

OPENSTACK DAYS  
**CHINA**

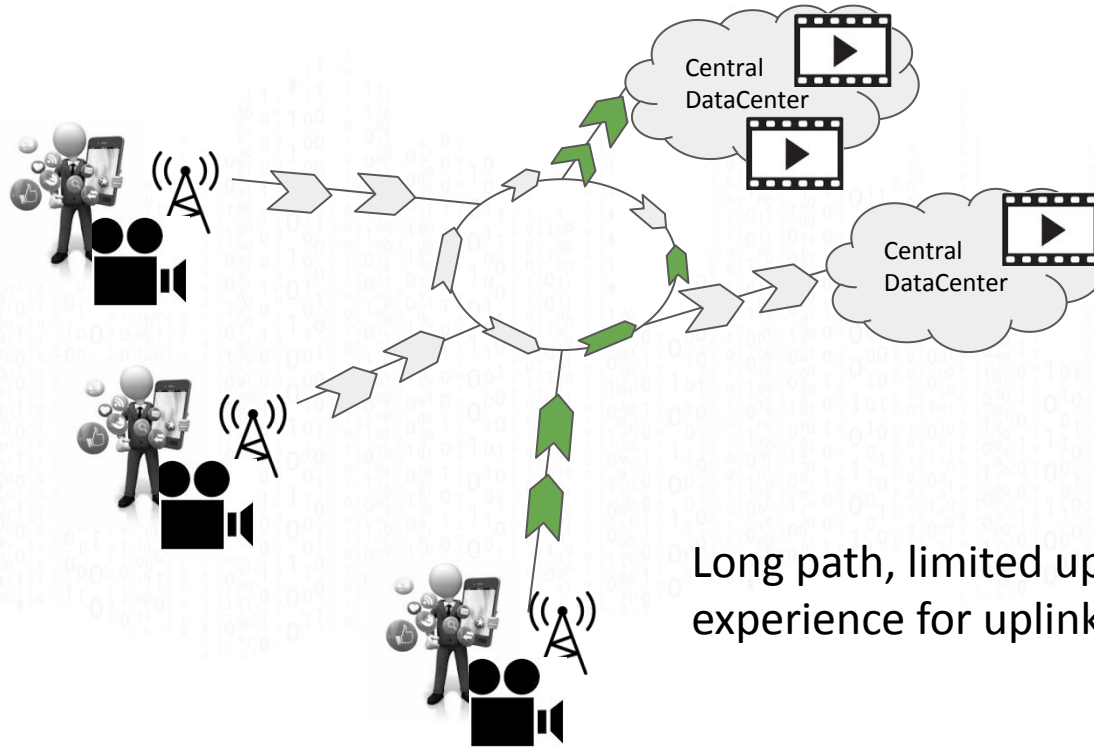
# 分布式DC面临的挑战及解决之道

How we meet the challenges brought by  
Distributed Data Center

Zhiyuan Cai, Yin Ding  
Huawei



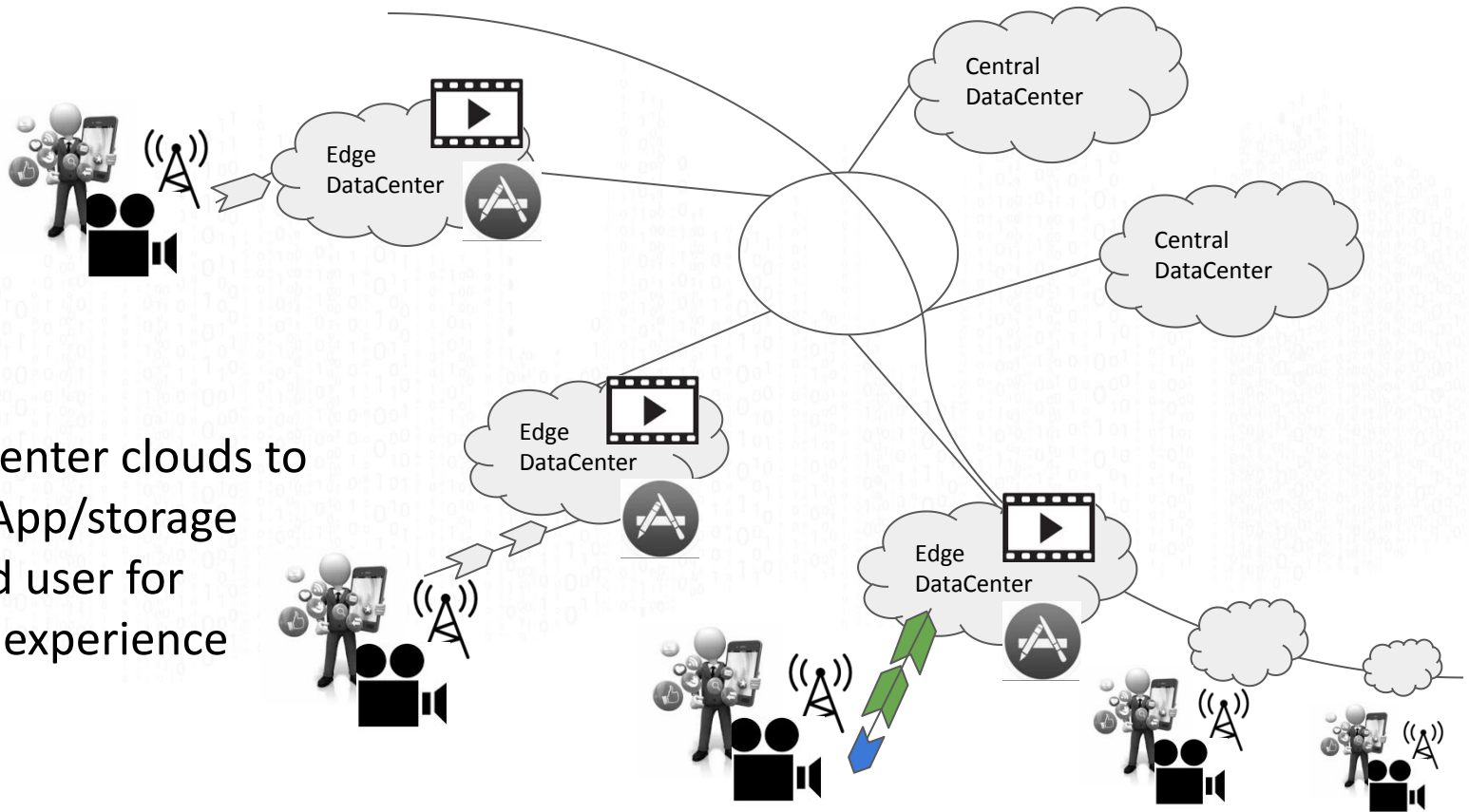
# Original Central DC



Long path, limited uplink bandwidth, bad experience for uplink in centralized cloud



# Distributed DC



Edge data center clouds to place VNF/App/storage close to end user for better user experience



# Challenges

- Resource provision
  - How to schedule an OpenStack instance to create VM, volume
- Networking
  - How to connect VMs in different OpenStack instances
- Scaling
  - How to scale OpenStack cloud

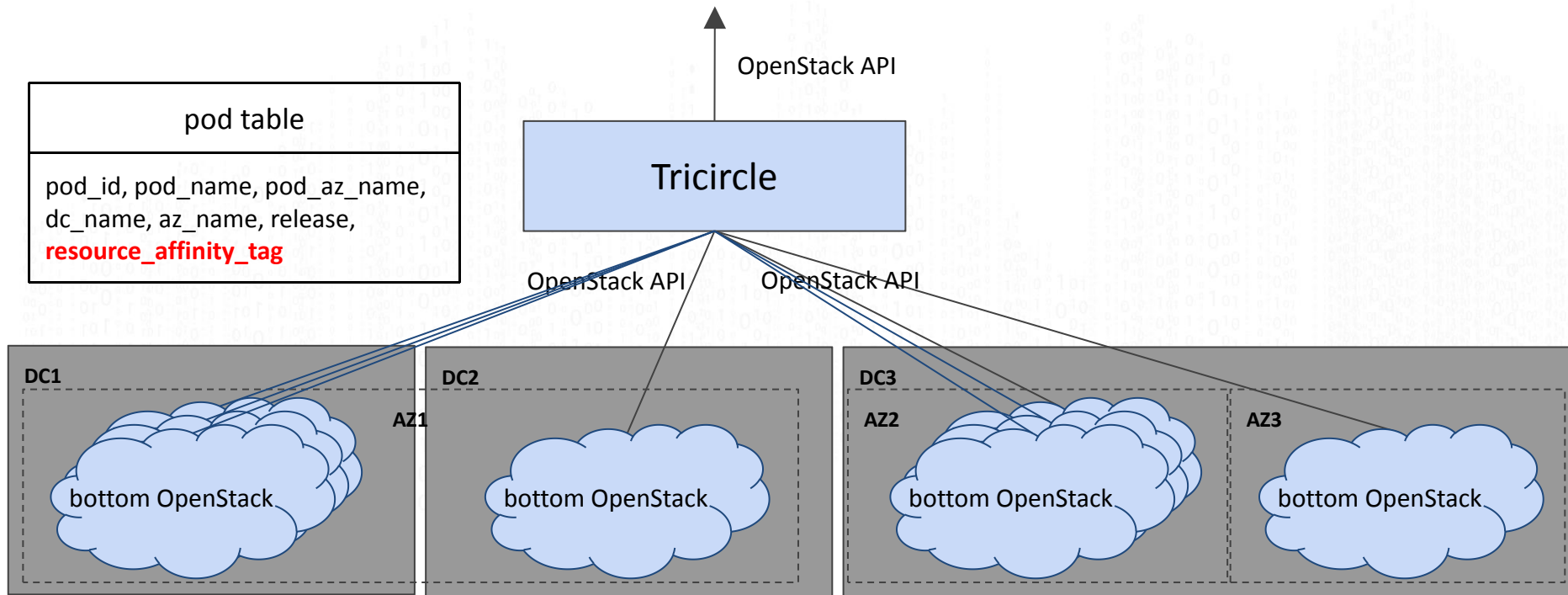


# Meet Tricircle

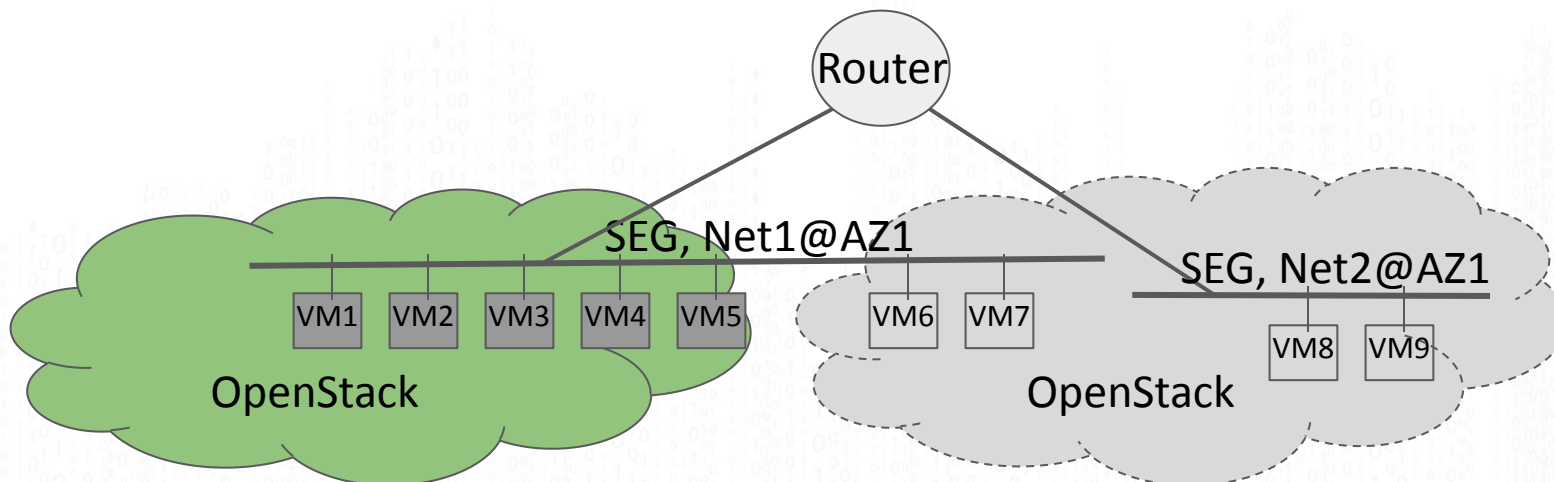
- **Tricircle** provides an **OpenStack API gateway** and **networking automation** to allow multiple OpenStack instances, spanning in one site or multiple sites or in hybrid cloud, to be managed as a single OpenStack cloud.



# Tricircle provides OpenStack federation affinity



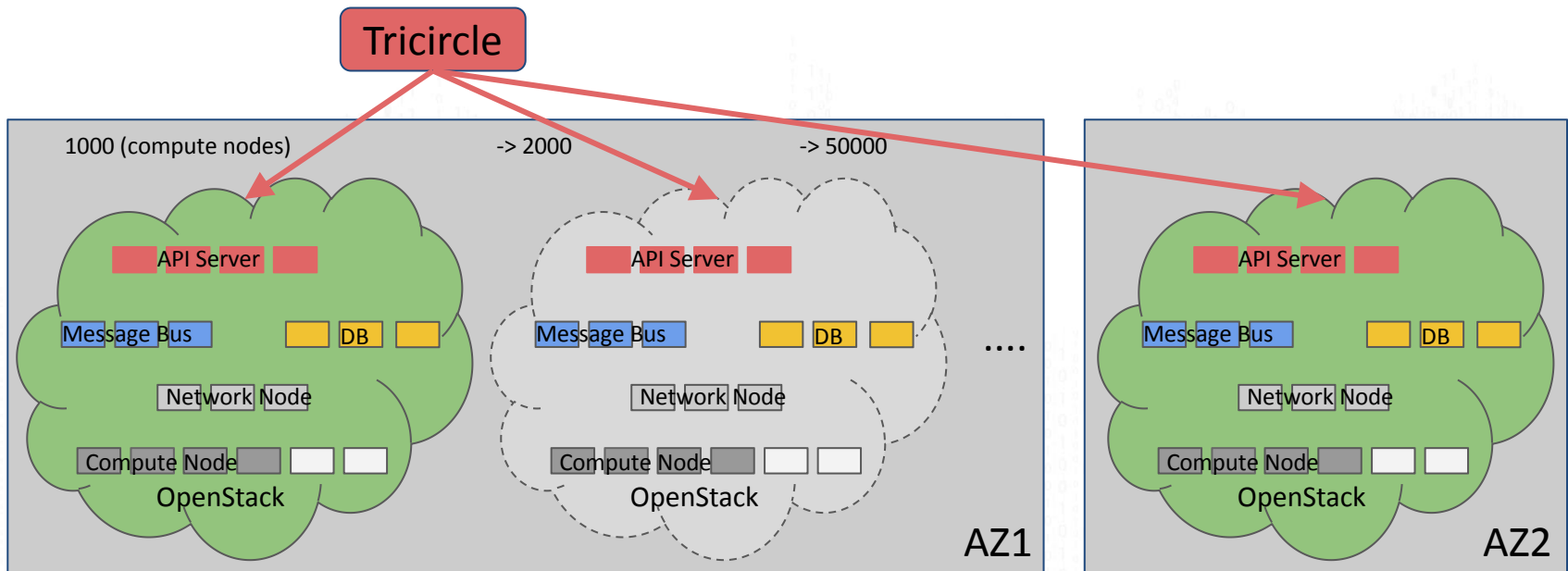
# Tricircle provides cross OpenStack network automation



- One AZ may contain more than one OpenStack
- Networking should be taken good care of to not ruin user's expectation
  - VM1 and VM6 should be able to communicate with each other since they are both in Net1
  - VM1 and VM8 should be able to communicate with each other via Router



# Tricircle provides modularized scaling

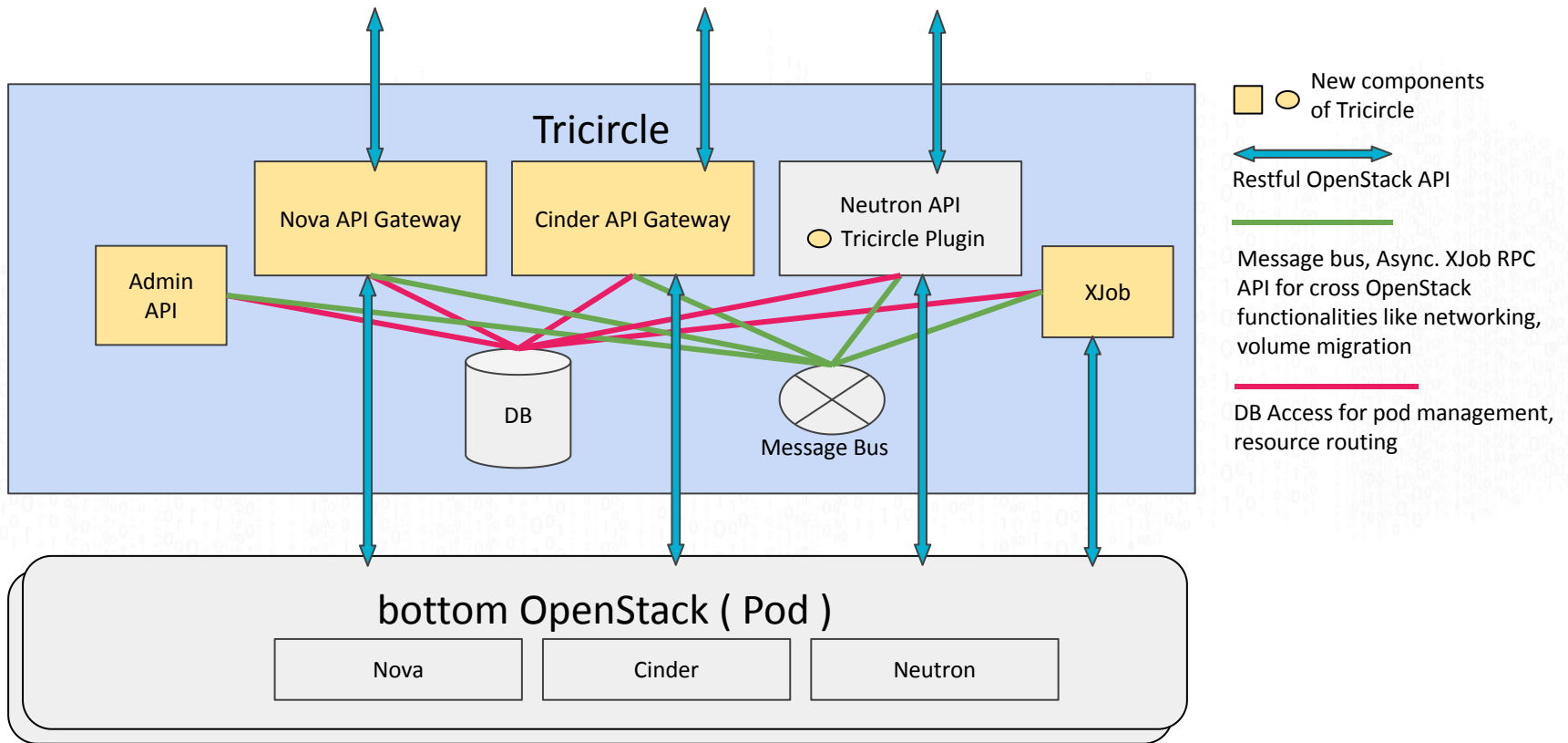


- Capability expansion is headache
- Utilize tested and verified building block for capacity expansion
- Treat one OpenStack as one building block
- Tricircle provides API to create a new pod(one OpenStack instance)

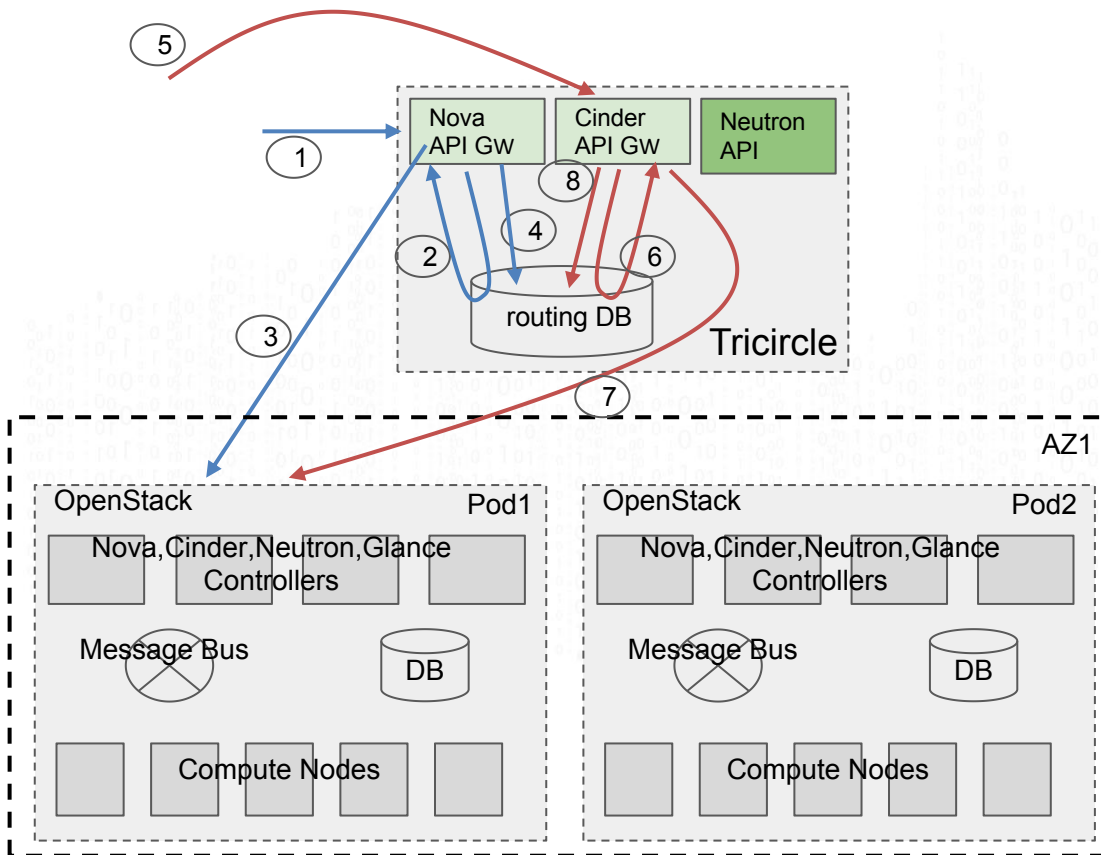




# Architecture of Tricircle



# Example – VM & volume collocation



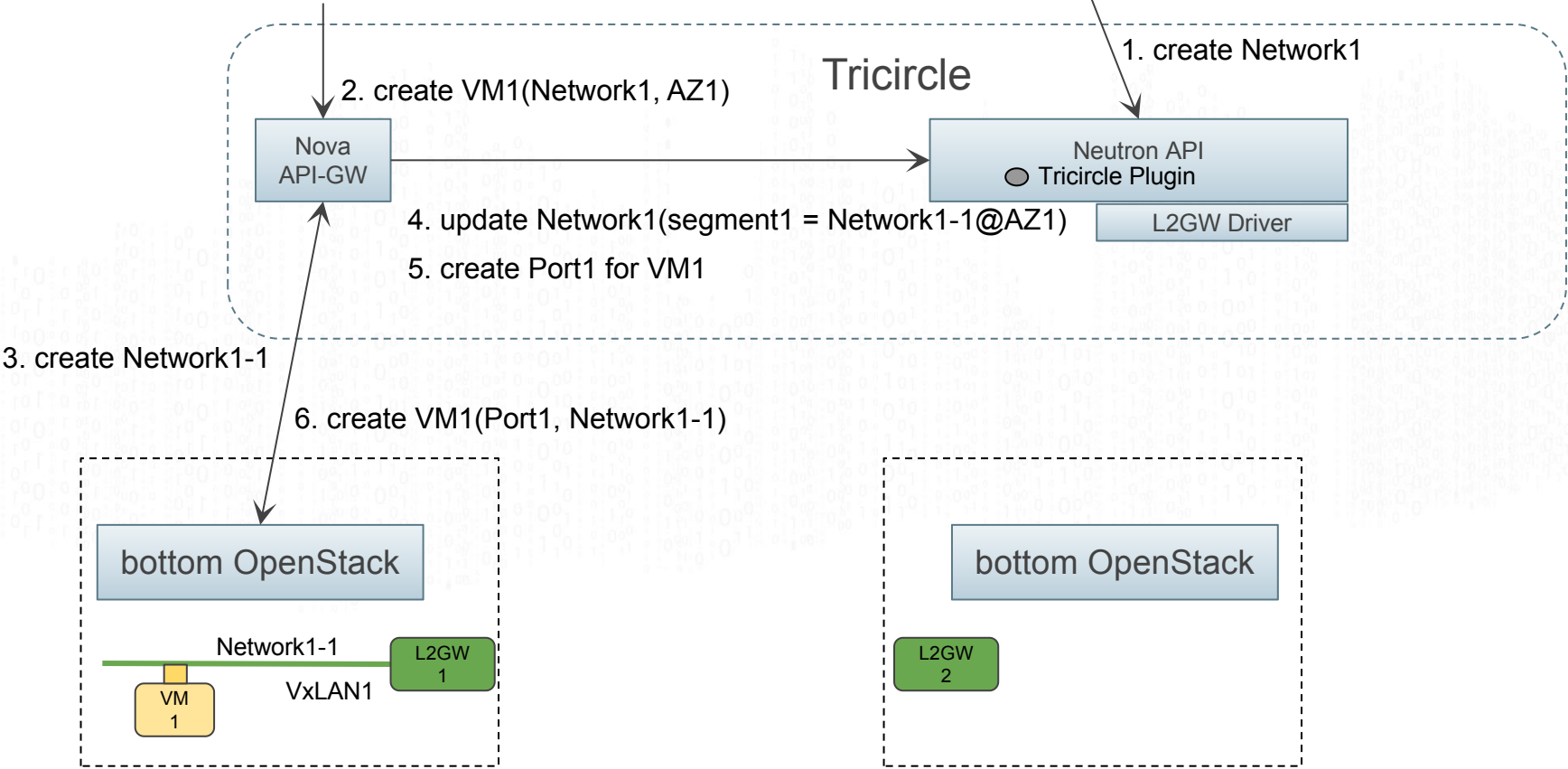
- (1) User sends VM create request specifying availability zone parameter as AZ1
- (2) AZ1 has two pods, Nova API gateway schedules one pod(Pod1) and bind this user to that pod in AZ1
- (3) Nova API gateway sends request to Pod1
- (4) Nova API gateway caches routing information
- (5) User sends volume create request specifying availability zone parameter as AZ1
- (6) Query DB to get user-pod binding information
- (7) Cinder API gateway sends request to Pod1
- (8) Cinder API gateway caches routing information

resource routing table
Id, top_id, bottom_id, pod_id, project_id, resource_type

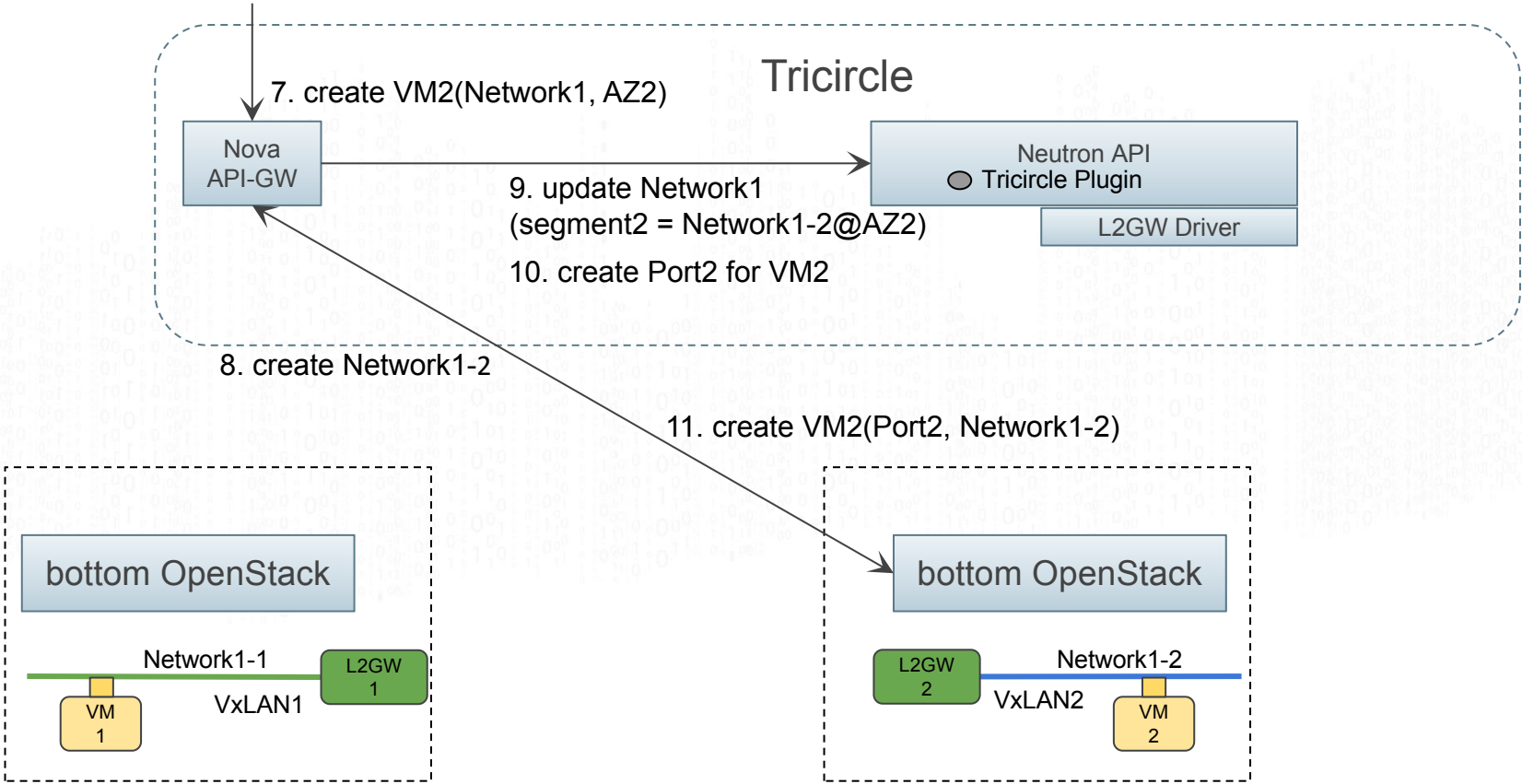
pod binding table
id, project_id, pod_id



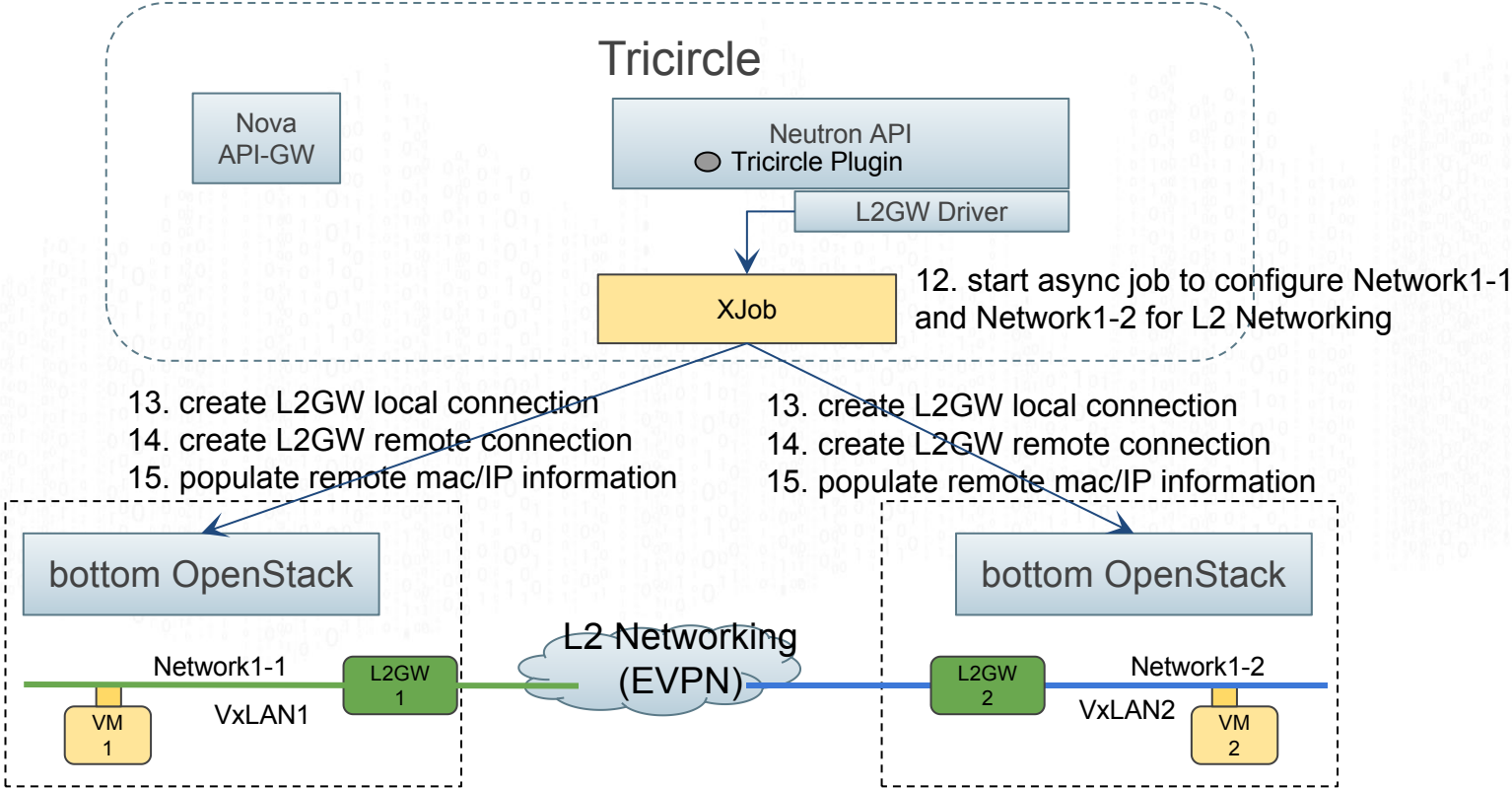
# Example - network automation



# Example - network automation

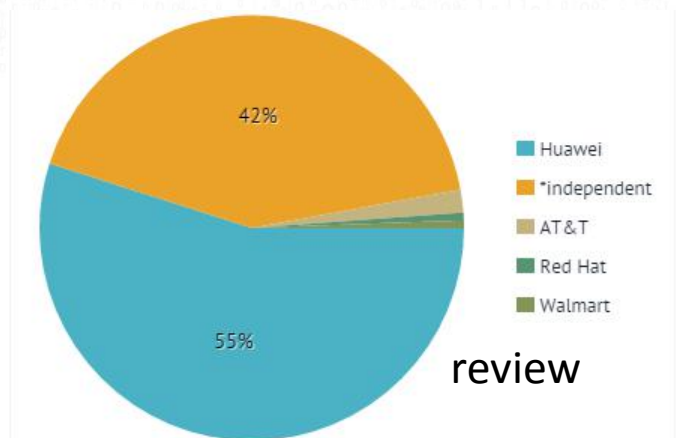


# Example - network automation

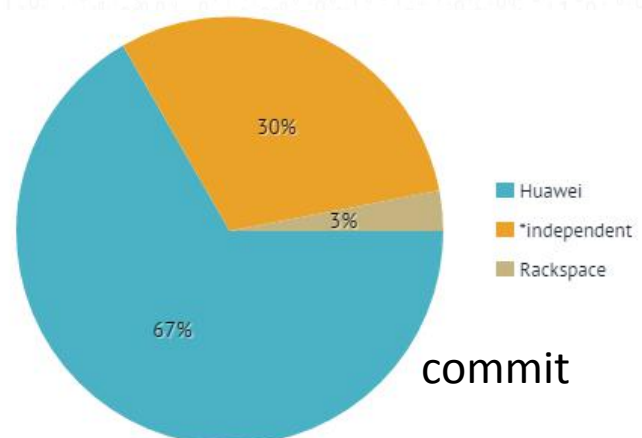


# Welcome to join us

- Wiki of Tricircle:
  - <https://wiki.openstack.org/wiki/Tricircle>
- Play and contribute:
  - <https://github.com/openstack/tricircle>
- Design document:
  - <https://docs.google.com/document/d/18kZZ1snMOCD9IQvUKI5NVDzSASpw-QKj7I2zNqMEd3g>



from <http://stackalytics.com/>



OPENSTACK DAYS CHINA



# Cloud Native OpenStack

- Cloud Native applications:
  - Applications or Services that are *container-packaged*, dynamically scheduled and *micro-services* oriented
- Container on OpenStack Challenges
  - Network
  - Storage



# Popular Cloud Providers

	Amazon	Google	Microsoft	Rackspace	Redhat
VM + Container Network	All VPC Network	Overlay	Overlay	Overlay	Overlay
Container Storage	Host Mount Volumns/ EBS volume	GCE provide Persistent volume		hostPath	Ceph, Gluster, NFS (iscsi)
Orchestration of (Mix use of PM/VM/Container Cluster)	No	No	No	No	No



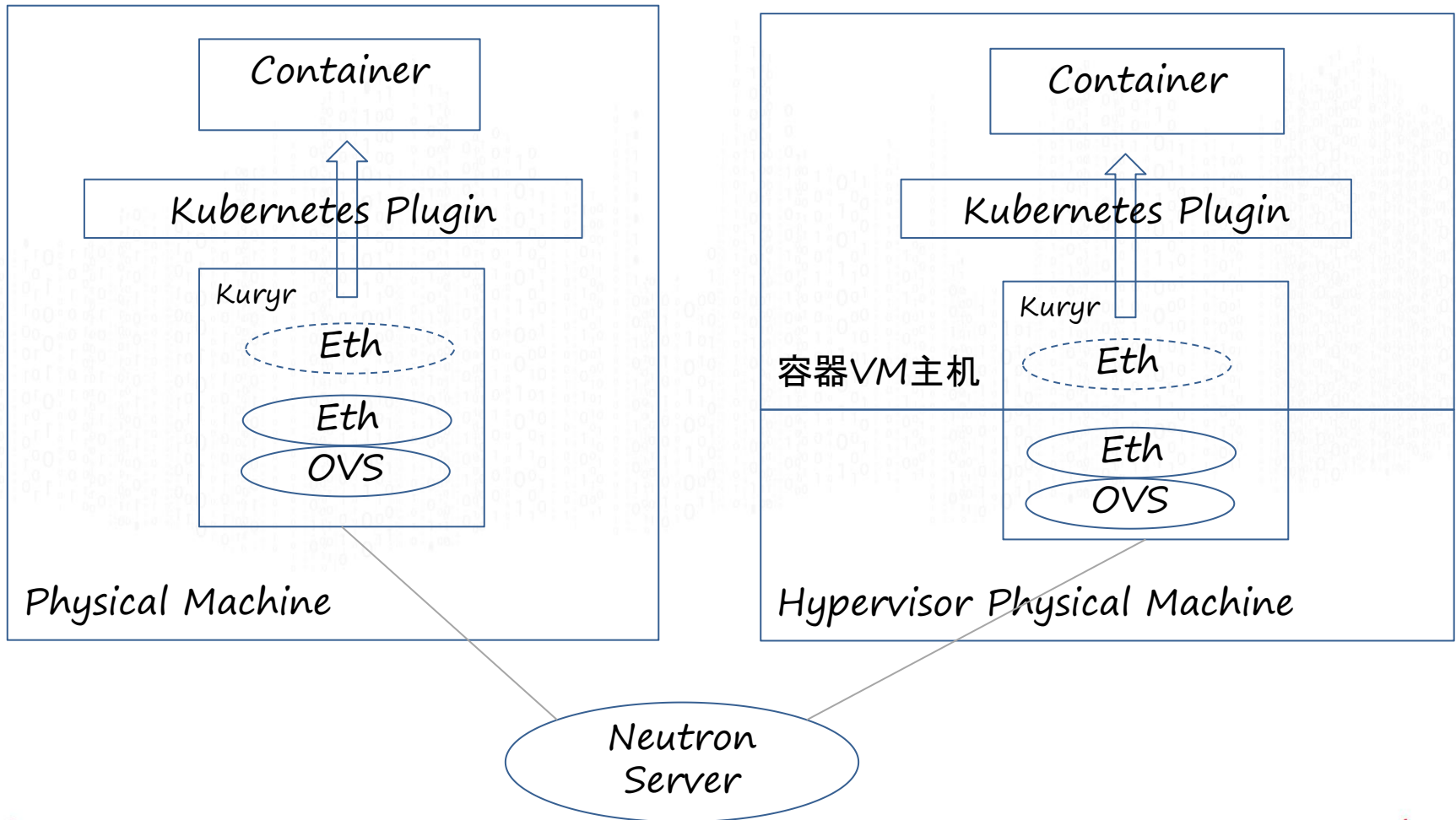


# Our Approach

- Kuryr:
  - One Network Management:  
VM + Container
- Fuxi:
  - Connect containers with OpenStack Cinder

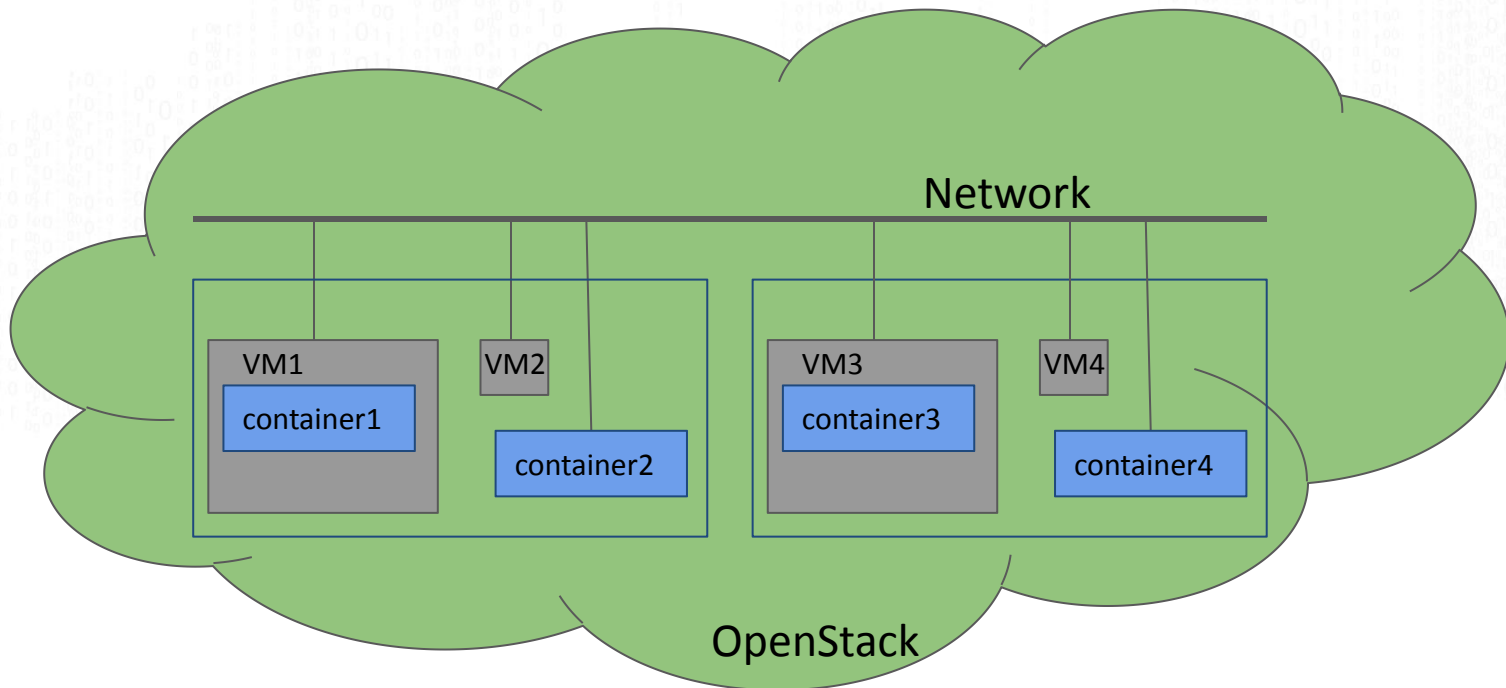


# Networking with Kuryr

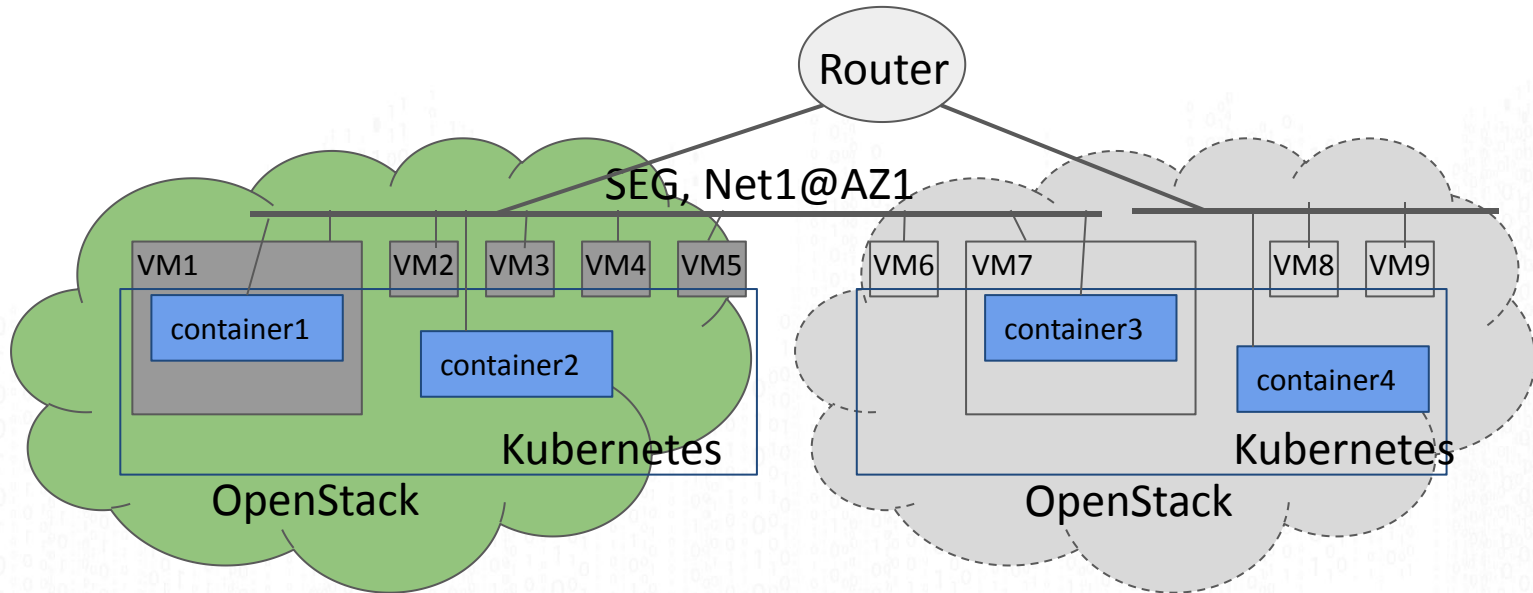


# Network

- create VM, Container on the same Network no matter Containers in VM or Bare Metal



# With TriCircle Network Automation

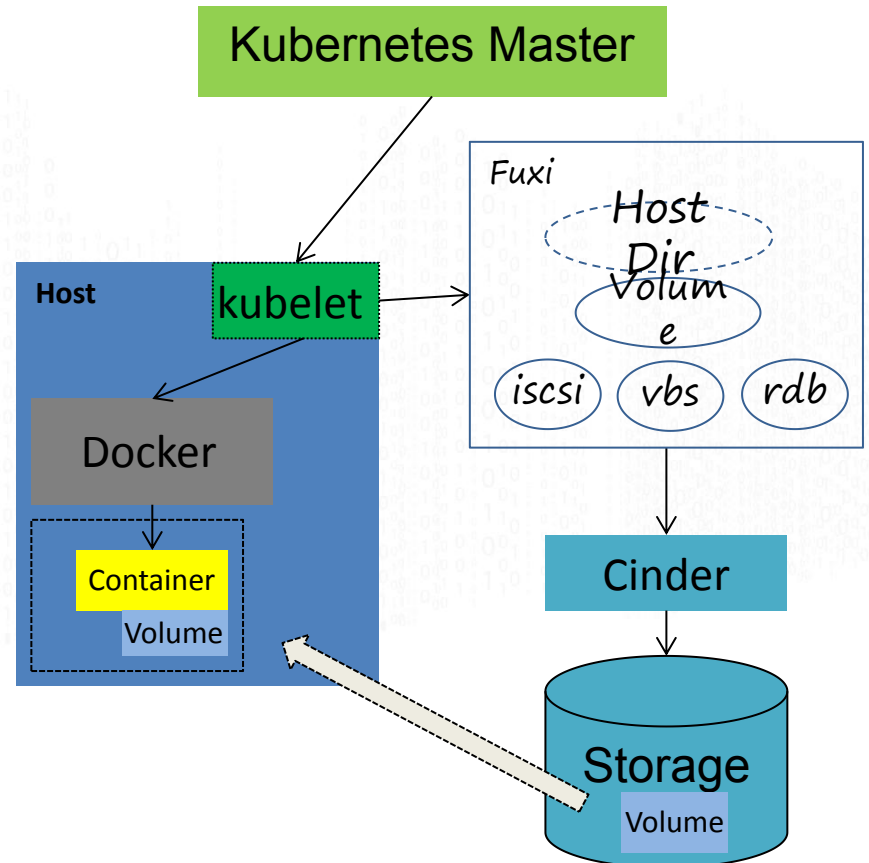


- Networking has been taken good care of
  - Container1, container2 and container3 should be able to communicate with each other since they are both in Net1
  - Container3 and Container4 should be able to communicate with each other via Router



# Storage

- In OpenStack, a Kubernetes volume
  - uses hostPath volume mounts a file or directory from the host node's file system.
- With Fuxi
  - Simply mounts Cinder volume to the host for kubelet



# What we achieve

- Bring Container as the First-Class resource in OpenStack
- One network management
- Stateful Container with persistent storage in OpenStack



# Q&A



Thanks

