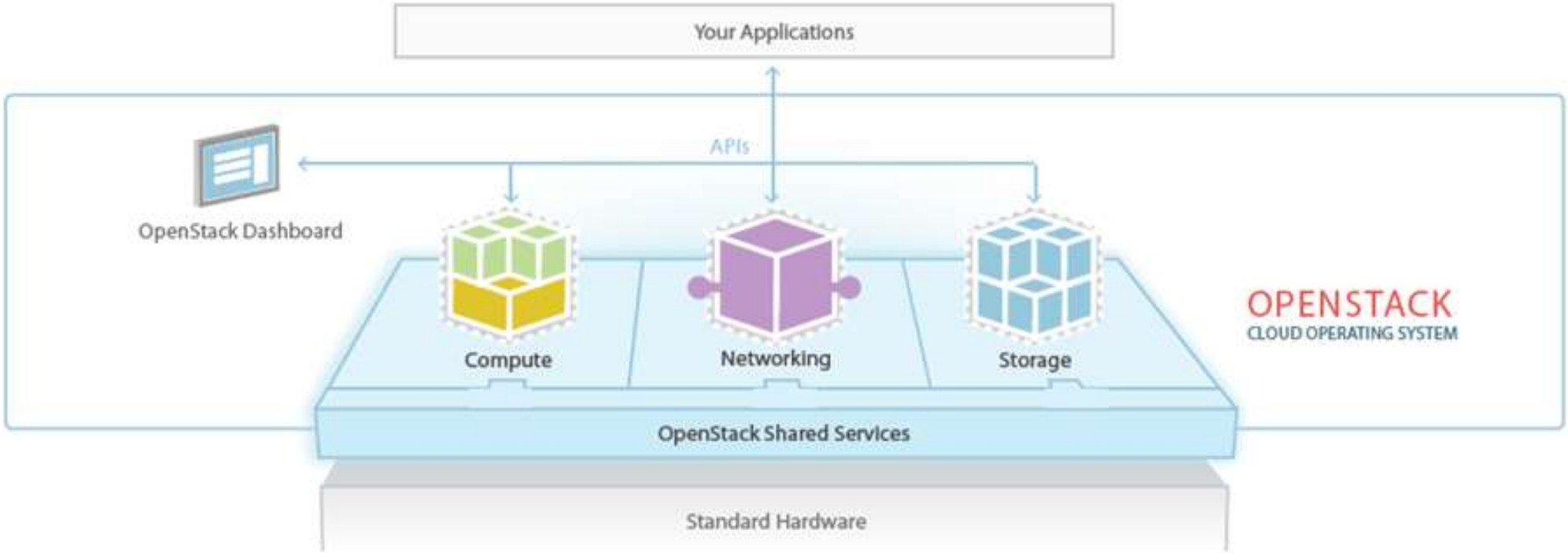
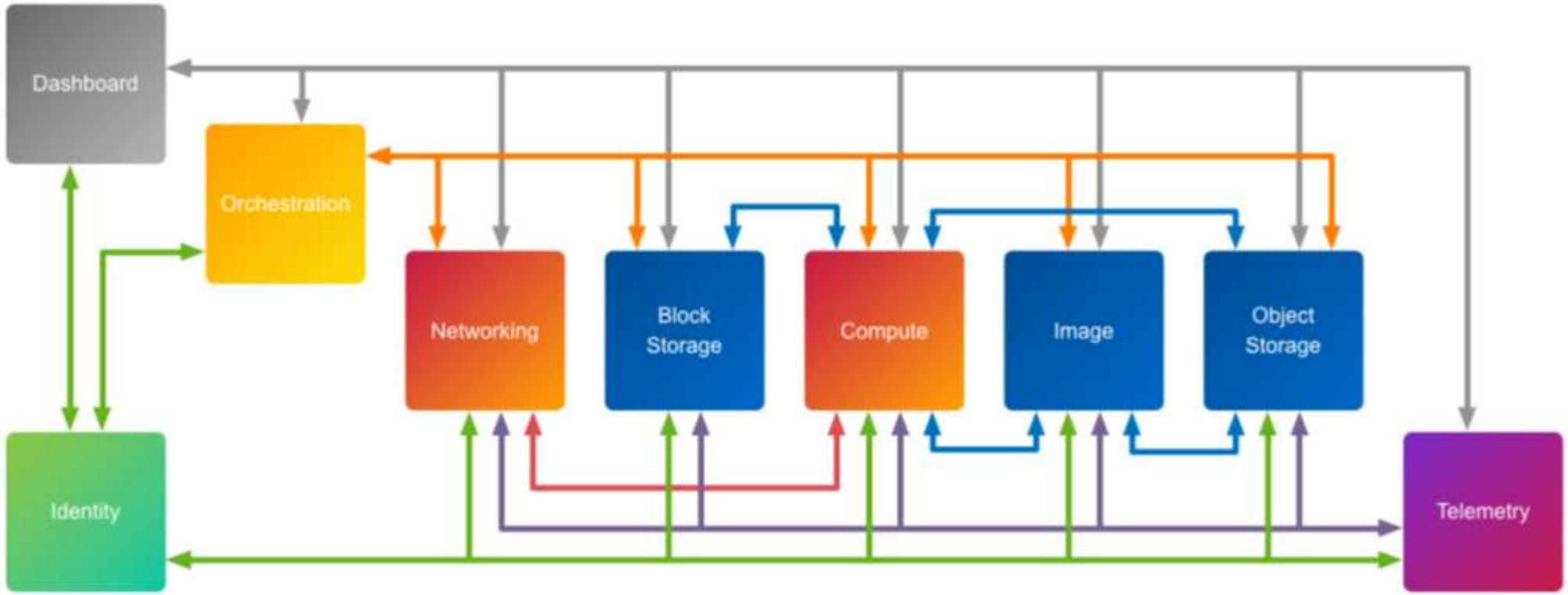
OpenStack

Ahmad Tfaily
Jalal Mostafa

OpenStack Architecture



High Level Architecture

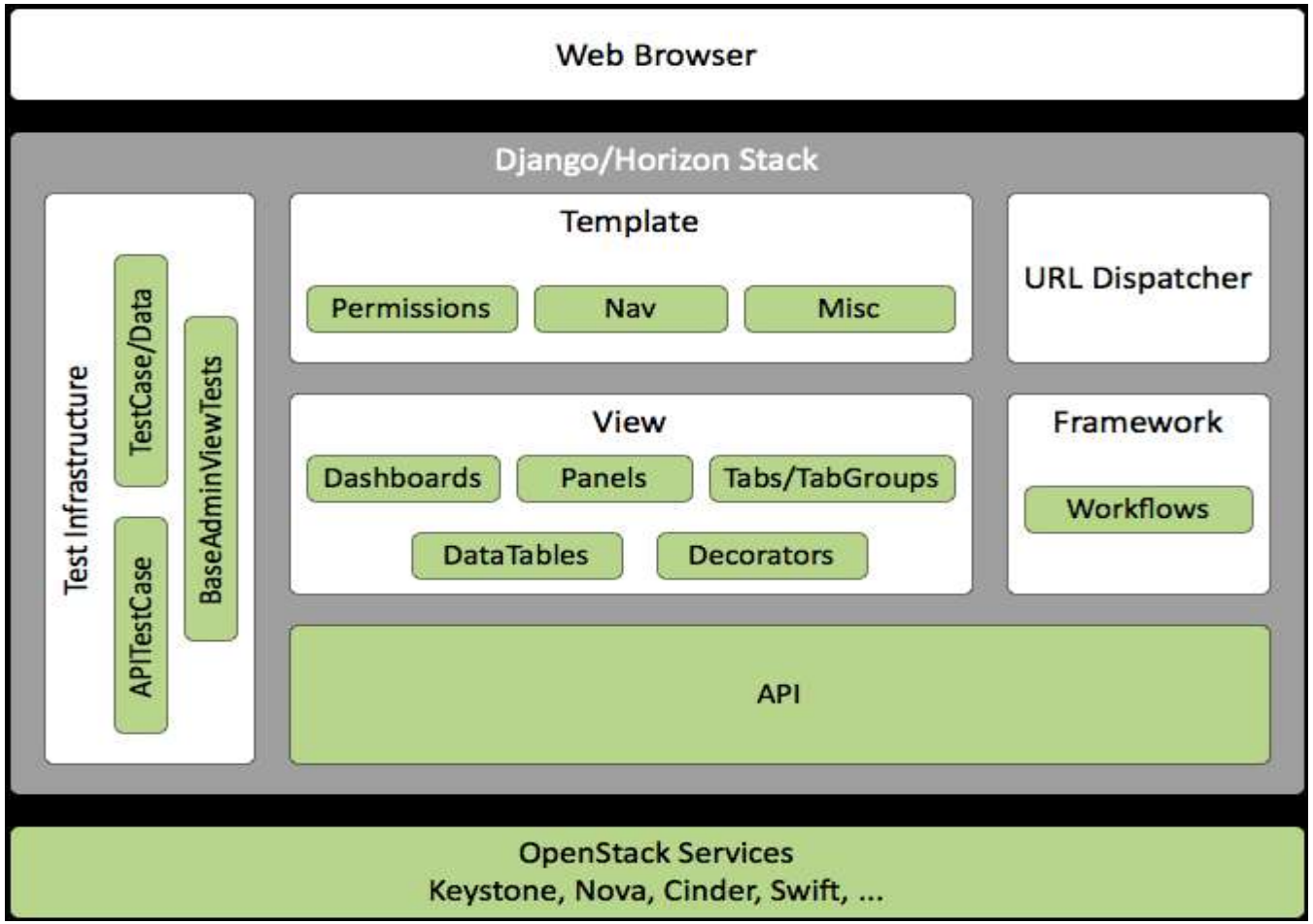


OpenStack Components

- Compute (Nova)
- Networking (Neutron)
- Block Storage (Cinder)
- Identity (Keystone)
- Image (Glance)
- Object Storage (Swift)
- Dashboard (Horizon)
- Orchestration (Heat)
- Workflow (Mistral)
- Telemetry (Ceilometer)
- Database (Trove)
- Elastic Map Reduce (Sahara)
- Bare Metal (Ironic)
- Messaging (Zaqar)
- Shared File System (Manila)
- DNS (Designate)
- Search (Searchlight)
- Key Manager (Barbican)

Horizon

- A dashboard provides administrators and users a graphical interface to access.
- such as billing, monitoring, and additional management tools for



ORACLE SOLARIS **OpenStack** Dashboard

Logged in as: glfoste | Settings | Help | Sign Out

Project

CURRENT PROJECT
sct

Manage Compute

- Overview
- Instances
- Volumes
- Images & Snapshots
- Access & Security

Manage Network

- Network Topology
- Networks
- Routers

Object Store

- Containers

Overview

Limit Summary

Instances	VCPUs	RAM	Floating IPs	Security Groups
Used 11 of 60	Used 58 of 200	Used 58.0 GB of 4.9 TB	Used 57 of 60	Used 0 of 10

Select a period of time to query its usage:

From: 2014-07-01 To: 2014-07-11 Submit The date should be in YYYY-mm-dd format.

Active Instances: 11 **Active RAM:** 58GB **This Period's VCPU-Hours:** 9.86 **This Period's GB-Hours:** 215.17

Download CSV Summary

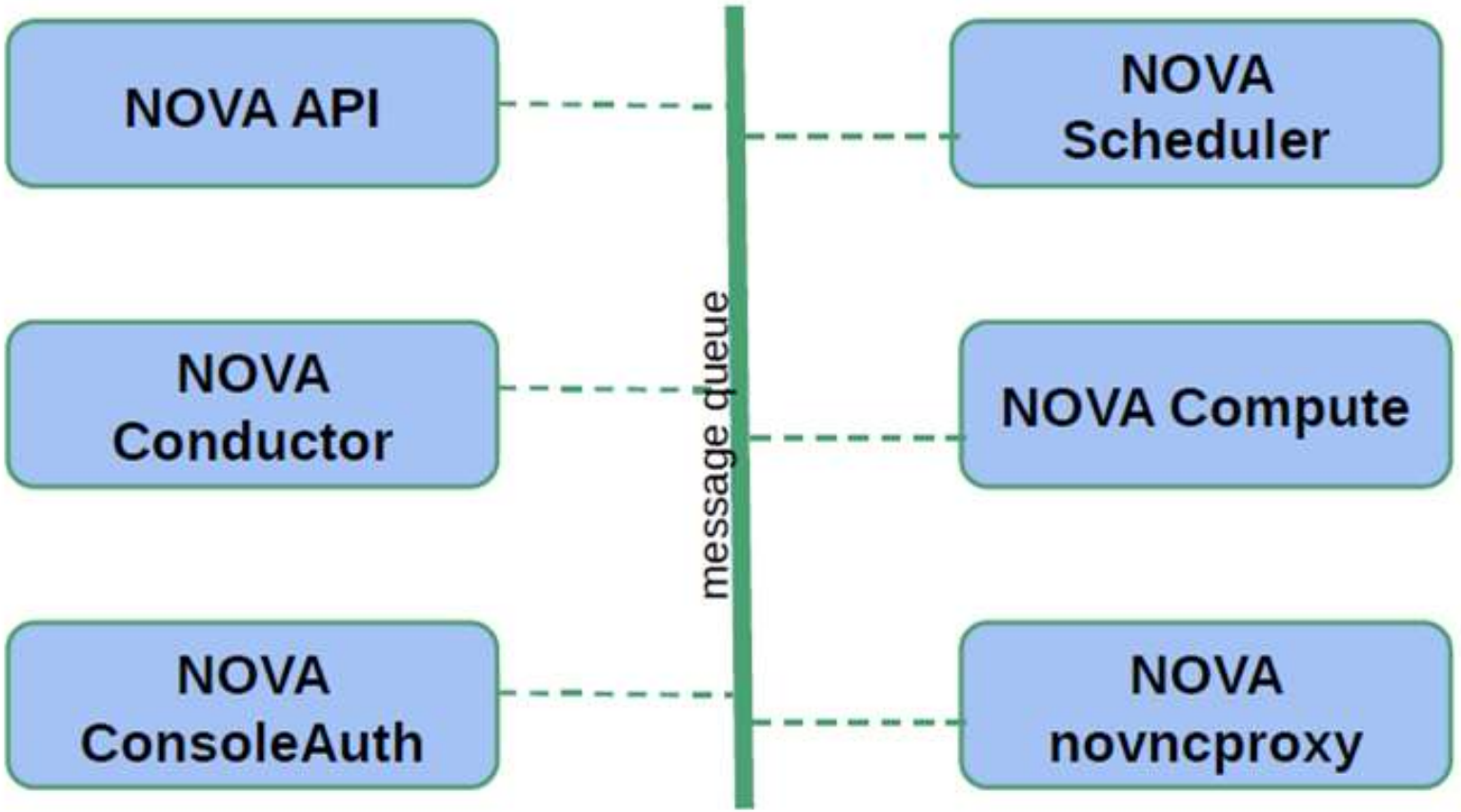
Instance Name	VCPUs	Disk	RAM	Uptime
dminer-x86-ngz	1	10	2GB	1 week
dminer-x86-1	1	10	2GB	1 week
dminer-11.2-x86	1	10	2GB	1 week
dminer-x86-2	1	10	2GB	1 week
jlbutler-x86	8	40	4GB	3 days, 2 hours

https://docs.oracle.com/cd/E36784_01/html/E52463/golpl.html

Nova

- Provides compute as a service
- The main part of an IaaS system
- It is designed to manage and automate pools of computer resources
- Compute's architecture is designed to scale horizontally

Nova - Components

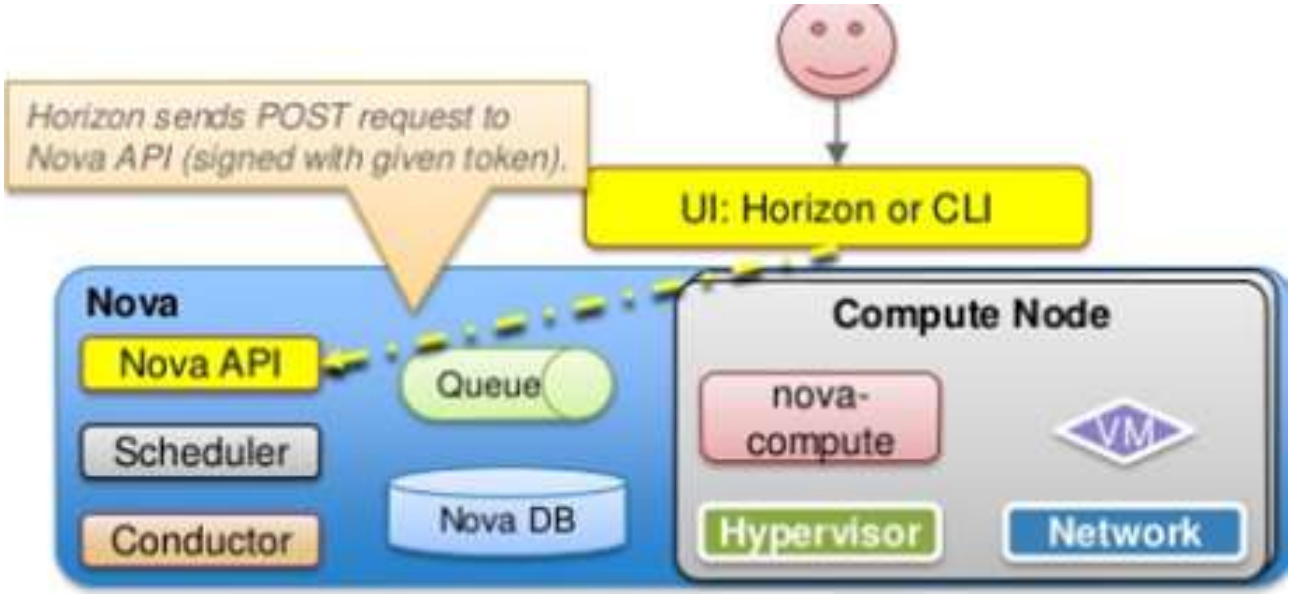


Nova - Components

- nova-conductor: Provides database-access support for Compute nodes
- nova-consoleauth: Handles console authentication
- nova-novncproxy: Provides a VNC proxy for browsers

Nova API

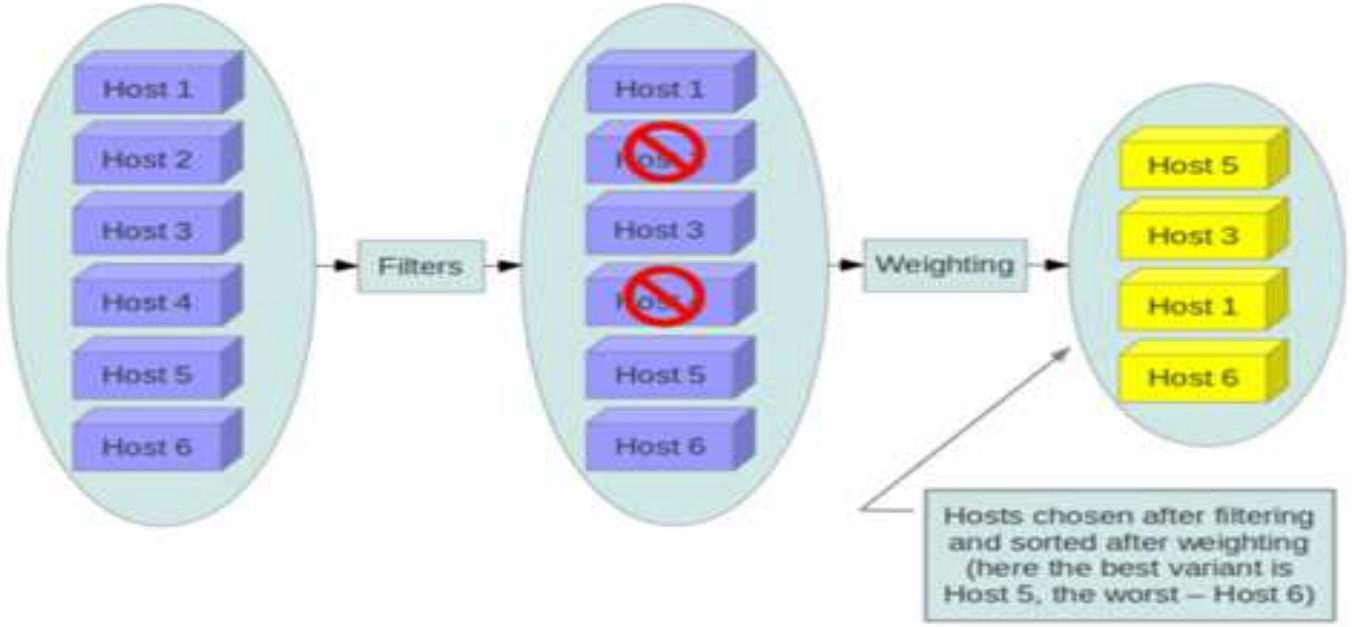
- nova-api is responsible to provide an API for users and services to interact with NOVA



https://docs.openstack.org/api-ref/compute/

Nova-scheduler:

- Using Filters,dispatches requests for new virtual machines to the correct node.



https://docs.openstack.org/nova/latest/user/filter-scheduler.html

Nova Scheduler

Vigneshvar.A.S

DEPUTY MANAGER CLOUD
RELIANCE JIO INFOCOMM
BANGALORE

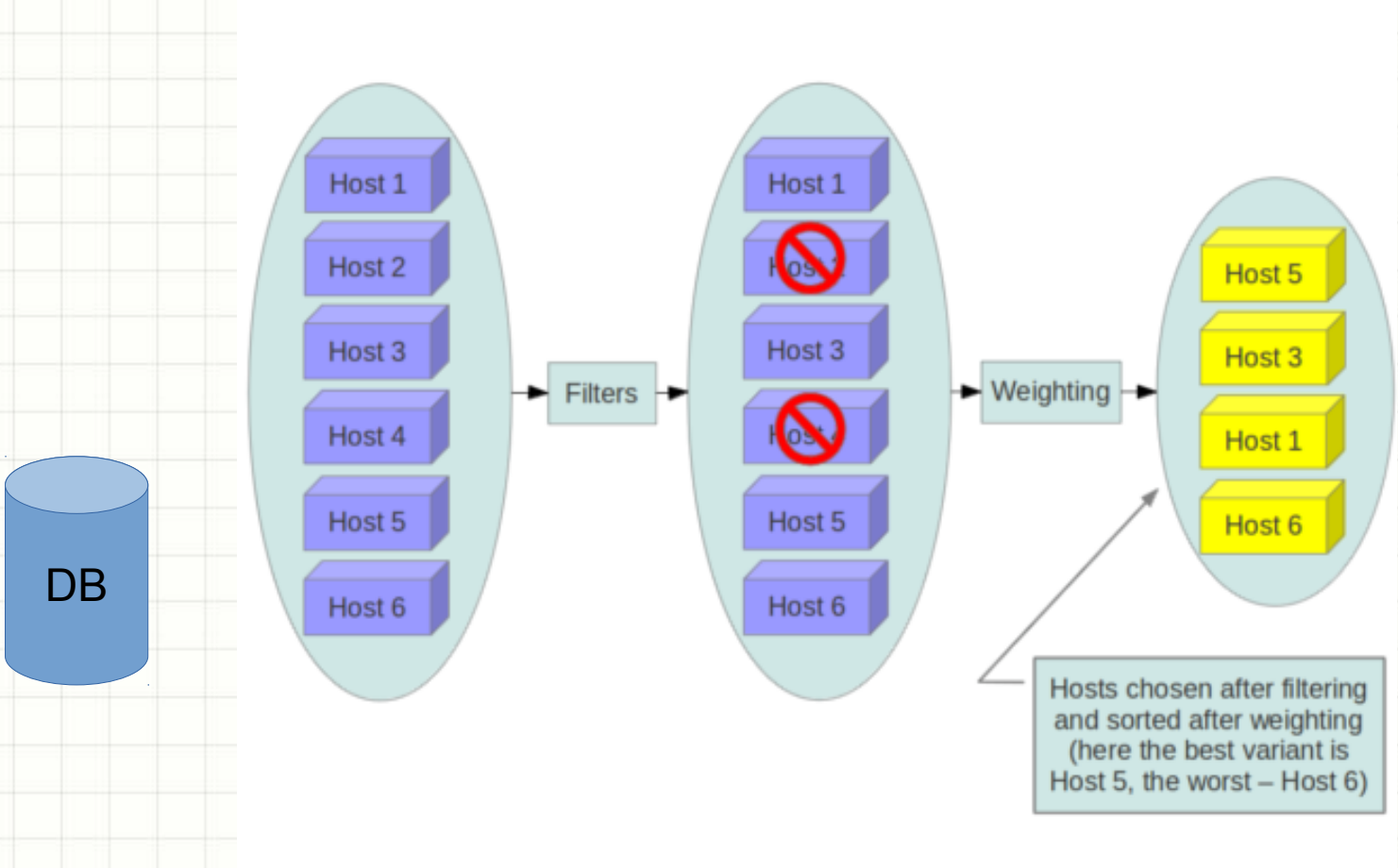
On Compute Node

- There is a periodic task (Resource Tracker), which collects host information.
- This information is then stored to DB

On Controller Node

- Request from nova API reaches conductor
- Conductor interacts with the scheduler.
- Scheduler uses filters to identify the best node from the information stored in DB
- Selected host information is sent back to conductor.
- Now conductor uses the compute queue and directs it to the selected host
- The compute node then launches the instance

Filters and Weights



There are many standard filter classes which may be used (**`nova.scheduler.filters`**):

- **`AllHostsFilter`** - does no filtering. It passes all the available hosts.
- **`ImagePropertiesFilter`** - filters hosts based on properties defined on the instance's image. It passes hosts that can support the properties specified on the image used by the instance.
- **`AvailabilityZoneFilter`** - filters hosts by availability zone. It passes hosts matching the availability zone specified in the instance properties. Use a comma to specify multiple zones. The filter will then ensure it matches any zone specified.
- **`ComputeCapabilitiesFilter`** - checks that the capabilities provided by the host compute service satisfy any extra specifications associated with the instance type. It passes hosts that can create the specified instance type.

https://docs.openstack.org/nova/latest/user/filter-scheduler.html

Filters and Weights

Some common filters are,

AvailabilityZoneFilter:

Return hosts where node_availability_zone name is the same as the one requested.

RamFilter:

Return hosts where (free ram * ram_allocation_ratio) is greater than requested ram.

ComputeFilter:

Return hosts where asked instance_type (with extra_specs) match capabilities

Filters and Weights

DiskFilter:

Returns hosts with sufficient disk space available for root and ephemeral storage.

RetryFilter:

Filters out hosts that have already been attempted for scheduling purposes.

Filters and Weights

Weights:

Scheduler applies cost function on each host and calcaulates the weight.

Some of cost functions could be

- Considering Free RAM among filtered hosts. Highest free RAM wins
- Considering least workload (io ops) among filtered hosts.
- Can consider any specific metric we want to consider in a similar fashion. Can be enabled from configuration file

- **RAMWeigher** Compute weight based on available RAM on the compute node. Sort with the largest weight winning.

- **CPUWeigher** Compute weight based on available vCPUs on the compute node. Sort with the largest weight winning.

- **DiskWeigher** Hosts are weighted and sorted by free disk space with the largest weight winning.

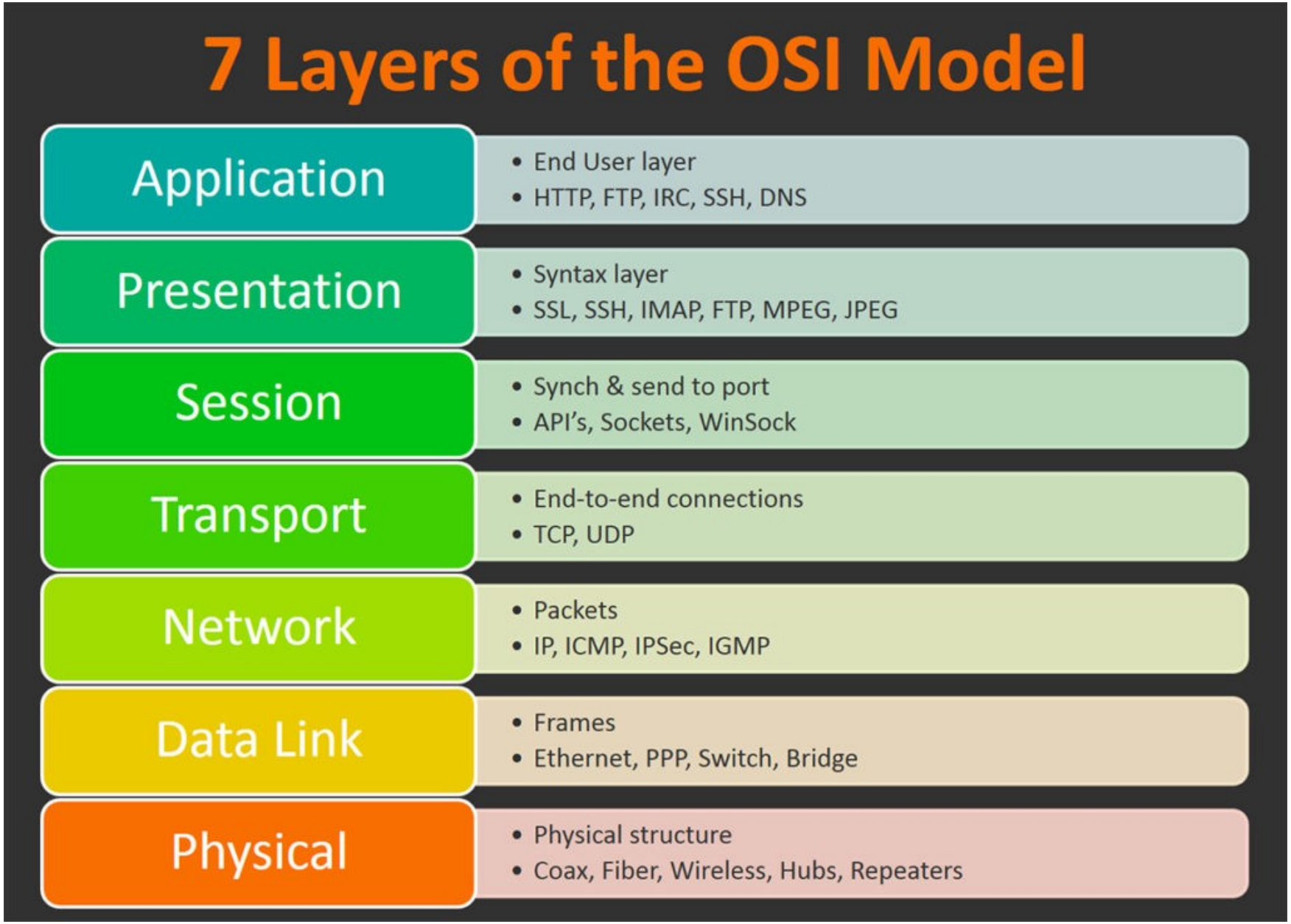
- **MetricsWeigher** This weigher can compute the weight based on the compute node host's various metrics. The to-be weighed metrics and their weighing ratio are specified in the configuration file as the followings:

```
metrics_weight_setting = name1=1.0, name2=-1.0
```

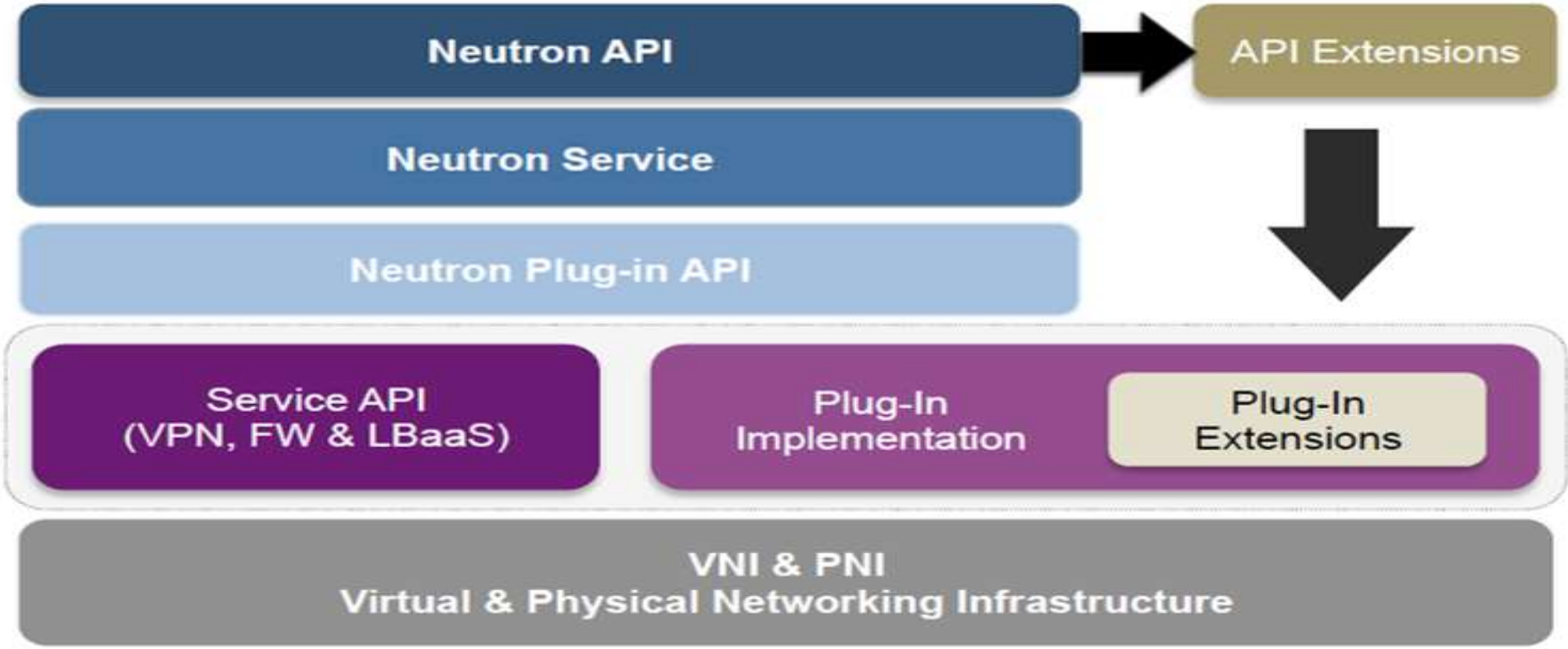
Neutron

- Network as a Service (NaaS)
- Provides REST APIs to manage network connections for the resources managed by other OpenStack Services
- Complete control over the following network resources in OpenStack(Networks, Ports and Subnets)
- Build complex network topologies
- Limited L3 functionality (IP tables rules at host level)

7	Application Layer	Human-computer interaction layer, where applications can access the network services
6	Presentation Layer	Ensures that data is in a usable format and is where data encryption occurs
5	Session Layer	Maintains connections and is responsible for controlling ports and sessions
4	Transport Layer	Transmits data using transmission protocols including TCP and UDP
3	Network Layer	Decides which physical path the data will take
2	Data Link Layer	Defines the format of data on the network
1	Physical Layer	Transmits raw bit stream over the physical medium


7 Layers of the OSI Model
Application
End User layer
HTTP, FTP, IRC, SSH, DNS
Presentation
Syntax layer
SSL, SSH, IMAP, FTP, MPEG, JPEG
Session
Synch & send to port
API's, Sockets, WinSock
Transport
End-to-end connections
TCP, UDP
Network
Packets
IP, ICMP, IPSec, IGMP
Data Link
Frames
Ethernet, PPP, Switch, Bridge
Physical
Physical structure
Coax, Fiber, Wireless, Hubs, Repeaters

Neutron Architecture



VPN: Virtual Private Network; FW: Firewall; LBaaS: Load Balancing as a Service

Neutron Plug-Ins

- Modular Layer 2 (ML2)
- Linux Bridge
- Open vSwitch

The Modular Layer 2 (ml2) plugin is a framework allowing OpenStack Networking to simultaneously utilize the variety of layer 2 networking technologies found in complex real-world data centers.

It currently works with the existing openvswitch, linuxbridge, and hyperv L2 agents, and is intended to replace and deprecate the monolithic plugins associated with those L2 agents.

The ml2 framework is also intended to greatly simplify adding support for new L2 networking technologies, requiring much less initial and ongoing effort than would be required to add a new monolithic core plugin. A modular agent may be developed as a follow-on effort.

source: https://wiki.openstack.org/wiki/Neutron/ML2

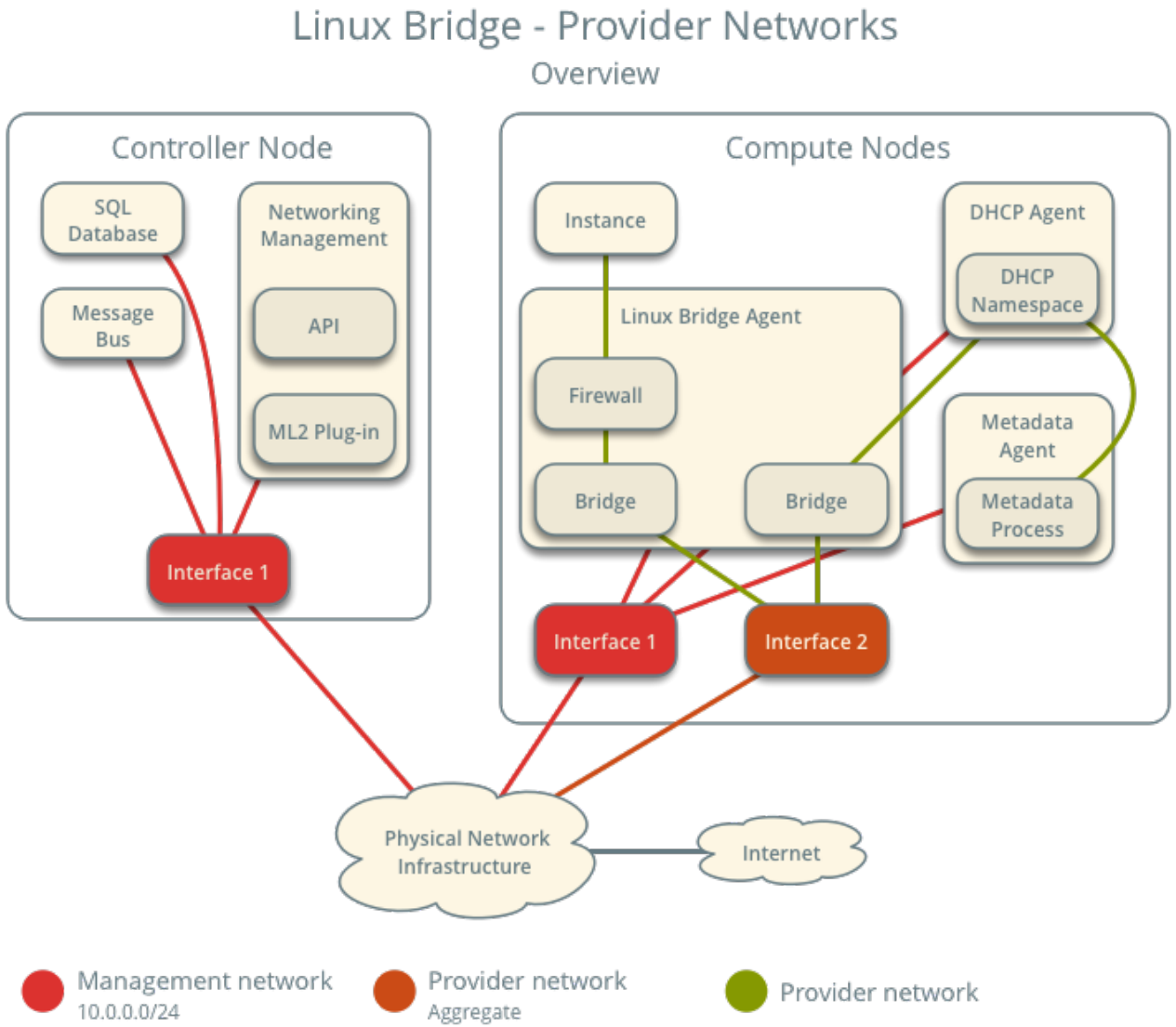
Neutron Plug-Ins

- Modular Layer 2 (ML2)
- Linux Bridge
- Open vSwitch

ML2 uses the Linux Bridge to provide L2 connectivity for VM instances running on the compute node to the public network.

In most common deployments, there is a compute and a network node. On both the compute and the network node, the Linux Bridge Agent will manage virtual switches, connectivity among them, and interaction via virtual ports with other network components such as namespaces and underlying interfaces.

Additionally, on the compute node, the Linux Bridge Agent will manage security groups.



Source: https://docs.openstack.org/neutron/queens/admin/deploy-lb-provider.html#architecture

Neutron Plug-Ins

- Modular Layer 2 (ML2)
- Linux Bridge
- Open vSwitch

Open vSwitch is an open-source project that allows hypervisors to virtualize the networking layer. This caters for the large number of virtual machines running on one or more physical nodes. The virtual machines connect to virtual ports on virtual bridges (inside the virtualized network layer.)

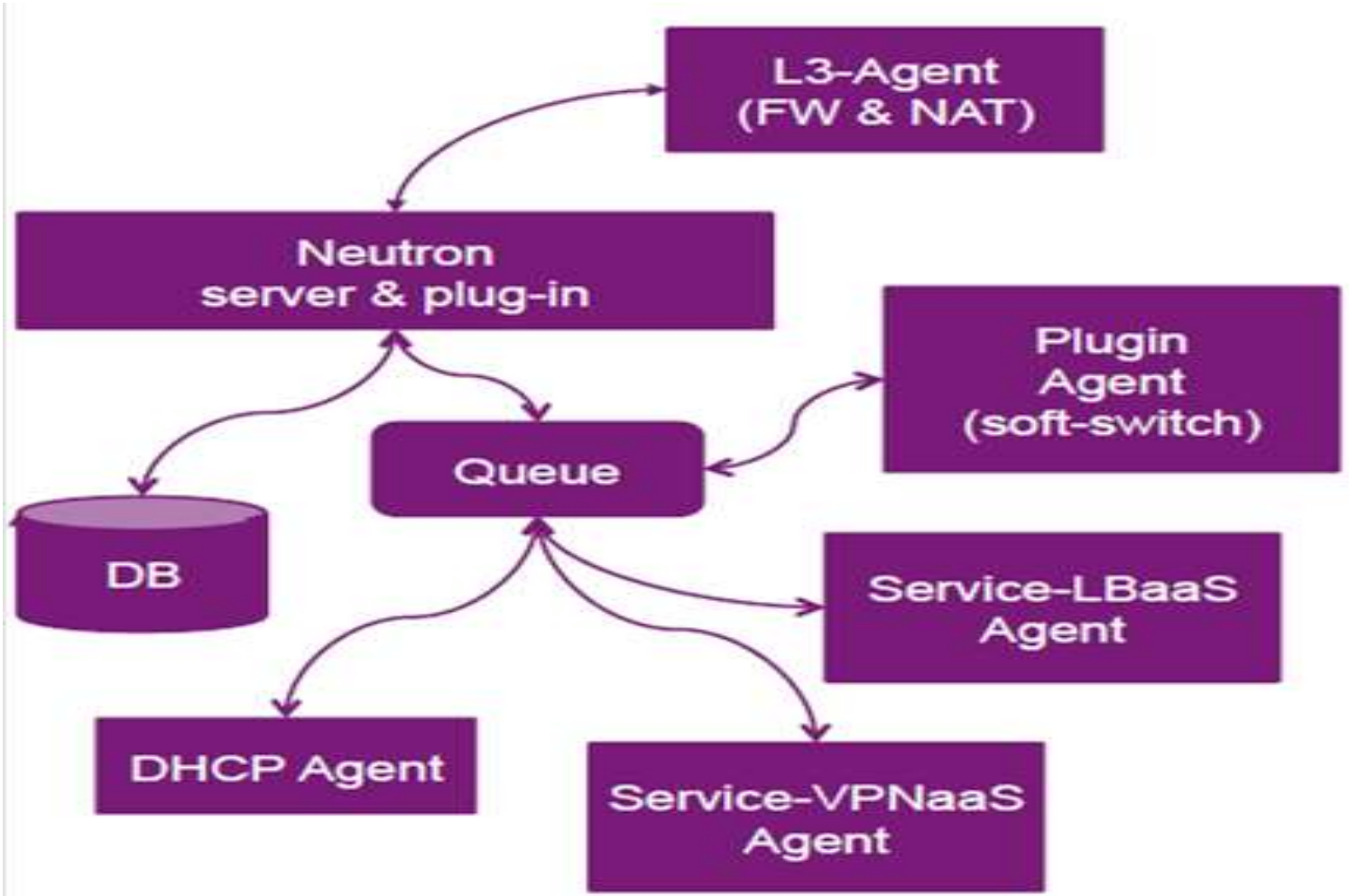
This is very similar to a physical server connecting to physical ports on a Layer 2 networking switch. These virtual bridges then allow the virtual machines to communicate with each other on the same physical node. These bridges also connect these virtual machines to the physical network for communication outside the hypervisor node.

In OpenStack, both the Neutron node and the compute node (Nova) are running Open vSwitch to provide virtualized network services.

Neutron Services

- Load Balancer as a Service (LBaaS)
- Virtual Private Network as a Service (VPNaaS)
- Firewall as a Service (FWaaS)

Neutron Components



Neutron Components

- Neutron Server
  - Implement REST APIs
  - Enforce network model
  - Network, subnet, and port
  - IP addressing to each port (IPAM)
- Plugin agent
  - Run on each compute node
  - Connect instances to network port
- Queue
  - Enhance communication between each
  - components of neutron
- Database
  - Persistent network model

Neutron Components

- DHCP Agent (*)
  - In multi-host mode, run on each compute node
  - Start/stop dhcp server
  - Maintain dhcp configuration
- L3 Agent (*)
  - To implement floating Ips and other L3 features,such as NAT
  - One per network

References

- http://openstack.org
  - Tokyo Summit 2015 https://www.openstack.org/summit/tokyo-2015/
  - Austin Summit 2016 https://www.openstack.org/summit/austin-2016/
  - Barcelona Summit 2016
  - https://wiki.openstack.org/
- http://linux.com
  - Linux Foundation Blog https://www.linux.com/blog/learn/chapter/openstack/essentials-openstack-administration-part-1-cloud-fundamentals
- https://www.opnfv.org/
- https://www.sdxcentral.com/cloud/open-source/definitions/
- https://www.opendaylight.org/news/user-story/2015/11/china-mobile-builds-next-generation-network-opendaylight
- http://about.att.com/innovationblog/openstack_superuser