized setting, based on flavors	Sizing based on need	Easily scalable for future growth	Sizing based on need
Example: 10 GB first disk, 30 GB/Core second disk	Example: 1 TB “Extra Hard Drive”	Exapmle: 10s of TBs of data set storage	Example: 1 TB of file share

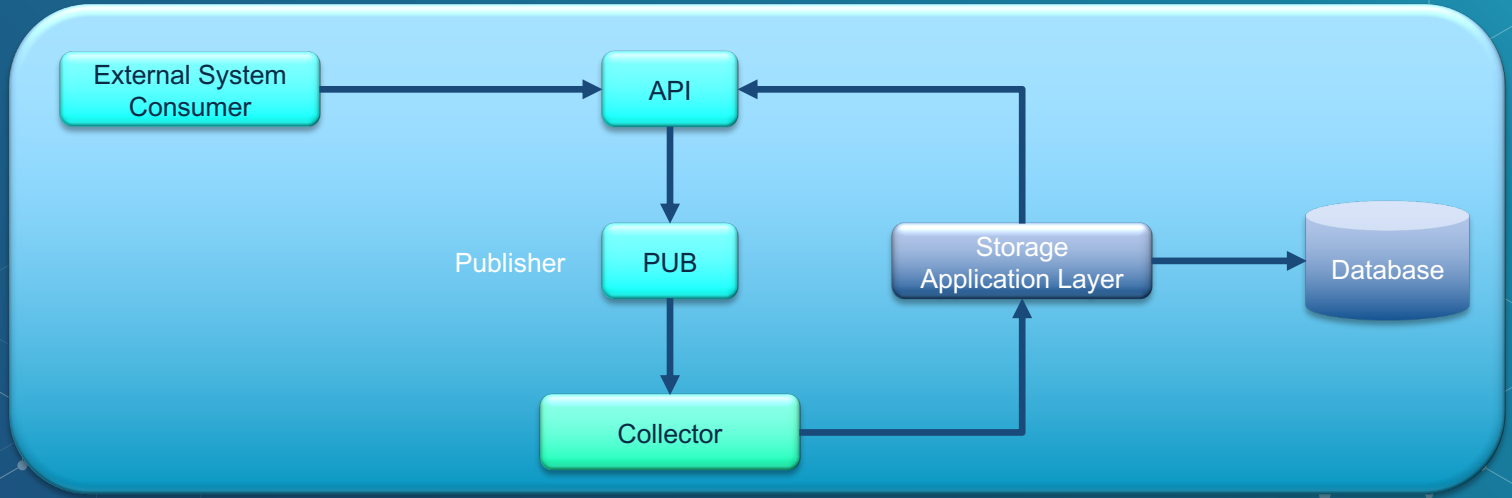
Horizon – Dashboard Service

- OpenStack Dashboard
- First component that everyone sees
- Provides all possibilities for system administrator and user to access and manage (provision and automation) of the cloud based resources
- Provides a web based interface to OpenStack Services
- Horizon is based on a Django module called django-openstack
- Developers will be able to access and deal with all components through API
- Dashboard is also brandable for service provider



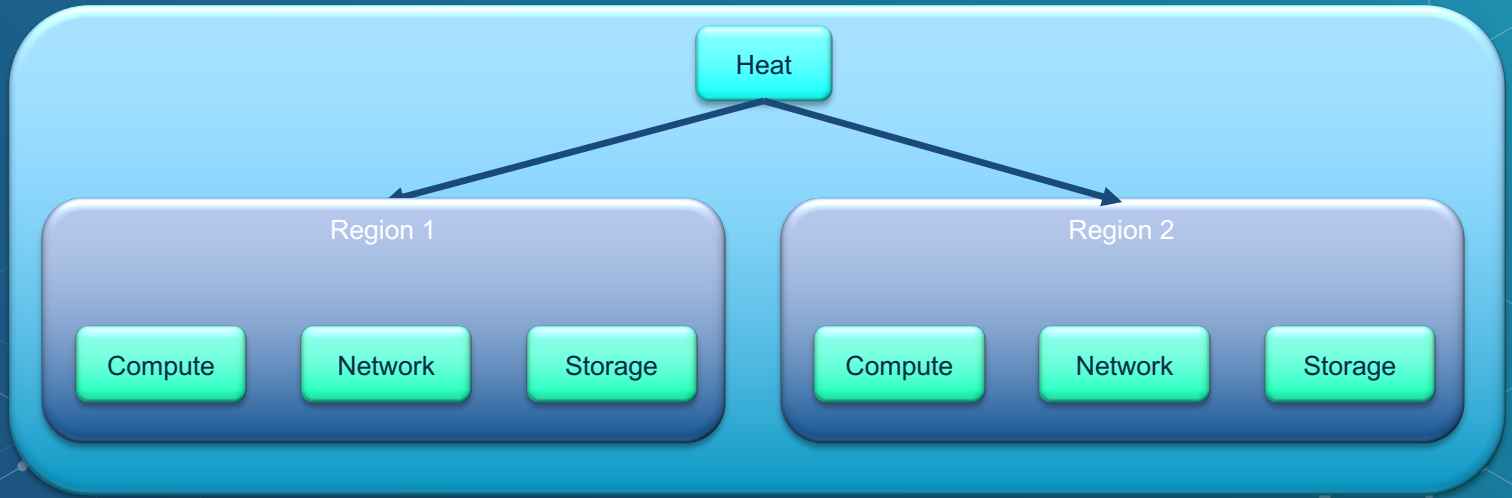
Ceilometer – Telemetry Service

- Part of telemetry project that provides telemetry services to its users
- Offers metering and billing to its consumer
- The delivery of established counters are traceable and auditable
- The counters are easily extensible to support new projects
- The agents doing data collections should be independent of overall system
- Line of items for billing

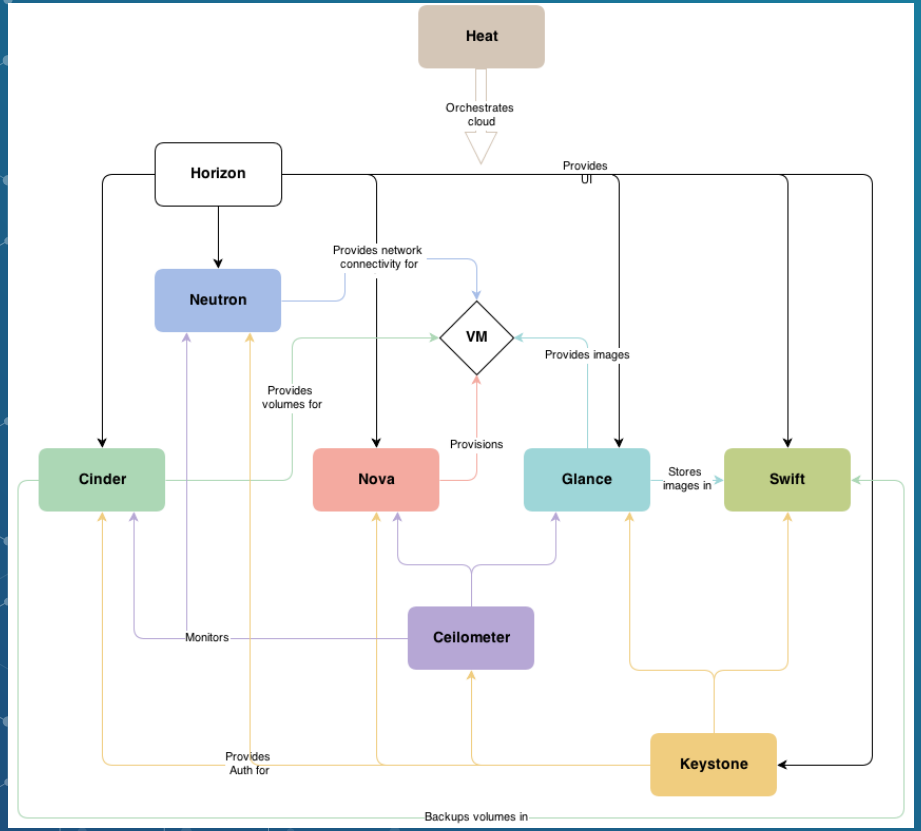


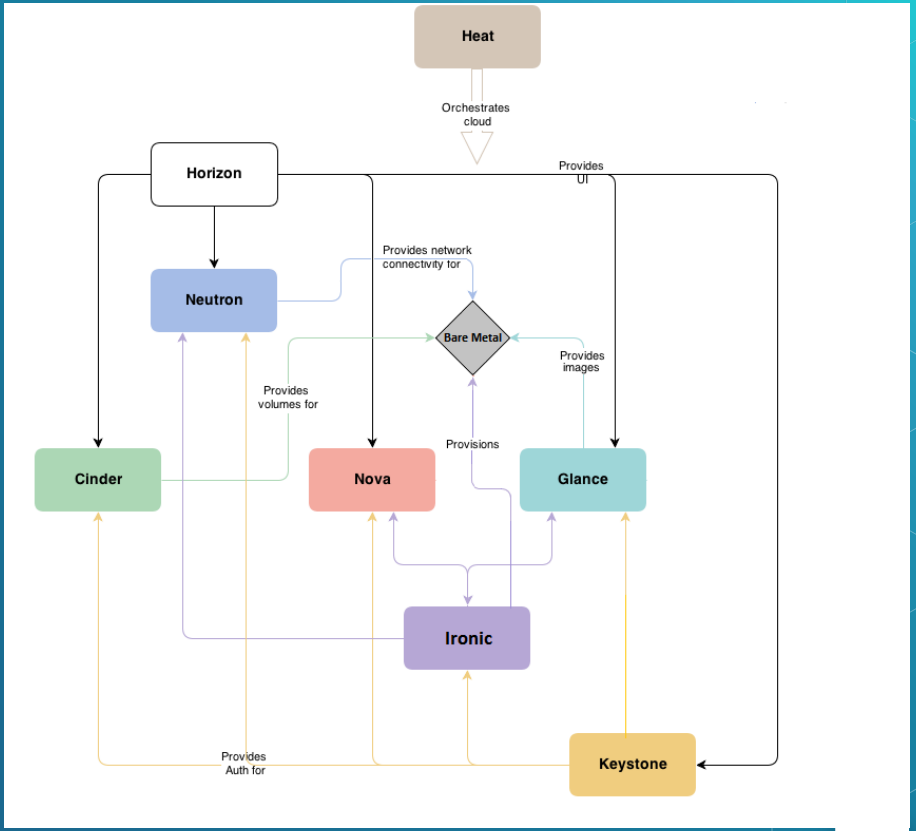
Heat – Orchestration Service

- Provide REST API to orchestrate (How each service would consume, or service will get triggered) different Services
- It uses template
- Developers store their requirement of cloud application in a file (Template)
- By trigerring the requirement, it assess the requirement of all necessary resources and provide the infrastructure to manage cloud application

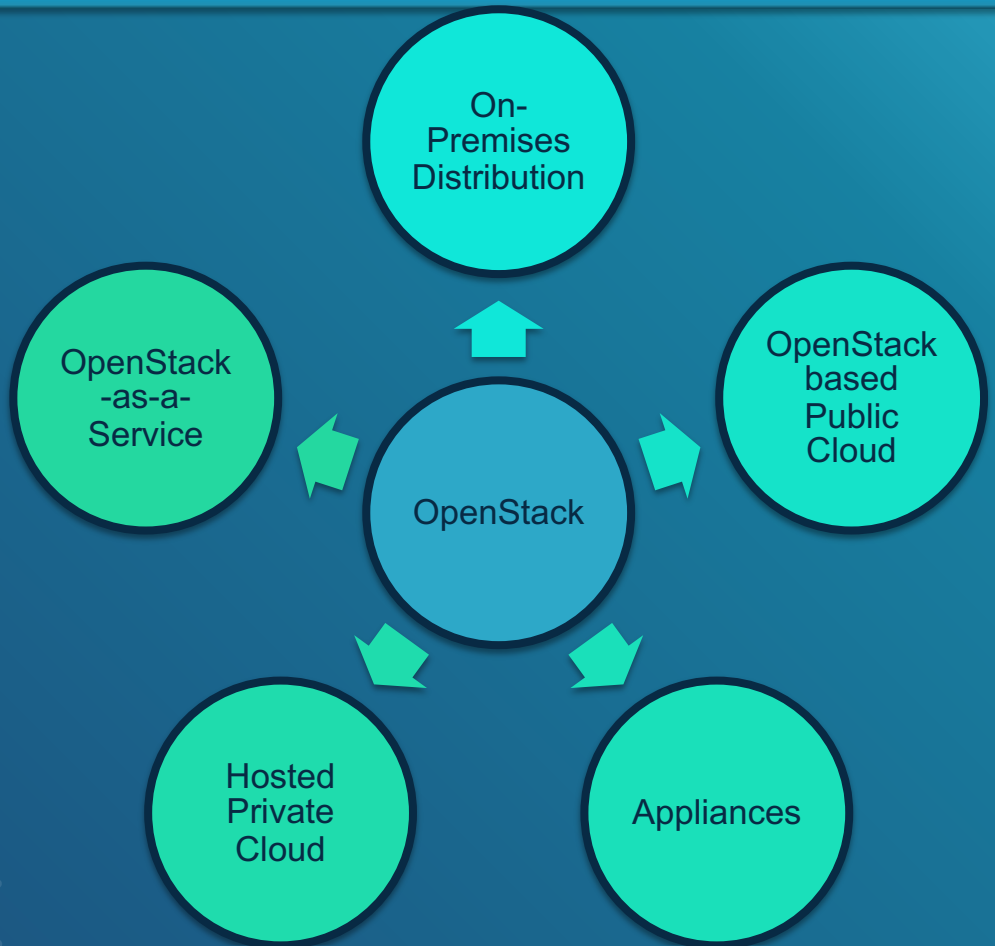


OpenStack IaaS

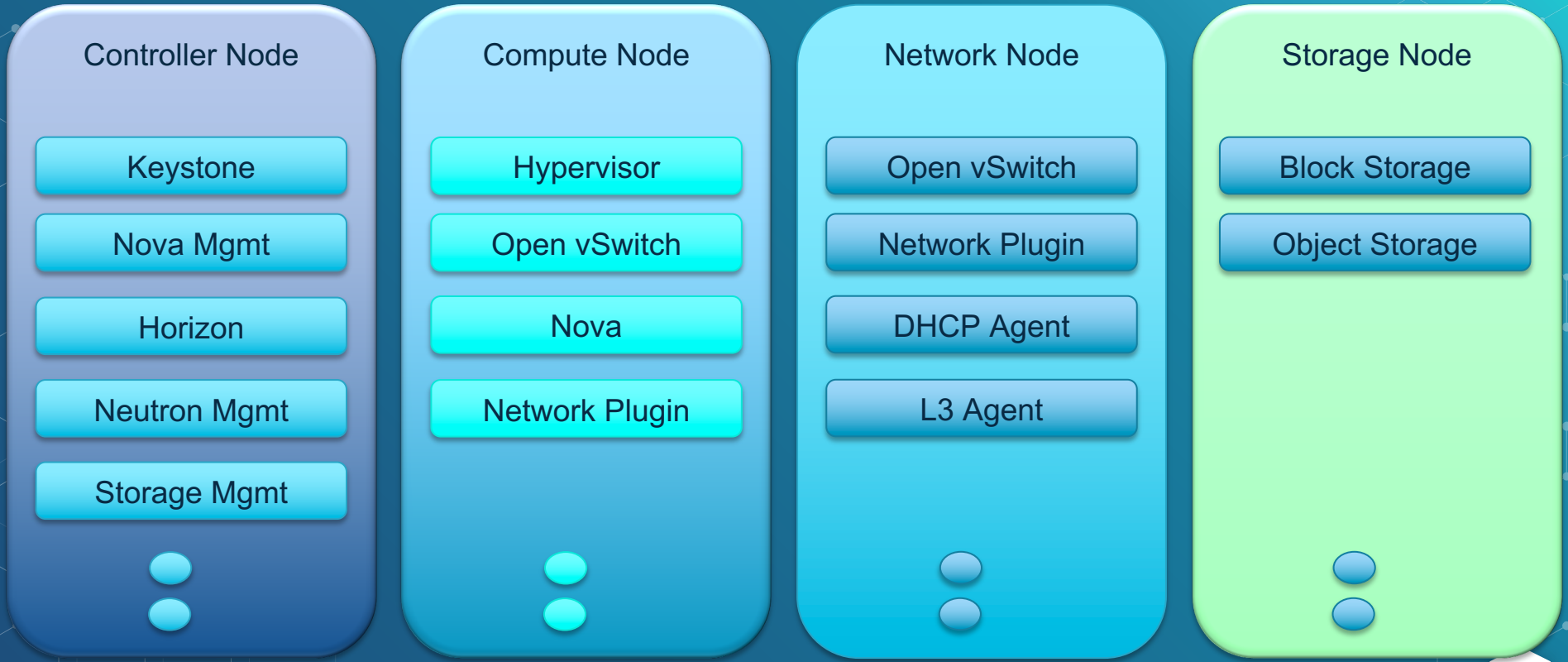




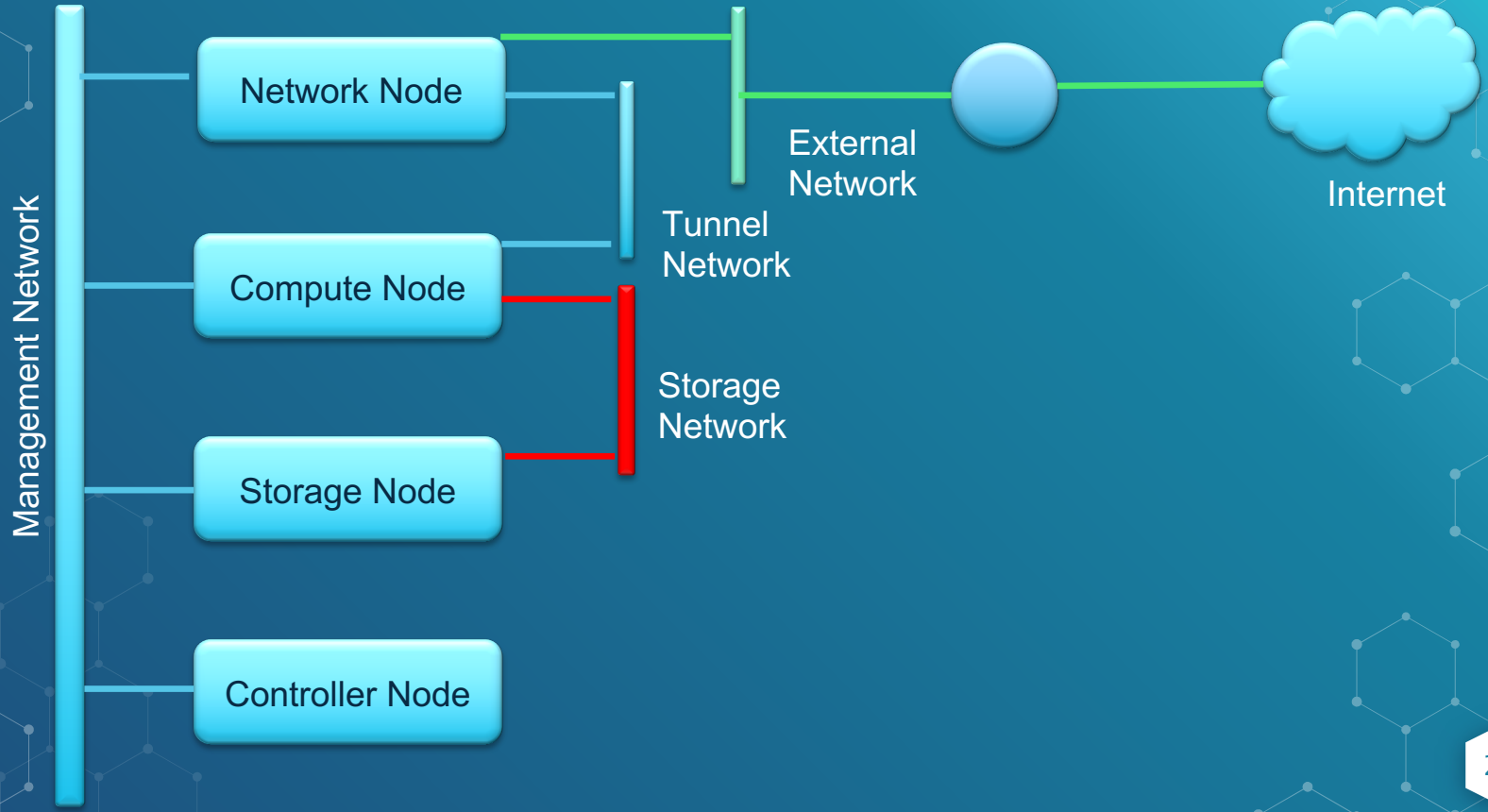
OpenStack Deployment Model



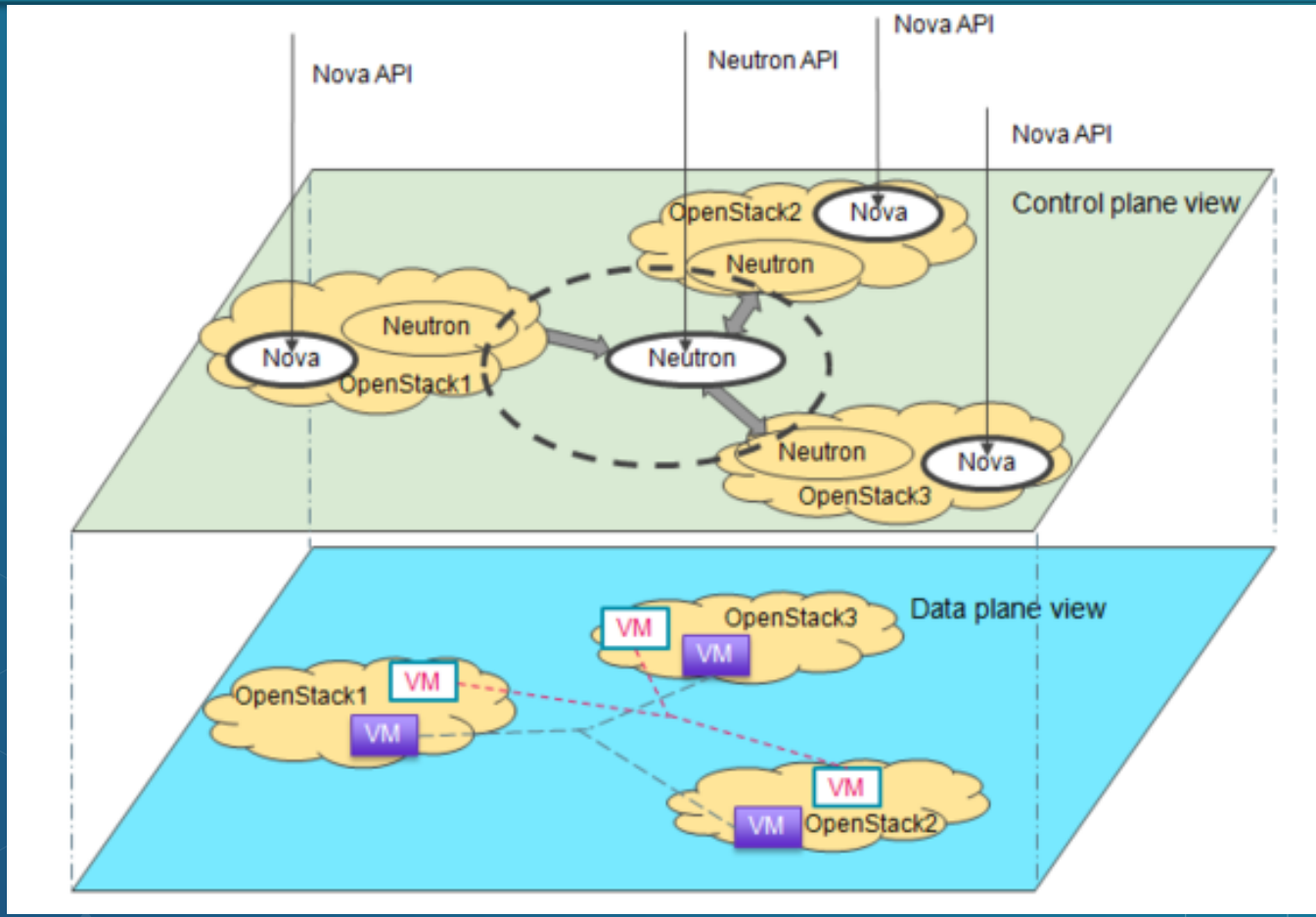
OpenStack Service Layout



OpenStack Node Connectivity



Virtual Data Center (vDC)



THANKS!

ANY QUESTIONS?