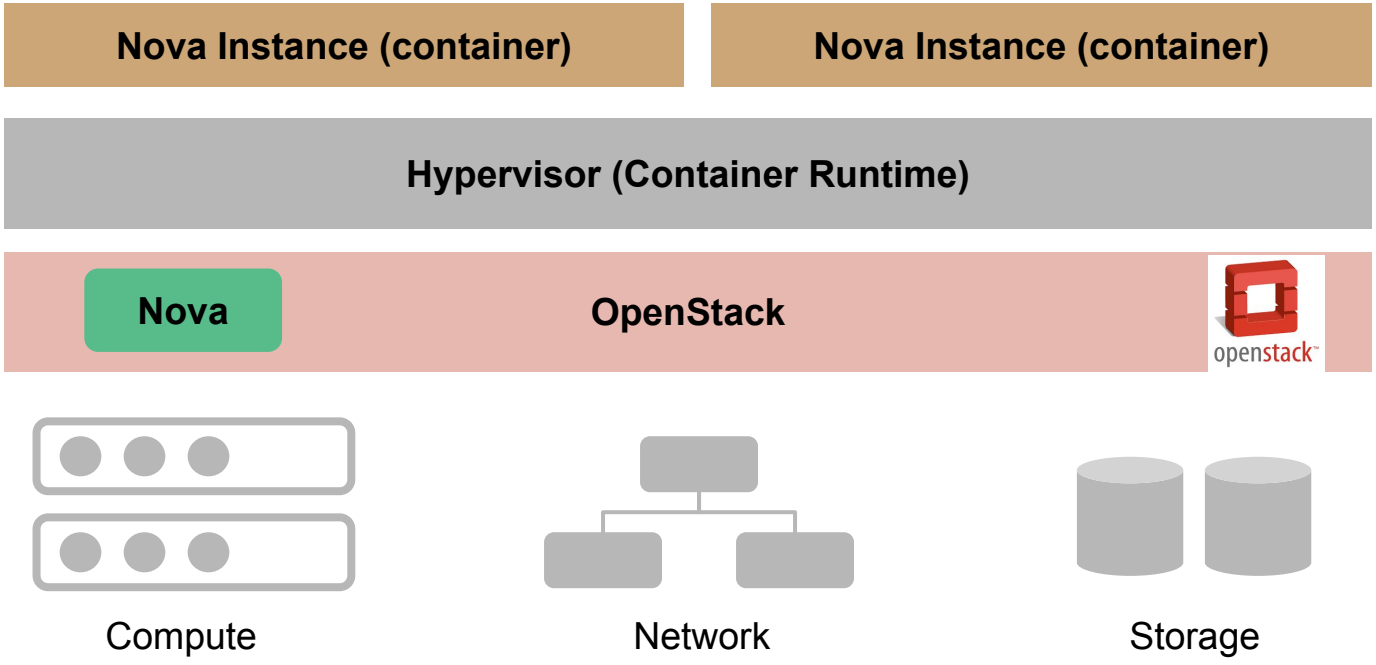
Magnum is not the OpenStack Container Service? How about Zun

Hongbin Lu (Huawei)
Qiming Teng (IBM)
Eli Qiao (Intel)
Madhuri Kumari (Intel)

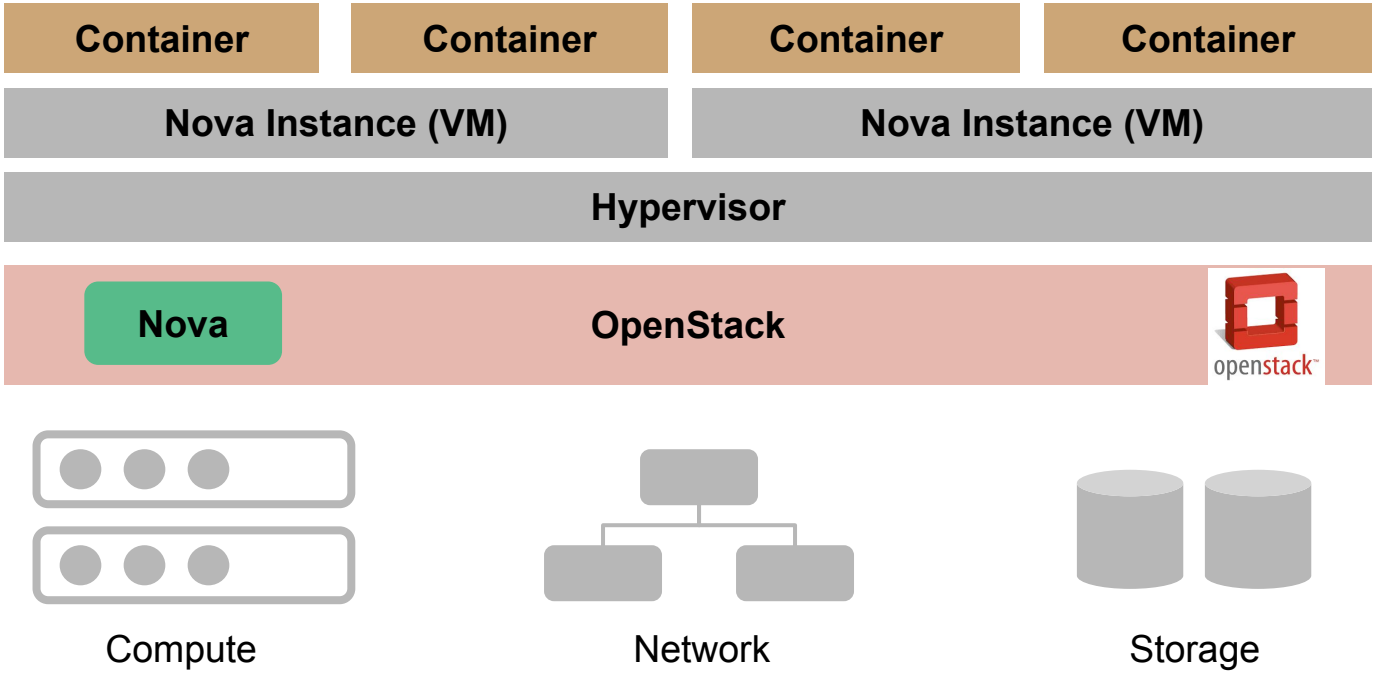
Agenda

- Containers on OpenStack
- Magnum
- Zun
- Demo

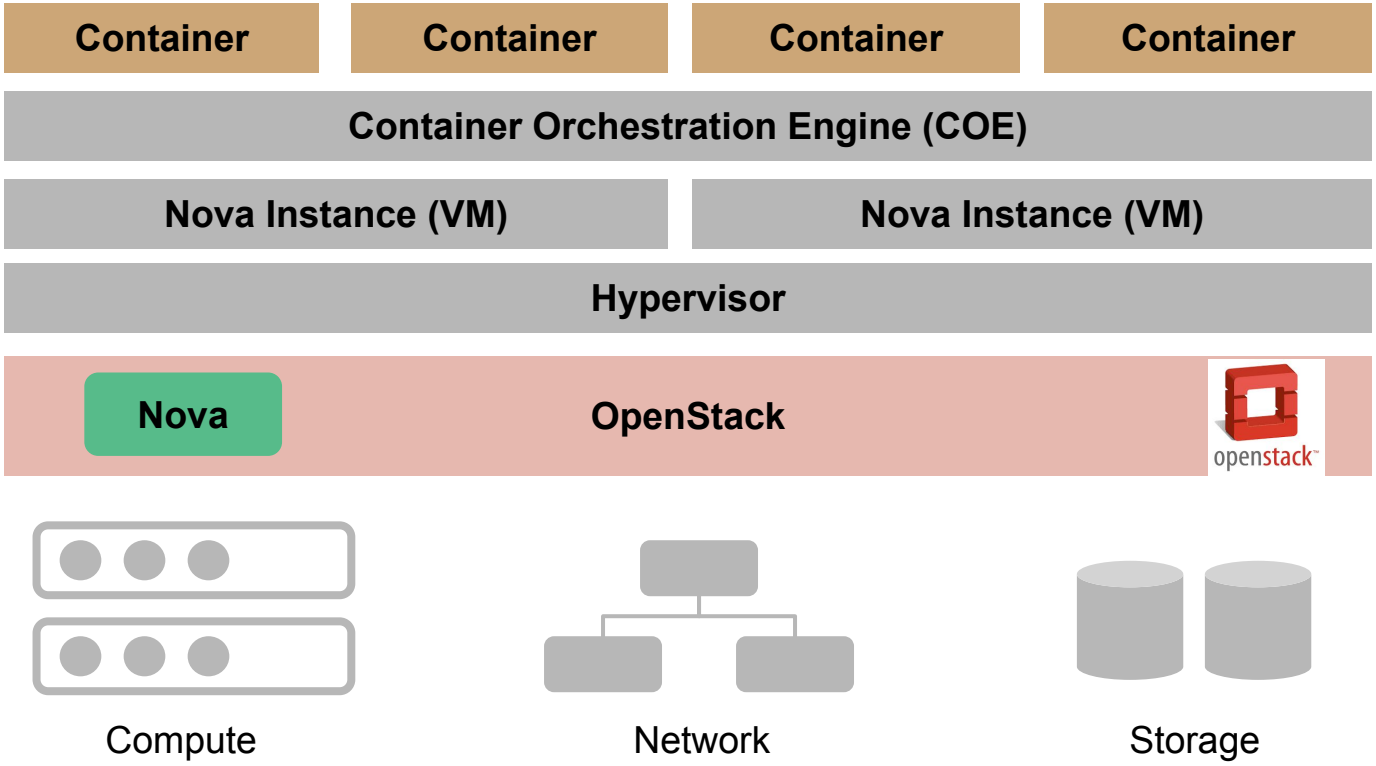
Containers on OpenStack



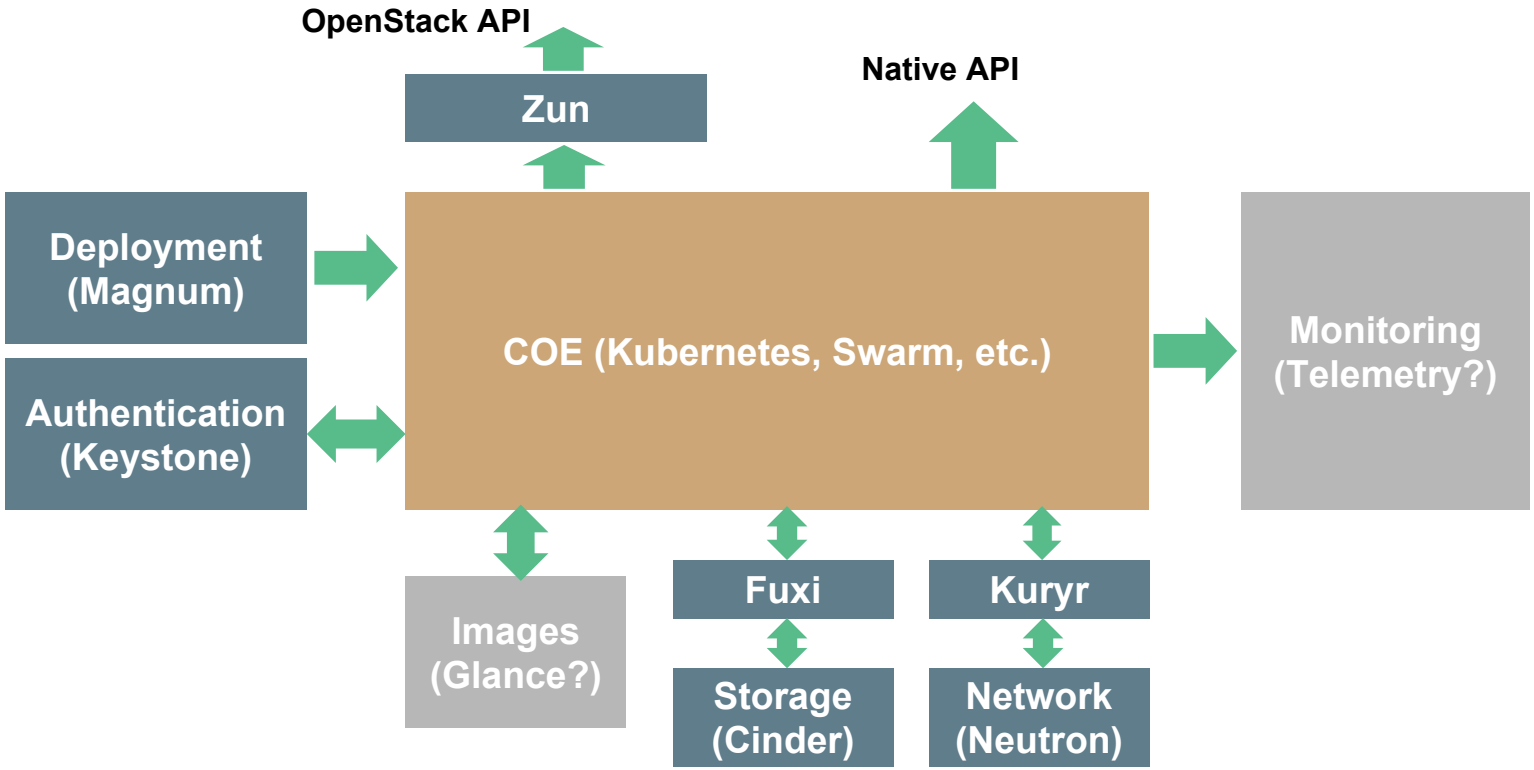
Containers on OpenStack



Containers on OpenStack



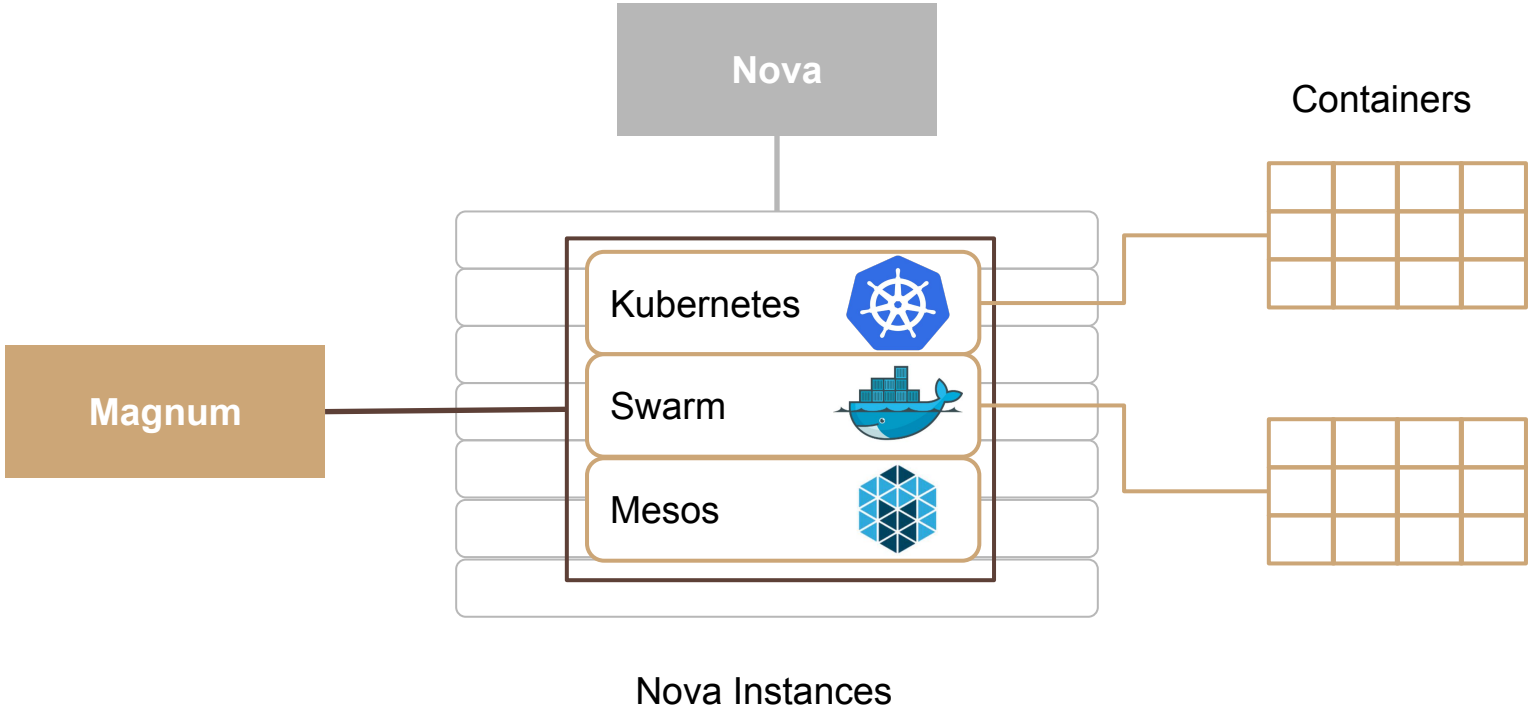
Containers on OpenStack



Agenda

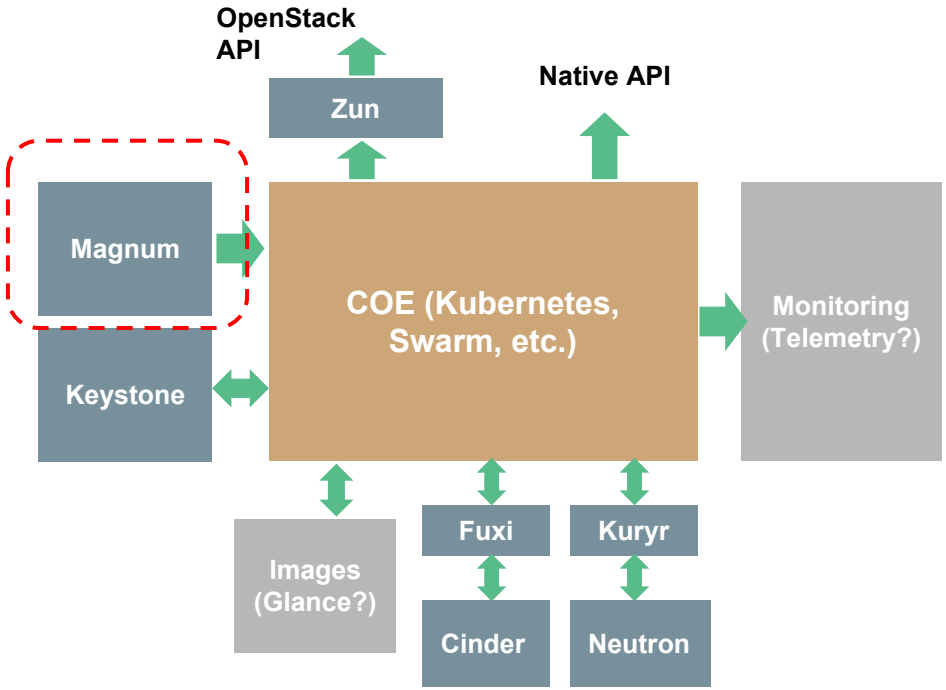
- Containers on OpenStack
- Magnum
- Zun
- Demo

Introduction to Magnum



Introduction to Magnum

- Provisioning
  - Kubernetes
  - Docker Swarm
  - Mesos
- Scaling
  - Add instances
  - Remove instances
- Security
  - Serve as Certificate Authority (CA)
  - Generate Keystone users



Magnum Mission Statement Update

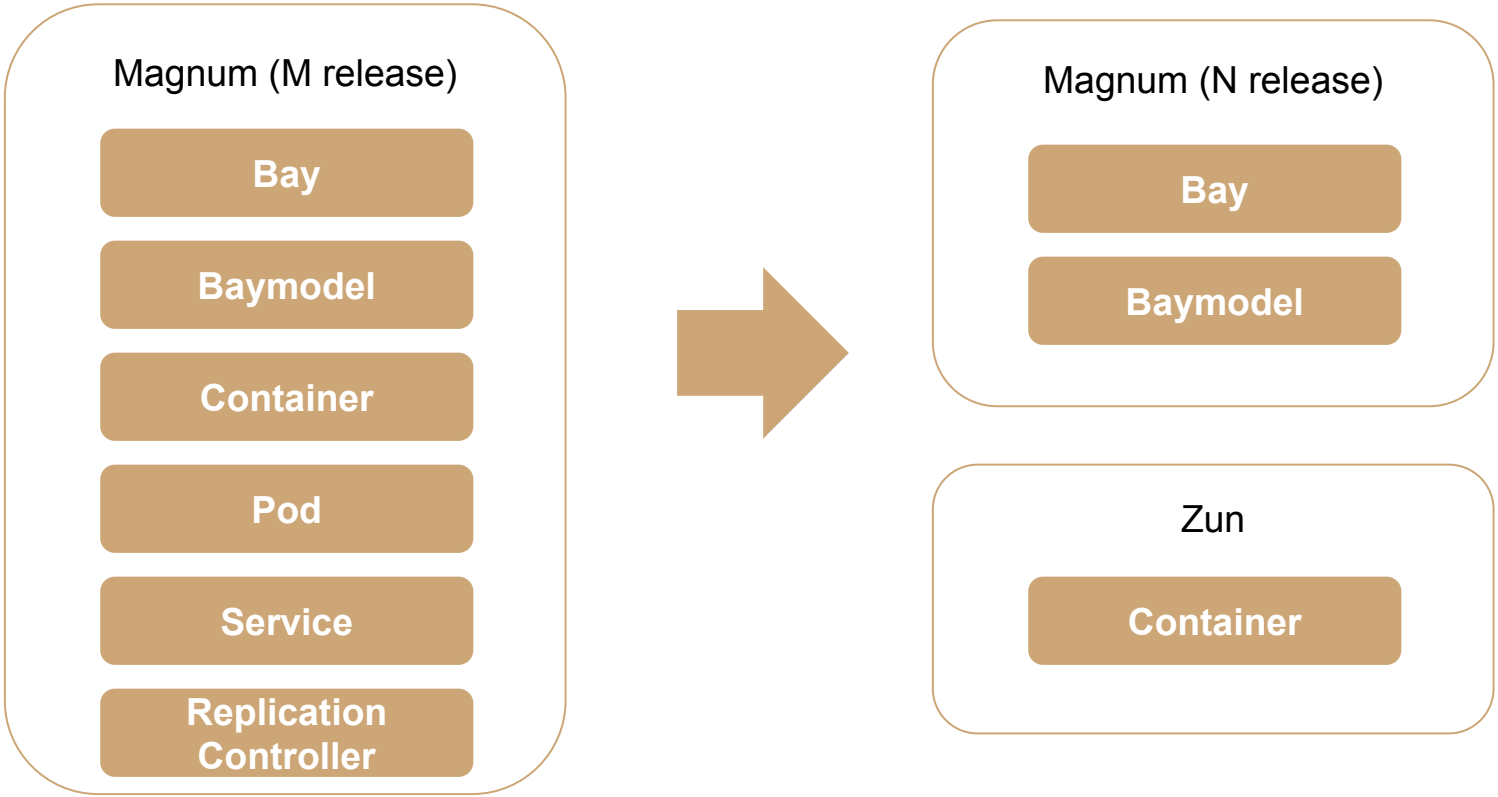
Containers service

Provide a set of services for management of **application containers** in a multi-tenant cloud environment.

Container Infrastructure Management service

Provide a set of services for provisioning, scaling, and managing **Container Orchestration Engines (COEs)**.

Magnum Mission Statement Update

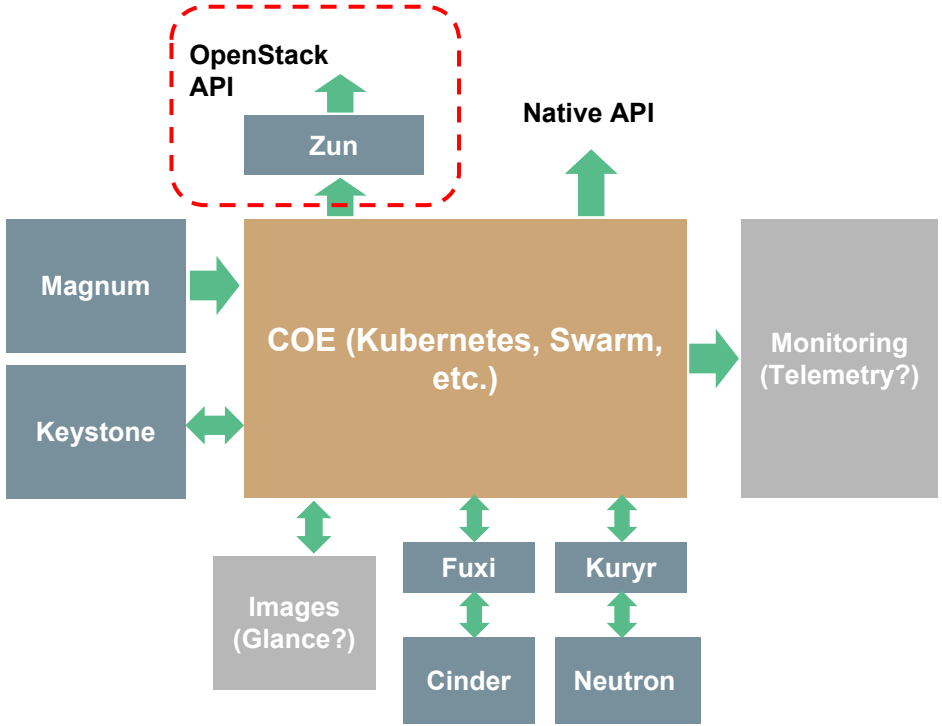


Agenda

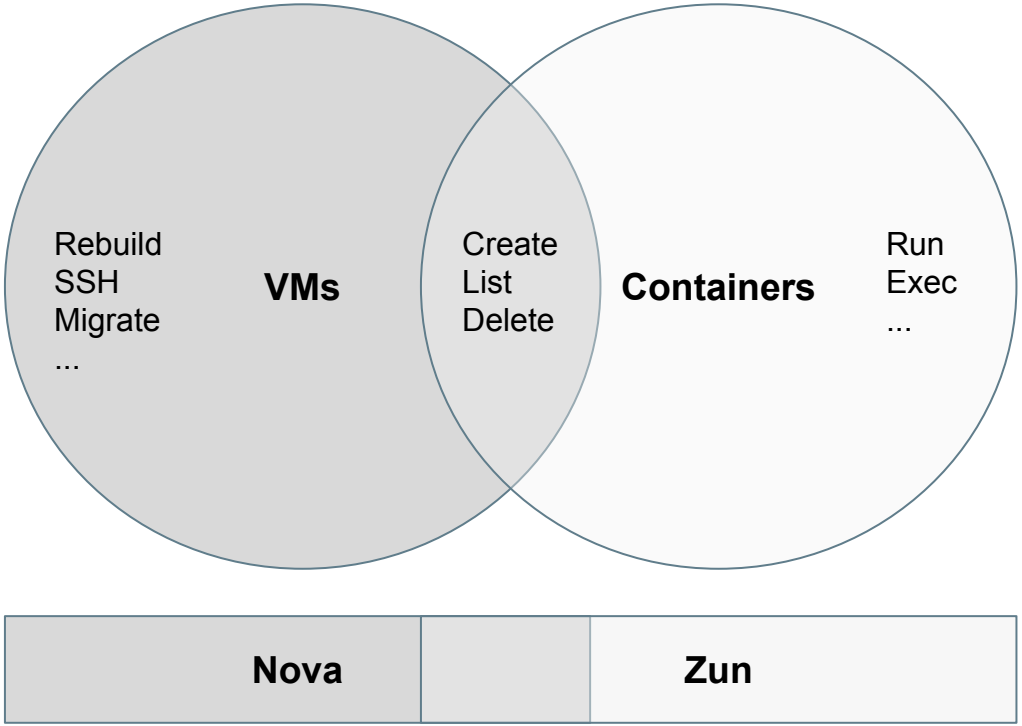
- Containers on OpenStack
- Magnum
- Zun
- Demo

What is Zun?

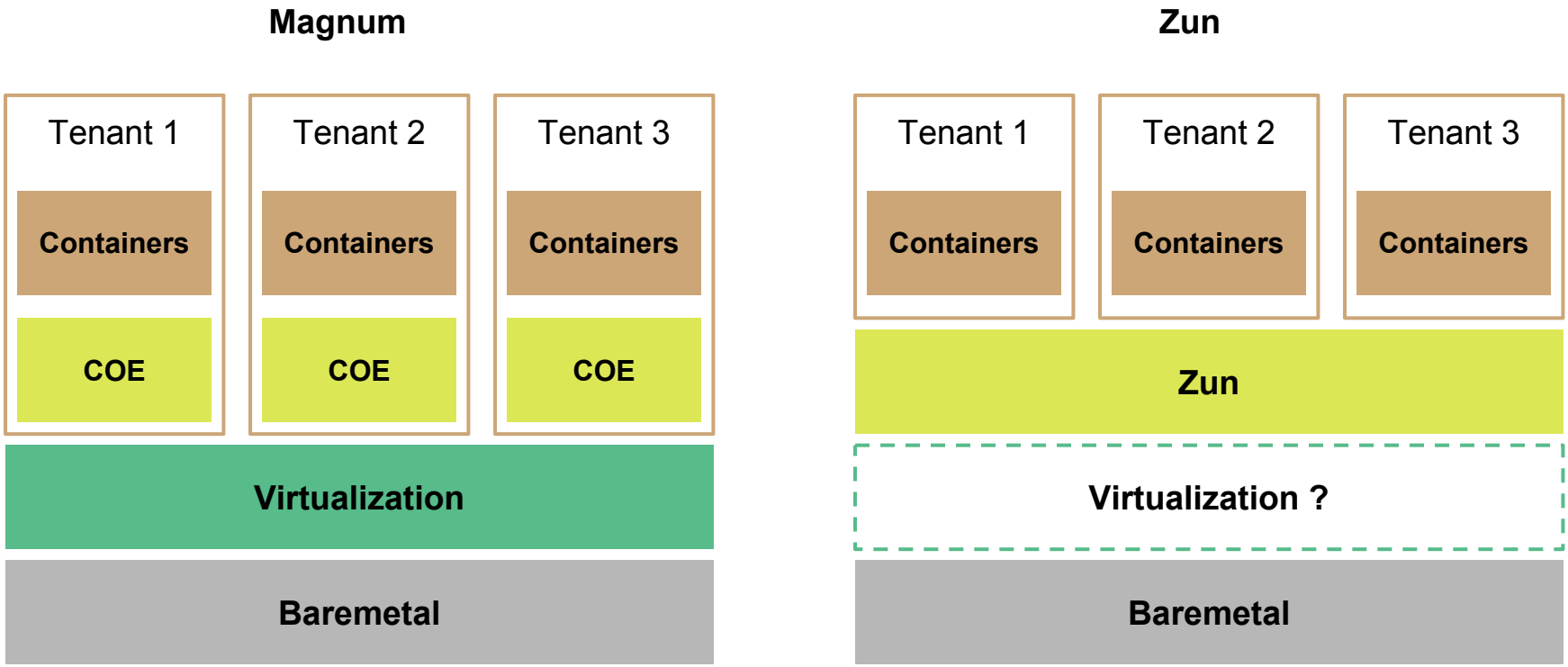
- Abstract container life-cycle management
- Simple API across different container technologies
- Deep Integrate with OpenStack
  - Keystone
  - Nova
  - Neutron
  - Glance
  - Horizon



Why Zun?


Rebuild
SSH
Migrate
...
VMs
Create
List
Delete
Containers
Run
Exec
...
Nova
Zun

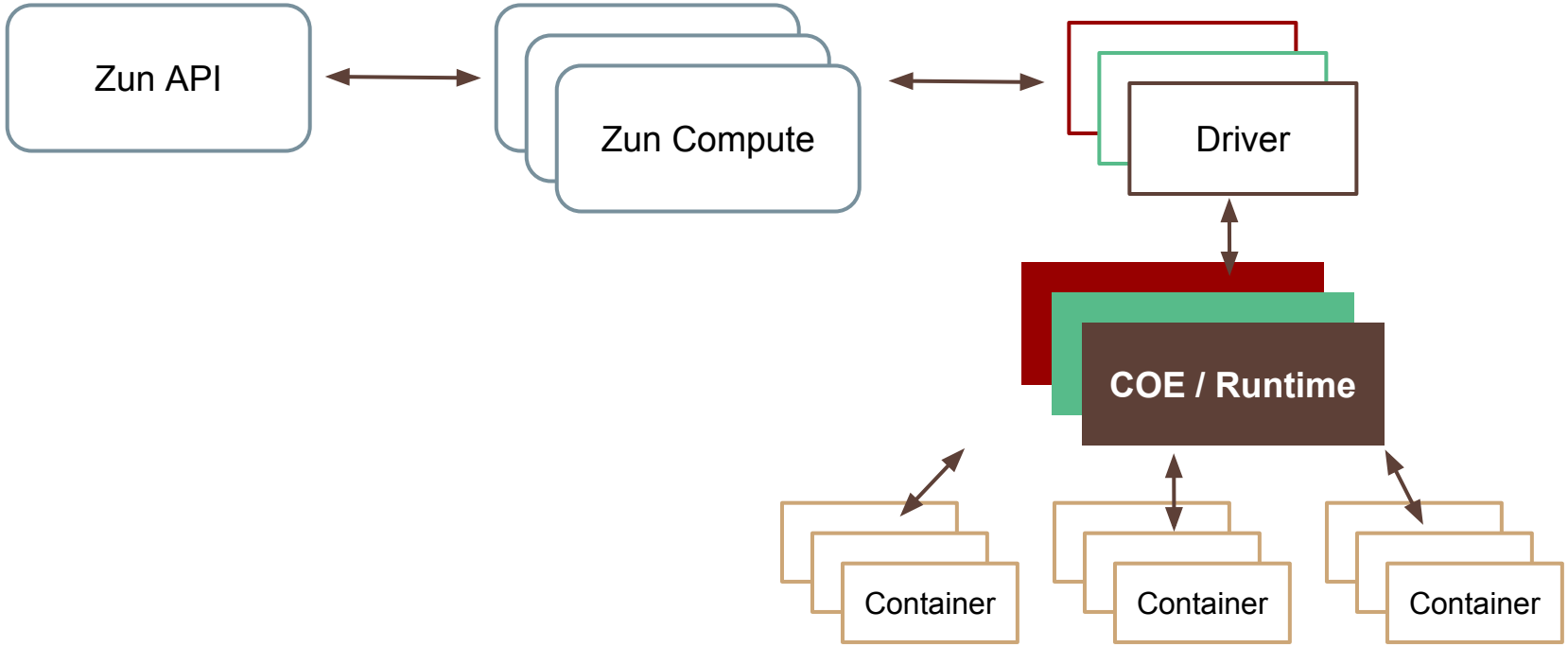
Why Zun?


Magnum
Tenant 1
Containers
COE
Tenant 2
Containers
COE
Tenant 3
Containers
COE
Virtualization
Baremetal
Zun
Tenant 1
Containers
Tenant 2
Containers
Tenant 3
Containers
Zun
Virtualization ?
Baremetal

Why Zun?

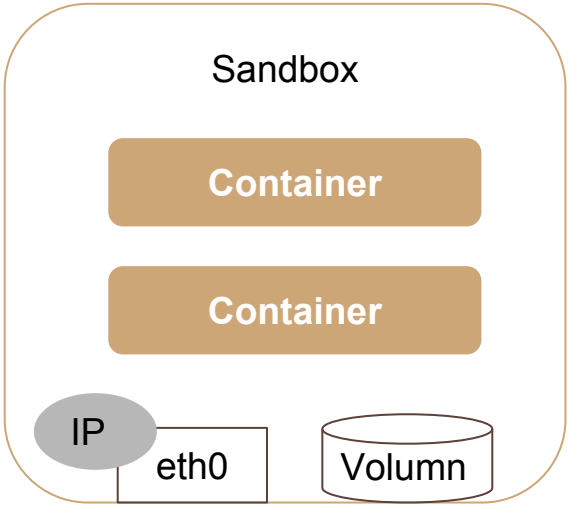
- OpenStack-native APIs
  - Simple
  - Container-oriented
  - Technology-agnostic
- Common infrastructure for VMs, baremetals, and containers
  - Common access control and roles management
  - Common network abstraction layer
  - Common images management
  - Single CLI / UI
  - Single orchestration template for VMs and containers
- No cluster provisioning & management

Architecture



Concepts

- Container
  - A Linux container (i.e. Docker container)
  - Run inside a sandbox
- Sandbox
  - Contain one or multiple containers
  - A placeholder for containers
  - Create an isolated environment
  - Contain network interface(s) and volume(s)
  - Enforce resource constraints (i.e. cpu, memory)



What Exactly is Sandbox?

- Sandbox can be interpreted differently
  - Could be a set of Linux namespaces
  - Could be a VM (i.e. hypervisor-based runtime)
  - Could be a pod (i.e. Kubernetes)
- Our first implementation
  - A container is a Docker container
  - A sandbox is also a Docker container
  - Create a Docker container will automatically create a Docker sandbox
- Potential improvements
  - Allow multiple containers in a single sandbox

Create a Docker Container

1. Create a Docker sandbox

```
$ docker run -d --name mybox kubernetes/pause
```

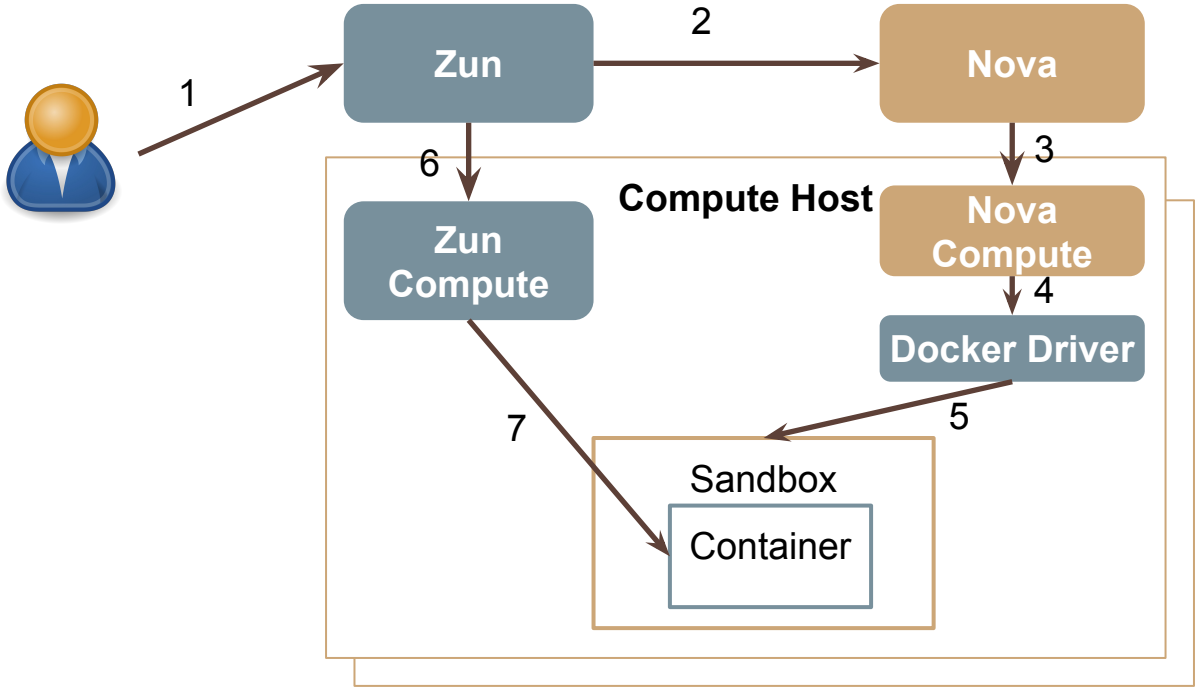
2. Create a container by using the sandbox

```
$ docker run -d --net container:mybox \
      --ipc container:mybox \
      --pid container:mybox \
      --volumes-from mybox \
      ...
```

Why Introduce Sandbox?

- Define a group of containers that are
  - Co-located and Co-scheduled
  - Share network namespace
  - Share volume
  - Share resource limits
- Decouple containers from resources management
  - Containers are managed by Zun
  - Sandbox are managed by **Nova** (with Docker virt driver)

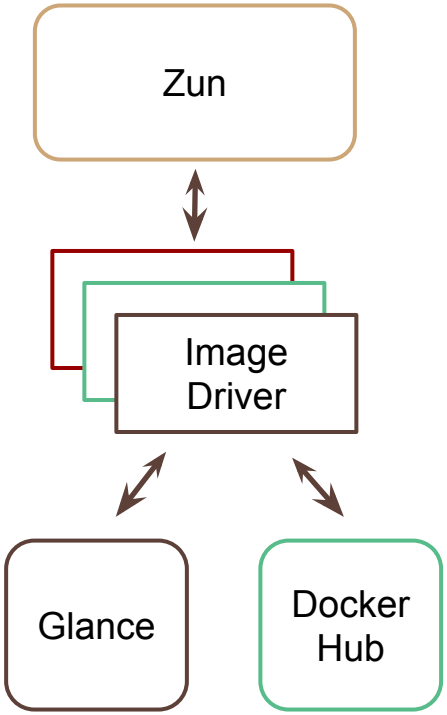
Create a Docker Container



1. End-user requests to create a container
2. Zun requests Nova to create a sandbox instance
3. Nova forwards the request to Nova Compute
4. Nova Compute forwards the request to a Zun-provided virt driver
5. The virt driver create the sandbox
6. Zun requests Zun Compute to create a container
7. A container is created inside the sandbox

Container Image

- Consistent API to manage container images
- Support multiple storage backend
  - Glance (stored as a tar file)
  - Docker Hub
  - Private Docker Registry
- Pluggable design
  - Easy to add support for additional image backend



Agenda

- Containers on OpenStack
- Magnum
- Zun
- Demo

Demo

https://www.youtube.com/watch?v=umcok662jkM