Multiple Networks in Kubernetes

Haibin Michael Xie

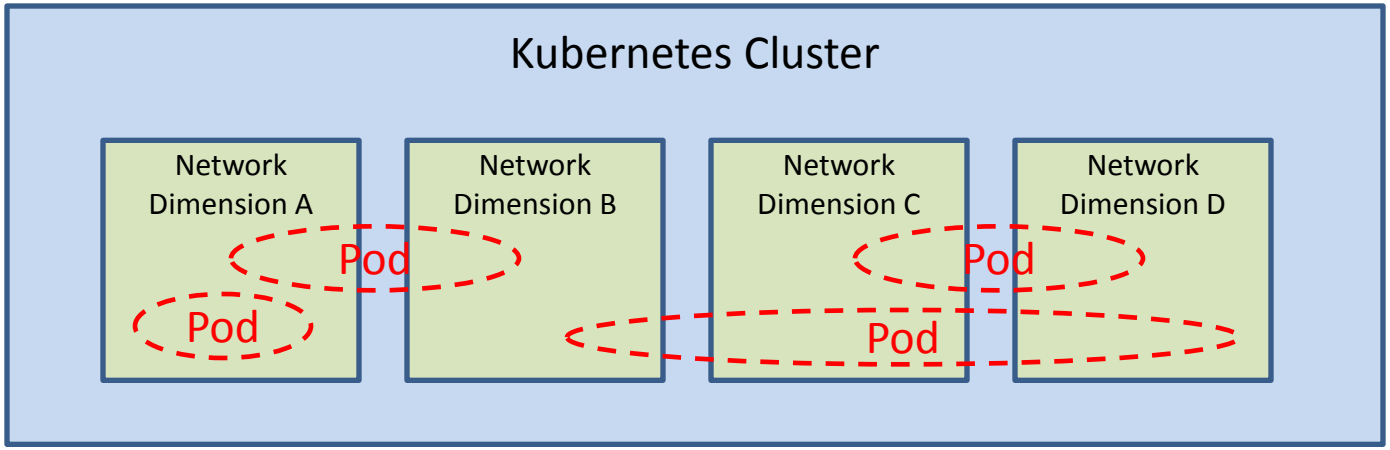
Haibin.michael.xie@huawei.com

Agenda

- Definition of multiple networks
- Why we need multiple networks
- Multiple networks in Kubernetes (not public yet)
- Open source plan
- Q&A

Definition of Multiple Networks

- Multiple physical networks
- Multiple logical networks
  - Multiple network interfaces per container
  - Multiple network solutions/plug-ins
  - Multiple network dimensions

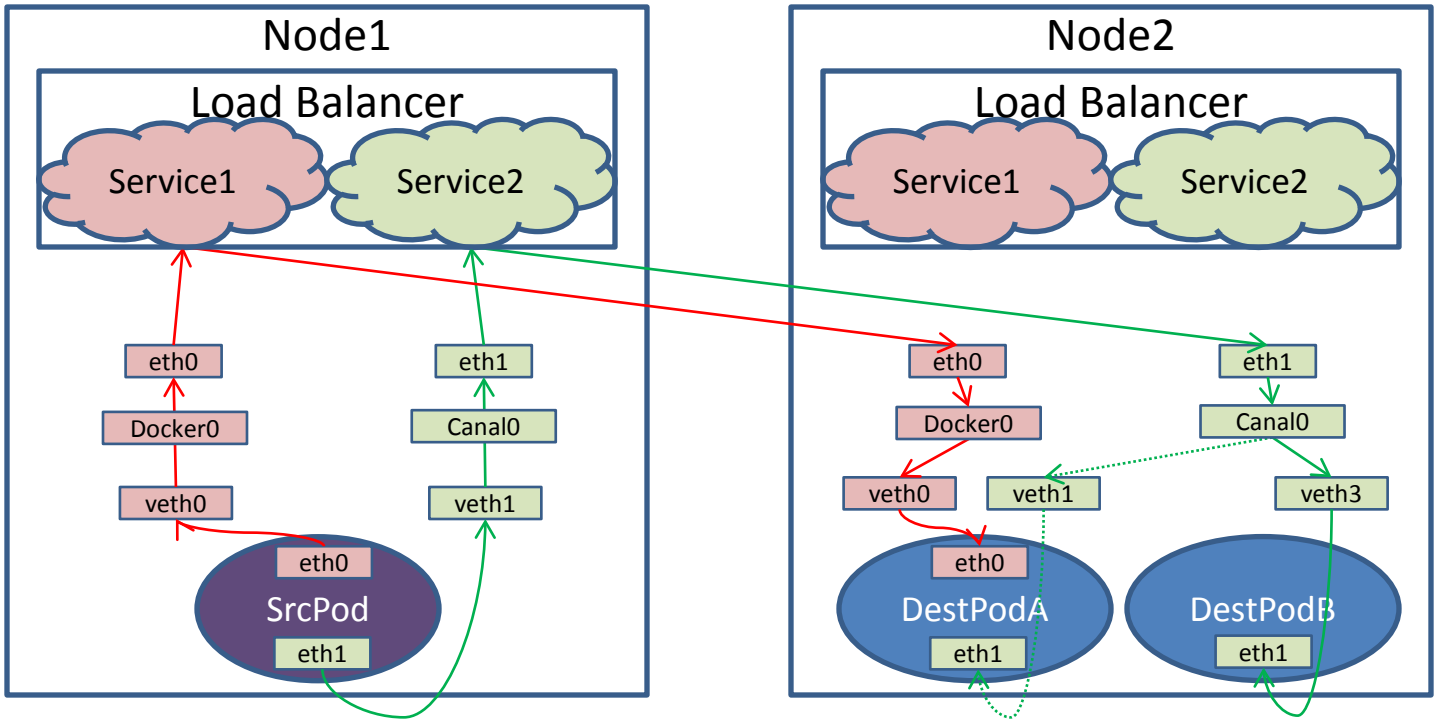


Note: One network dimension uses one network plug-in and has unique subnet

Why we need multiple networks

- Network dimension abstraction
- Network isolation
- Use multiple network solutions
- User scenarios:
  - NFV: virtual network functions need access to control plane, data plane and monitor plane
  - IPV6 co-existing with IPV4
  - Applications have both internal and public access
  - Servers that want to isolate traffic from multiple clients
  - Two applications on same host using different host network interface

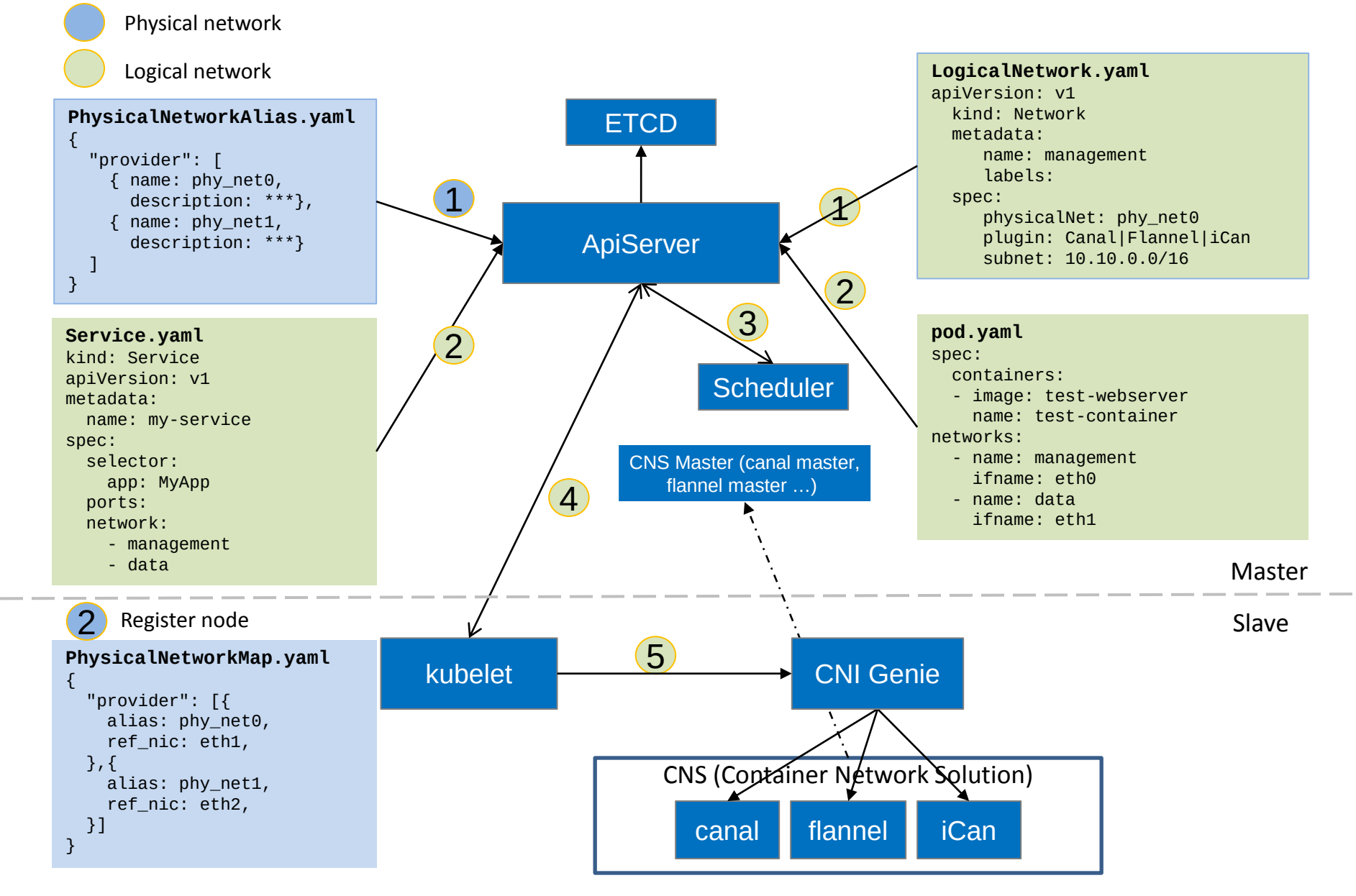
Multiple Network Topology



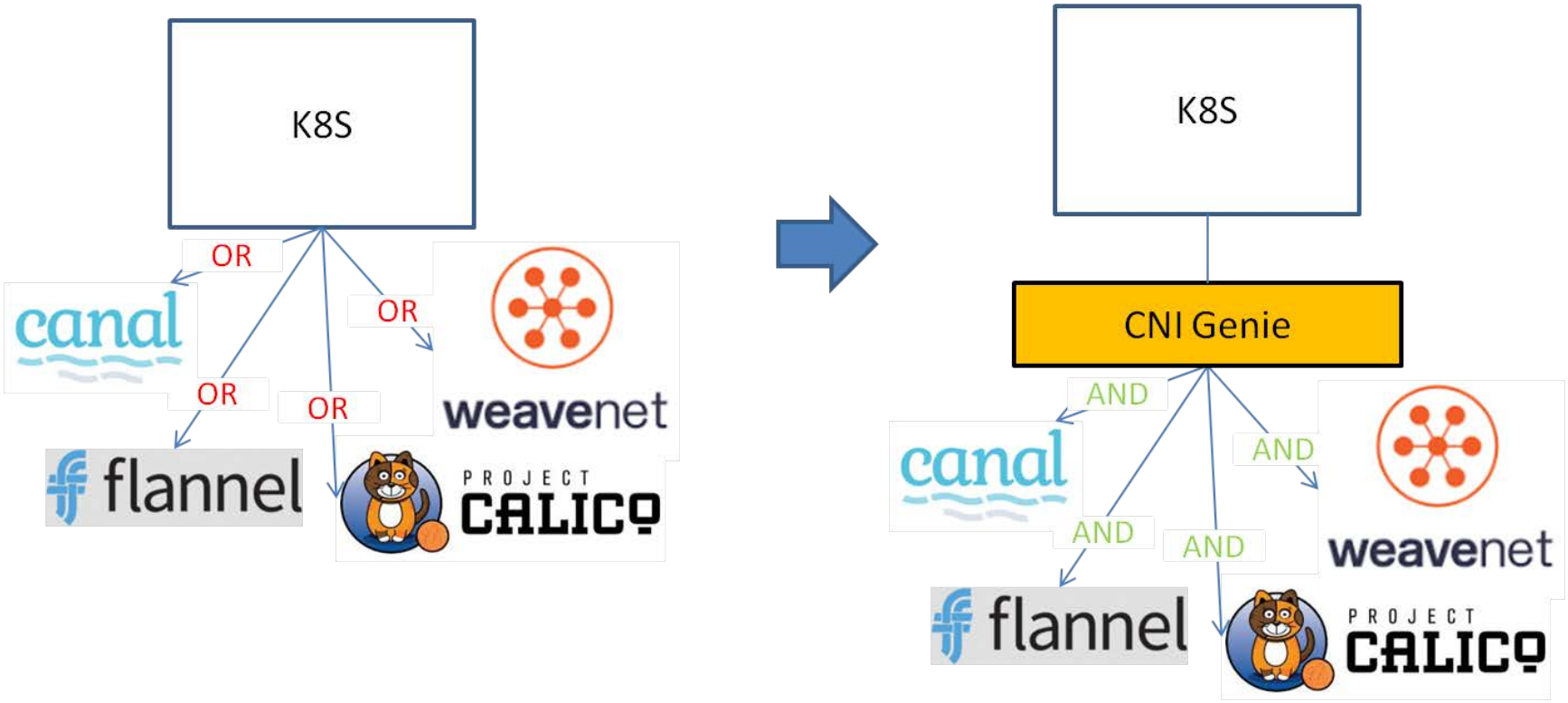
Changes to Kubernetes

- New physical network object
- New logical network object
- Pod object with multiple networks and IPs
- Service with multiple network dimensions
- Network based scheduling
- CNI Genie (network plug-in manager)

Kubernetes Multiple Networks Workflow



CNI-Genie


K8S
OR
OR
OR
OR
canal
flannel
weavenet
PROJECT
CALICO
K8S
CNI Genie
AND
AND
AND
AND
canal
flannel
weavenet
PROJECT
CALICO

Kubectl Usage

List of slave nodes

```
SHA1000136564:/home/test # /opt/paas/kubernetes/kubectl --client-certificate=tls.crt --client-key=tls.key --certificate-authority=ca.crt -s
https://100.106.74.140:5443 get nodes -n multinet
NAME         STATUS    AGE
multinet-1   Ready     5d
multinet-2   Ready     6d
multinet-3   Ready     6d
```

Node description

```
SHA1000136564:/home/test # /opt/paas/kubernetes/kubectl --client-certificate=tls.crt --client-key=tls.key --certificate-authority=ca.crt -s
https://100.106.74.140:5443 get node multinet-1 -n multinet -o yaml
apiVersion: v1
kind: Node
metadata:
  annotations:
    network.alpha.kubernetes.io/mappings: networkmapping1
    volumes.kubernetes.io/controller-managed-attach-detach: "true"
  creationTimestamp: 2017-06-08T06:18:15Z
  enable: true
  labels:
    beta.kubernetes.io/arch: amd64
    beta.kubernetes.io/os: linux
    kubernetes.io/hostname: multinet-1
    network.alpha.kubernetes.io/phynet1: eth1
    os.architecture: amd64
```

List of Physical Networks

```
SHA1000136564:/home/test # /opt/paas/kubernetes/kubectl --client-certificate=tls.crt --client-key=tls.key --certificate-authority=ca.crt -s
https://100.106.74.140:5443 get pn
NAME      TYPE         PVID      AGE
phynet1   overlay_l2   1         6d
```

List of Logical Networks

```
SHA1000136564:/home/test # /opt/paas/kubernetes/kubectl --client-certificate=tls.crt --client-key=tls.key --certificate-authority=ca.crt -s
https://100.106.74.140:5443 get net
NAME      PHYNET    TYPE         SUBNET          AGE
net1      phynet1   overlay_l2   122.20.0.0/16   6d
```

Kubectl Usage

Deploy pod

```
SHA1000136564:/home/test # cat pod-nginx.yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    name: nginx
  annotations:
    network.alpha.kubernetes.io/network: '[{"name":"net1", "interface":"eth1"}]'
  namespace: multinet
spec:
  containers:
    image: 100.106.74.140:20202/canal/nginx:1.7.8
    imagePullPolicy: IfNotPresent
    name: nginx
    ports:
      - containerPort: 443
  restartPolicy: Always
  securityContext: {}
  serviceAccount: default
  serviceAccountName: default
```

```
SHA1000136564:/home/test # /opt/paas/kubernetes/kubectl --client-certificate=tls.crt --client-key=tls.key --certificate-authority=ca.crt -s
https://100.106.74.140:5443 create -f pod-nginx.yaml
pod "nginx" created
```

Query pods

```
SHA1000136564:/home/test # /opt/paas/kubernetes/kubectl --client-certificate=tls.crt --client-key=tls.key --certificate-authority=ca.crt -s
https://100.106.74.140:5443 get pods -n multinet -o wide
NAME                    READY     STATUS    RESTARTS   AGE       IP                      NODE
fuxi-72bds              1/1       Running   0          6d        100.106.122.158         multinet-3
fuxi-qltmm              1/1       Running   0          6d        100.106.122.215         multinet-2
fuxi-x08dj              1/1       Running   0          5d        100.106.75.232          multinet-1
nginx                   1/1       Running   0          4m        172.16.0.99,122.20.0.99     multinet-1
```

Open Source Plan

- CNI Genie is open sourced
  - ✓ https://github.com/Huawei-PaaS/CNI-Genie
  - ✓ Pods annotated to use multiple network plug-ins
- Working on open source all of the work phase by phase…
  - ✓ Multiple plug-ins per node (Done in CNI-Genie)
  - ✓ Multiple IPs per pod (Done in CNI-Genie)
  - ➢ Introduce logical network
  - ➢ Multiple networks for Kubernetes services
  - ➢ Multiple physical networks per node
  - ➢ Network based scheduling

Thank you

Haibin Michael Xie
Haibin.michael.xie@huawei.com