
KubeCon
CloudNativeCon
Europe 2025
LONDON BUS

KubeCon | CloudNativeCon
Europe 2025

Mind Your Pod's Business: Network Isolation Workshop

Surya Seetharaman, Red Hat
Miguel Duarte Barroso, Red Hat
Keith Burdis, Goldman Sachs

Surya Seetharaman
Principal Software Engineer @ Red Hat
SIG-Network contributor
OVN-Kubernetes Maintainer
@tssurya

Miguel Duarte Barroso
Principal Software Engineer @ Red Hat
OVN-Kubernetes contributor
KubeVirt contributor
@maiqueb

Keith Burdis
Executive Director @ Goldman Sachs
Kubernetes Administrator
Tech Lead

KubeCon | CloudNativeCon
Europe 2025

Setup for the Workshop: Prerequisites

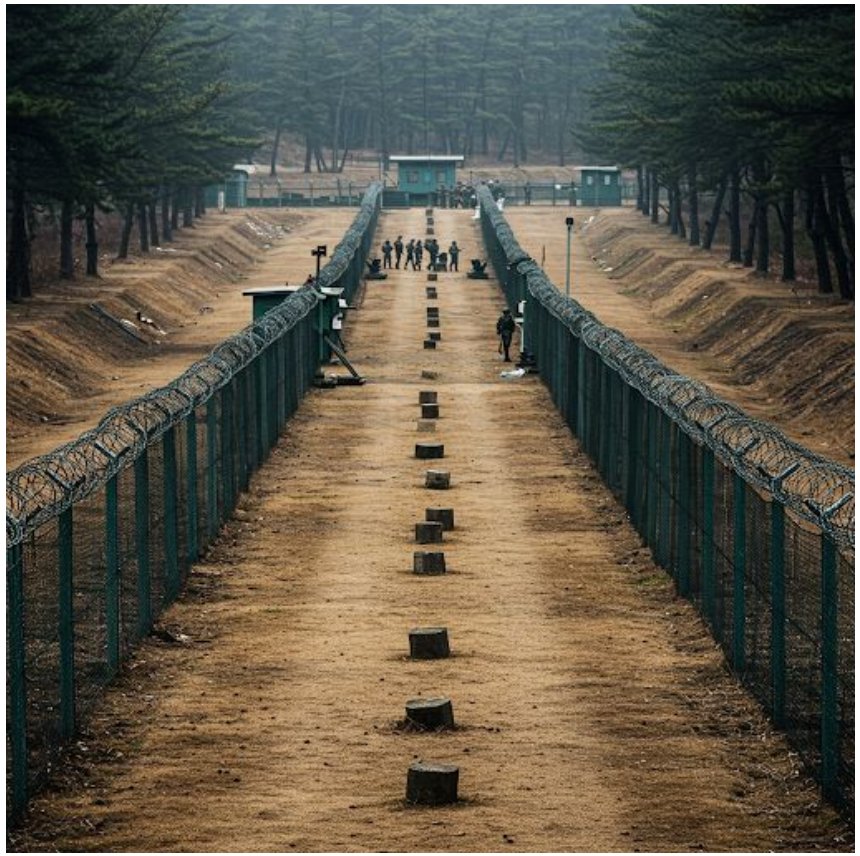
Please Ensure your VM Environment for the workshop is ready!

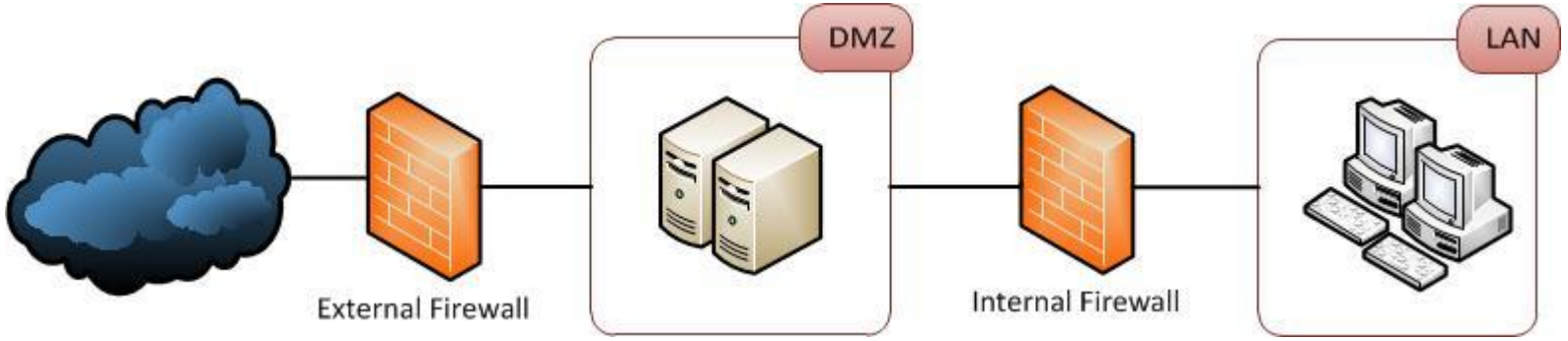
https://github.com/tssurya/kubecon-eu-2025-london-udn-workshop/tree/main?tab=readme-ov-file#kubecon-eu-2025-london-udn-workshop

Tools in your VM:

- git
- wget
- curl
- jq
- openssl
- golang
- docker - if you're using kind on your laptop, you can rely on podman
- python3-pip
- kind and kind-load-balancer
- kubectl
- virtctl

Introduction

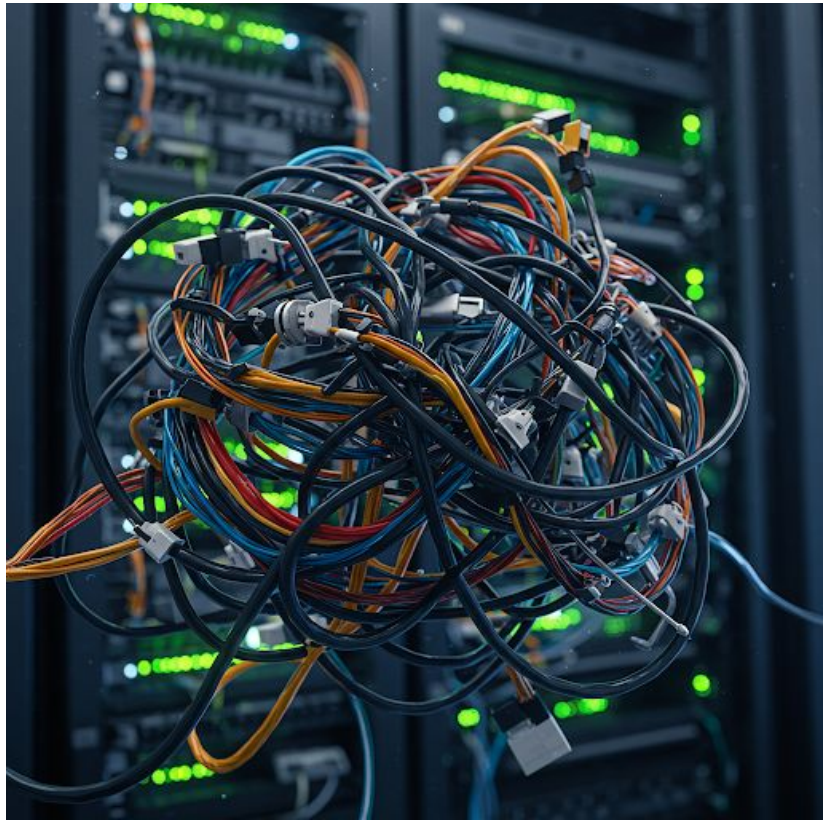


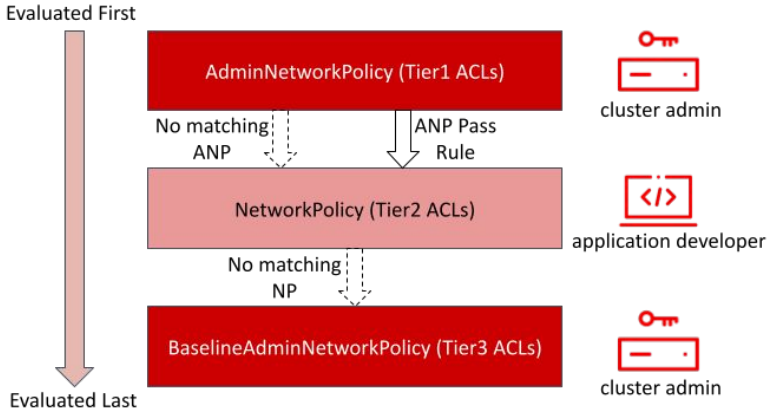


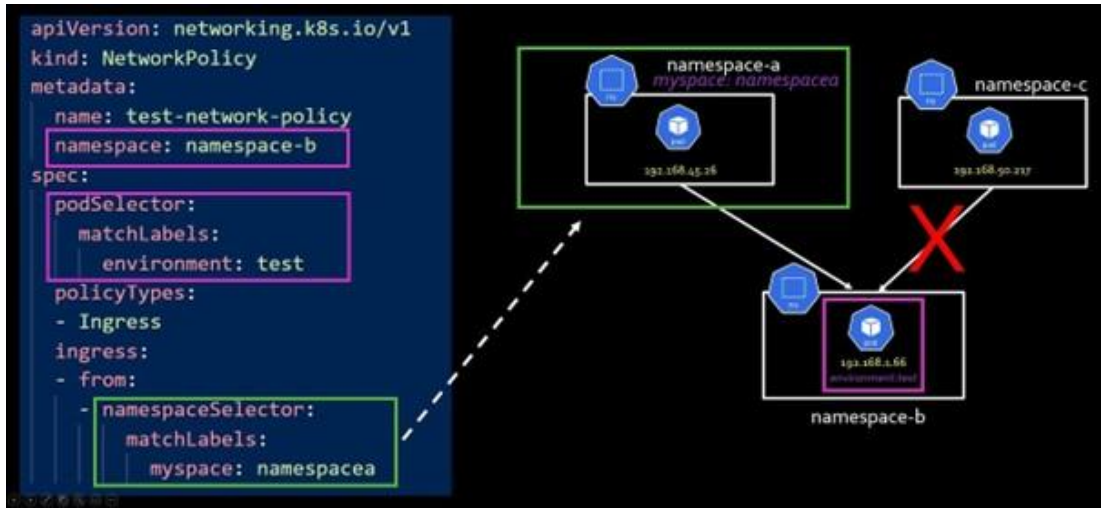
Introduction

Introduction

kubernetes
Network Policies







Open Source Projects Involved

KubeCon | CloudNativeCon
Europe 2025

Workshop Pod Section

https://www.menti.com/

← Mentimeter Quiz

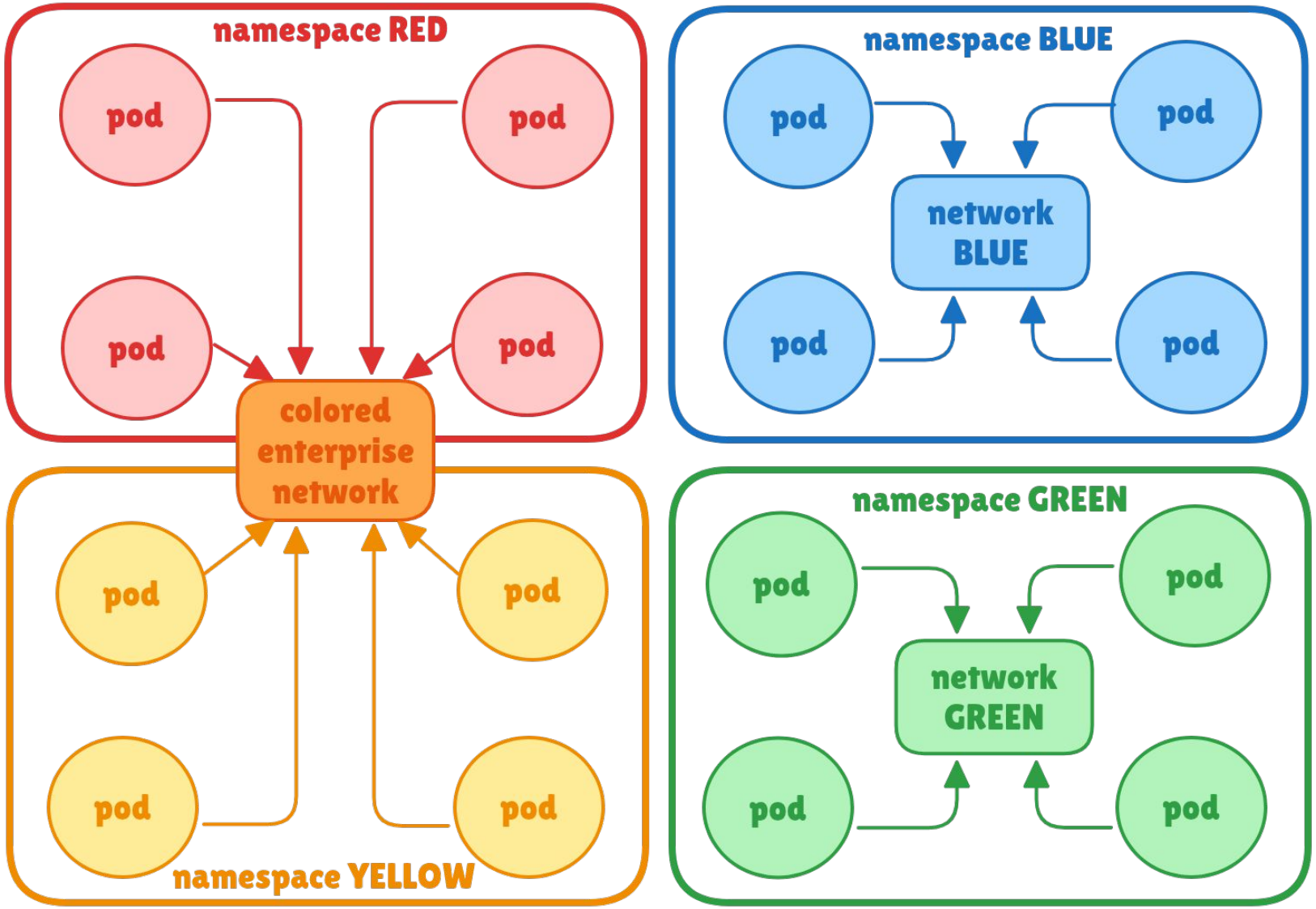
remove me!

Demo Agenda for Part-1

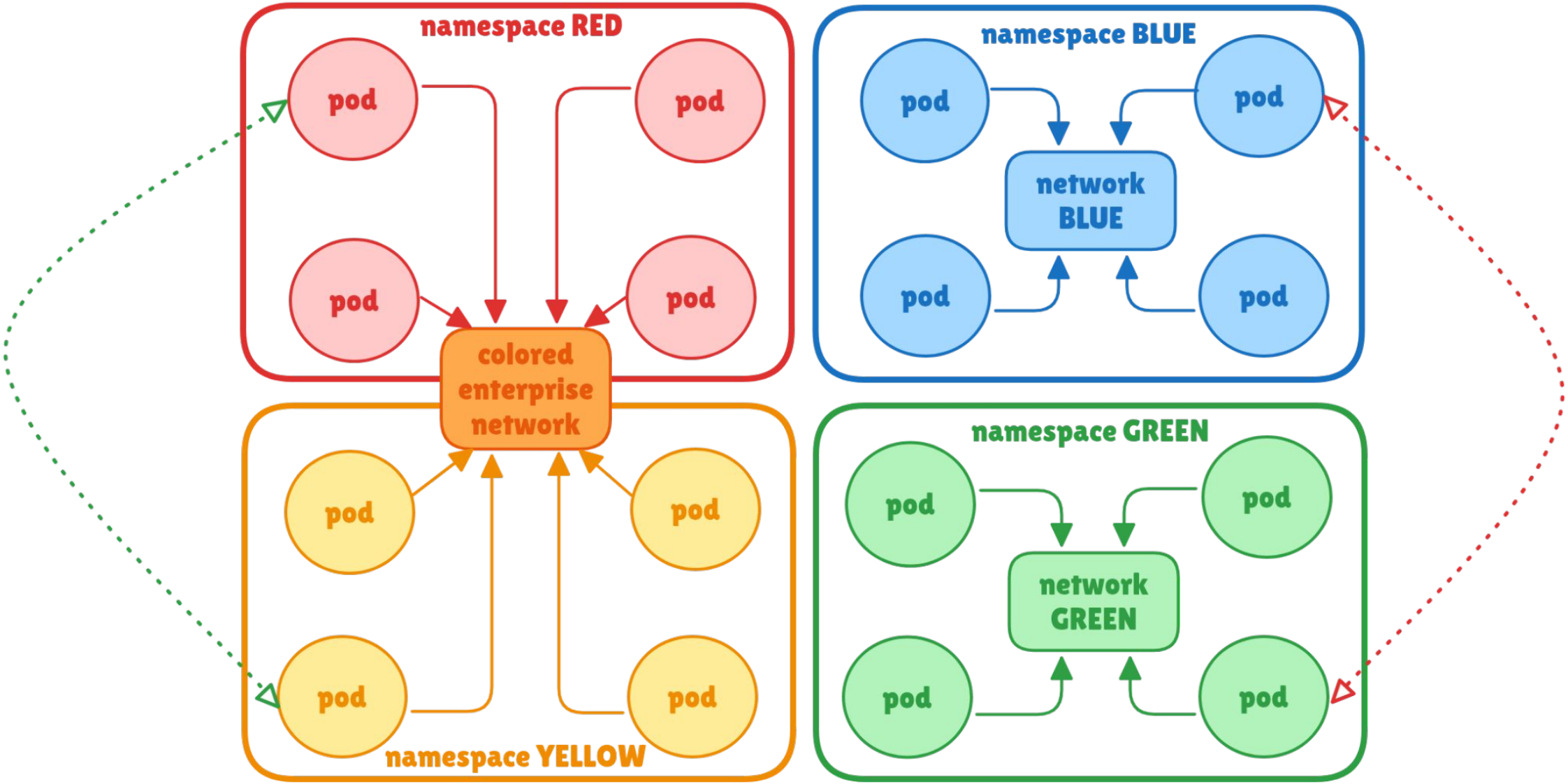
- Create UDNs and CUDN
- Attach pods to different UDNs
- Ensure pods are isolated
- Services + UDNs
- Ensure Ingress/Egress works for UDN pods
- NetworkPolicies + UDNs

← Workshop YAMLs, Cheat Sheet

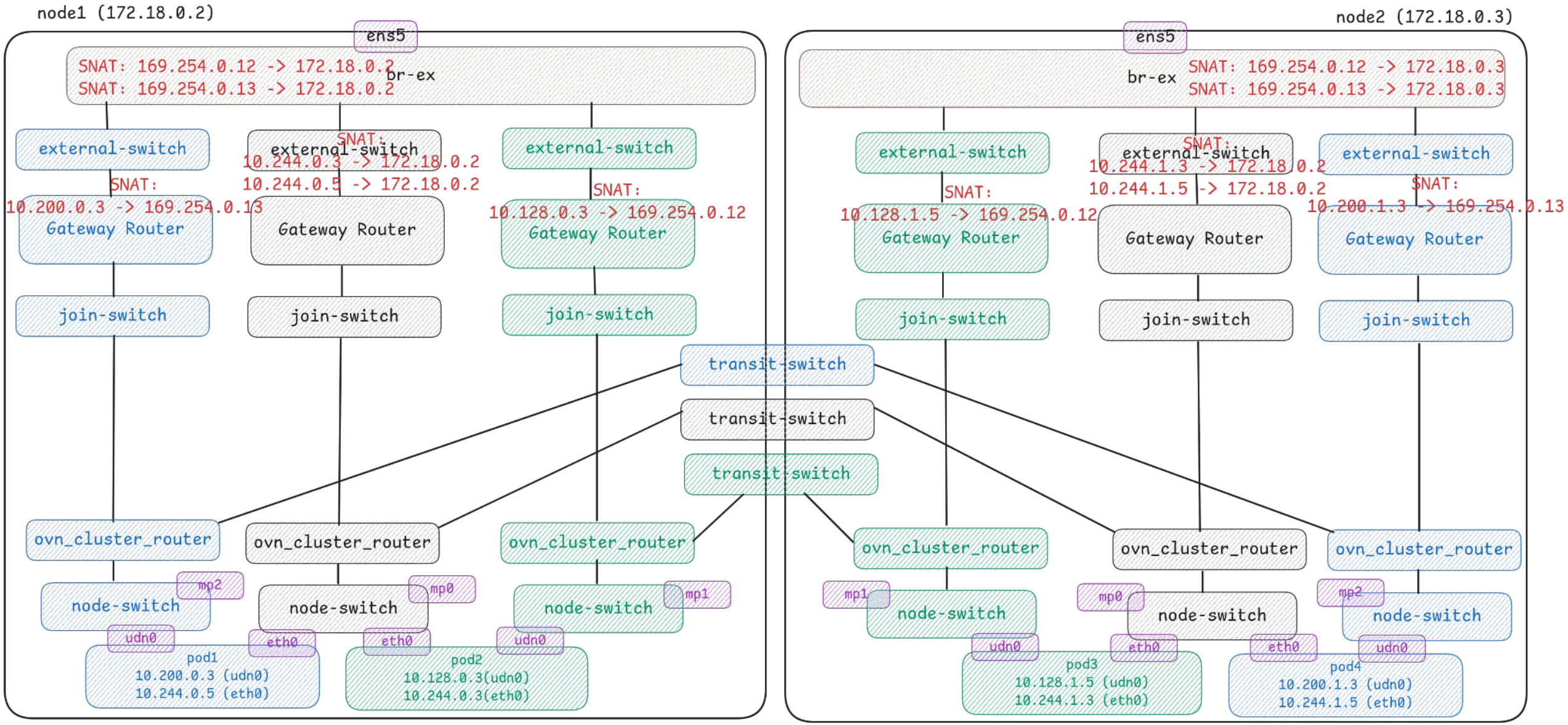
Demo setup for Pods on UDNs



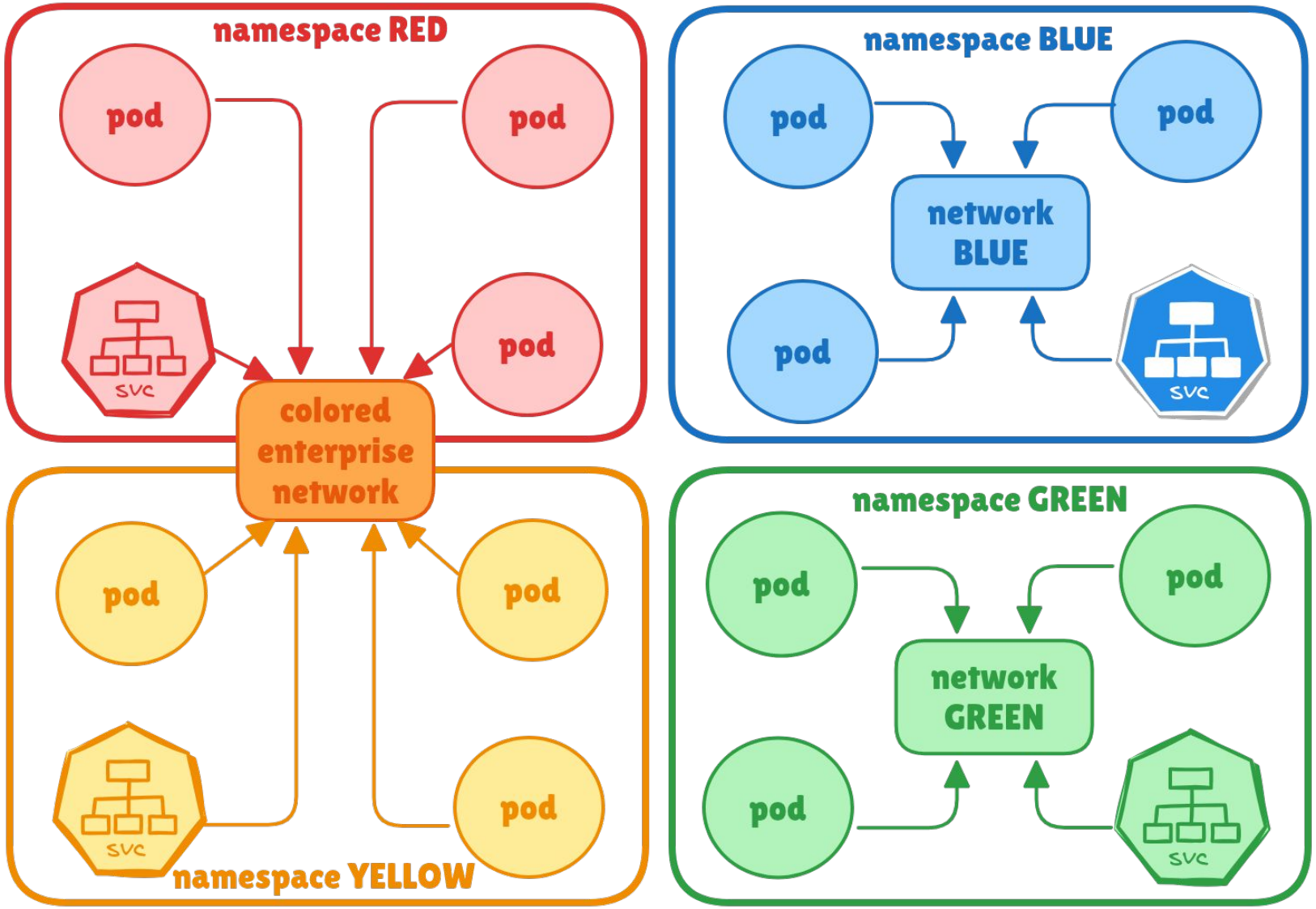
Pods on UDNs


namespace RED
pod
pod
pod
pod
namespace BLUE
pod
pod
network
BLUE
pod
pod
colored
enterprise
network
pod
pod
pod
pod
namespace YELLOW
namespace GREEN
pod
pod
network
GREEN
pod
pod

Isolation on UDNs by OVN-Kubernetes



Services on UDNs



QUIZ: How do you think serviceCIDR (clusterIP range) should work with UDNs?

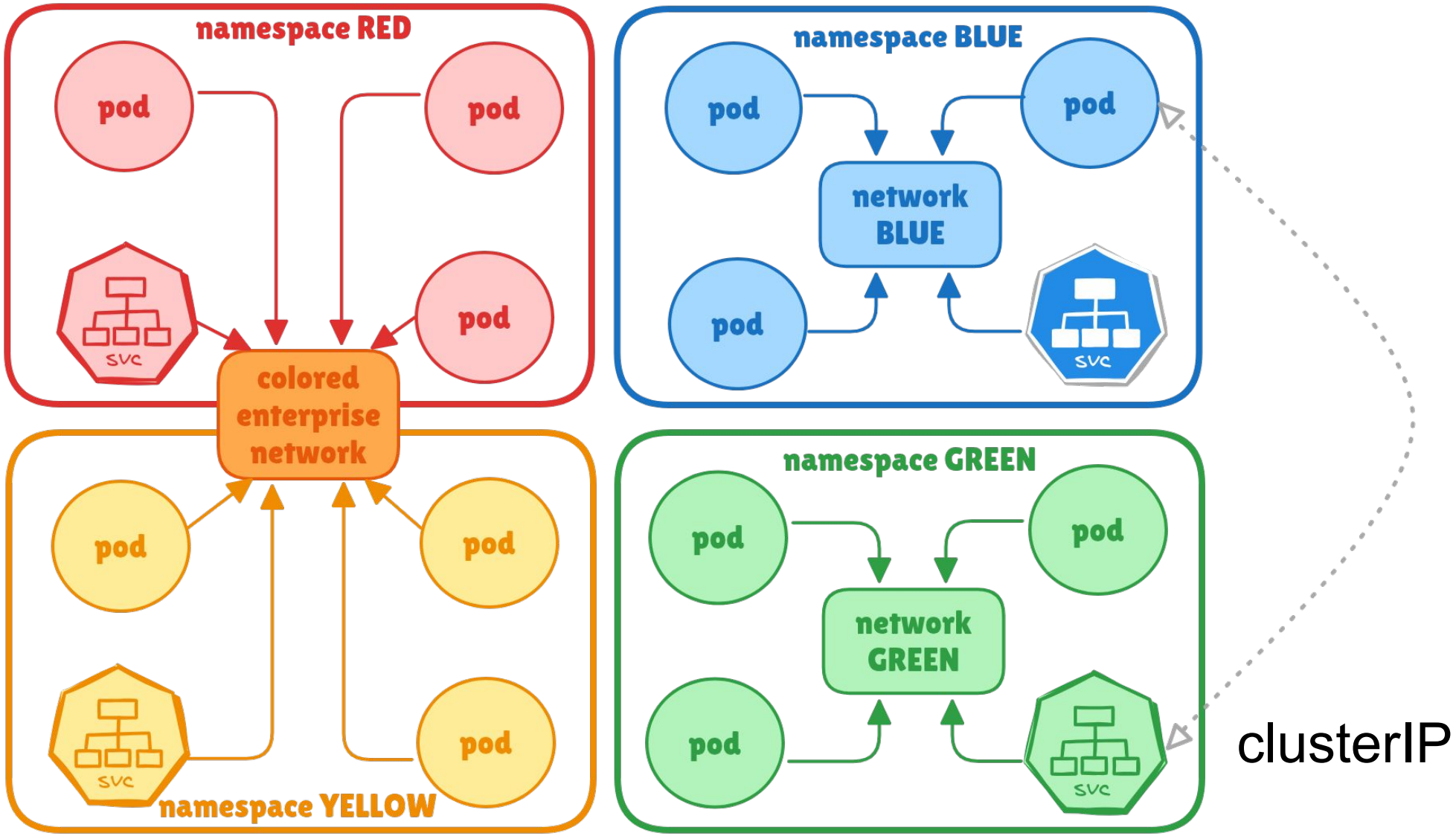
Options:

1. Each UDN should have its own serviceCIDR

2. All UDNs will share the same cluster wide serviceCIDR

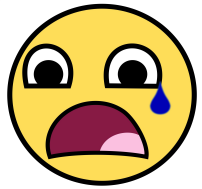
3. Neither, I have another idea I want to share

Services on UDNs

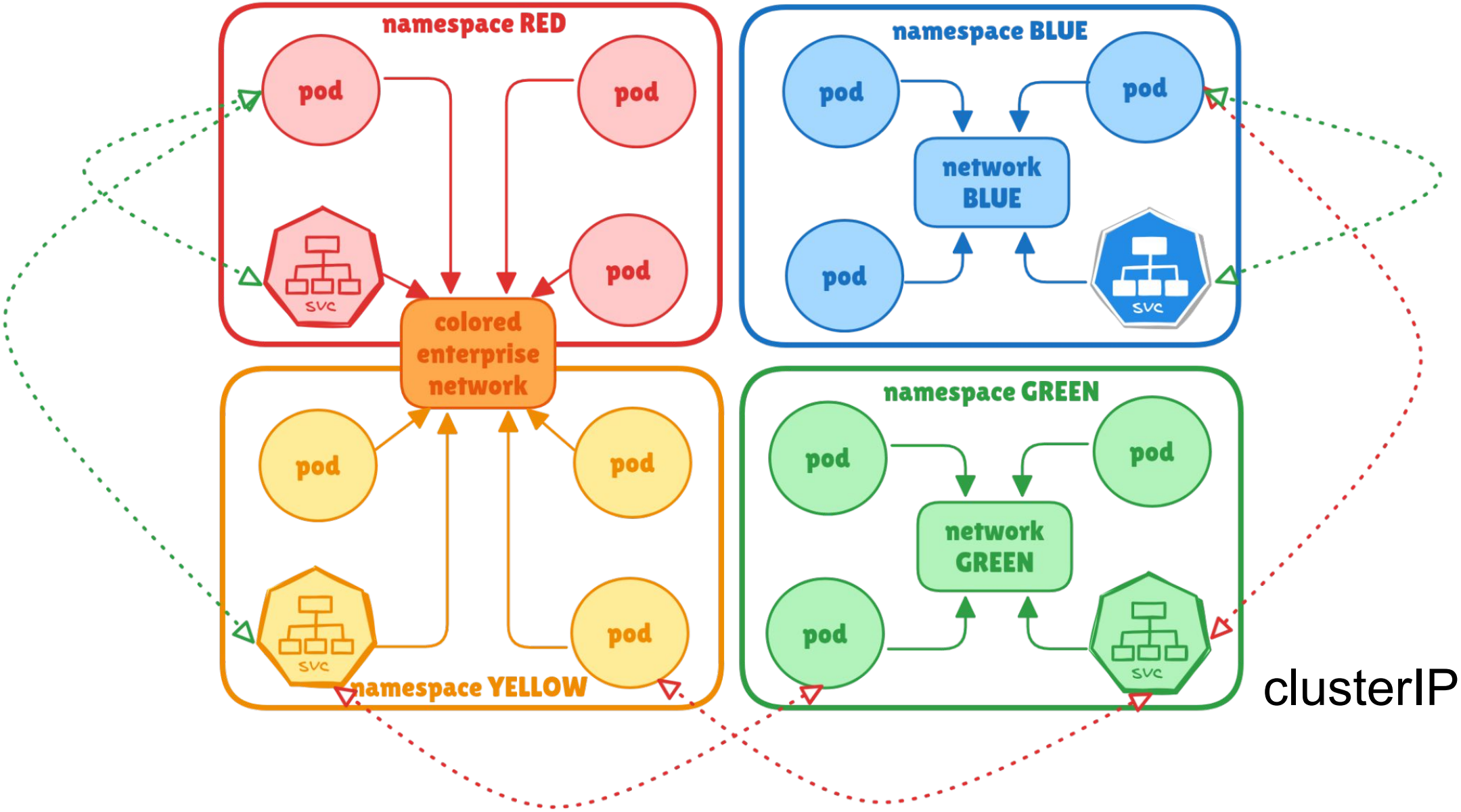


QUIZ: What color should that connection be?

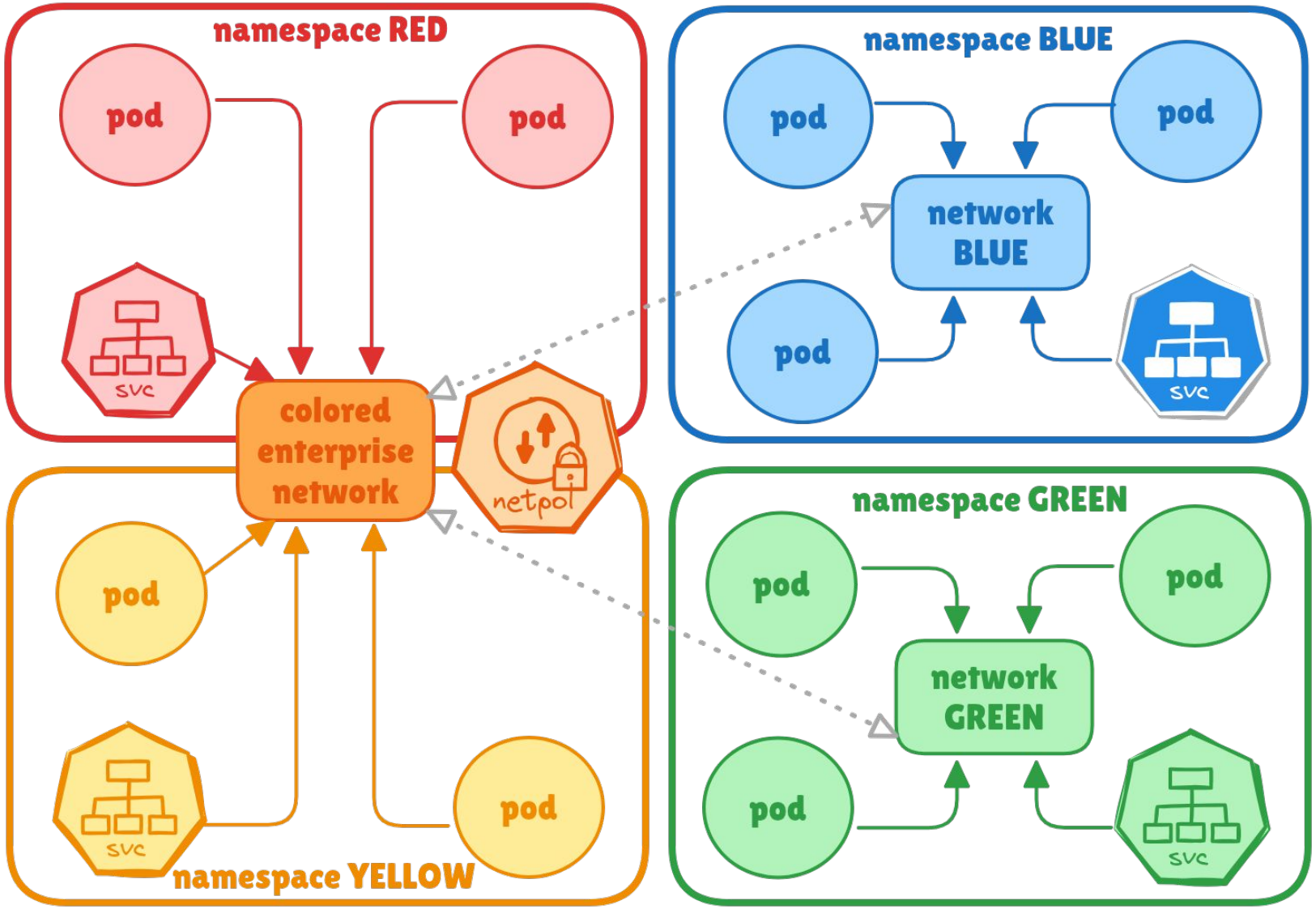
Options:

1. Green (It works!) 
2. Red (It doesn't work) 
3. Neither, should be user configurable 

Services on UDNs



Network Policies on UDNs

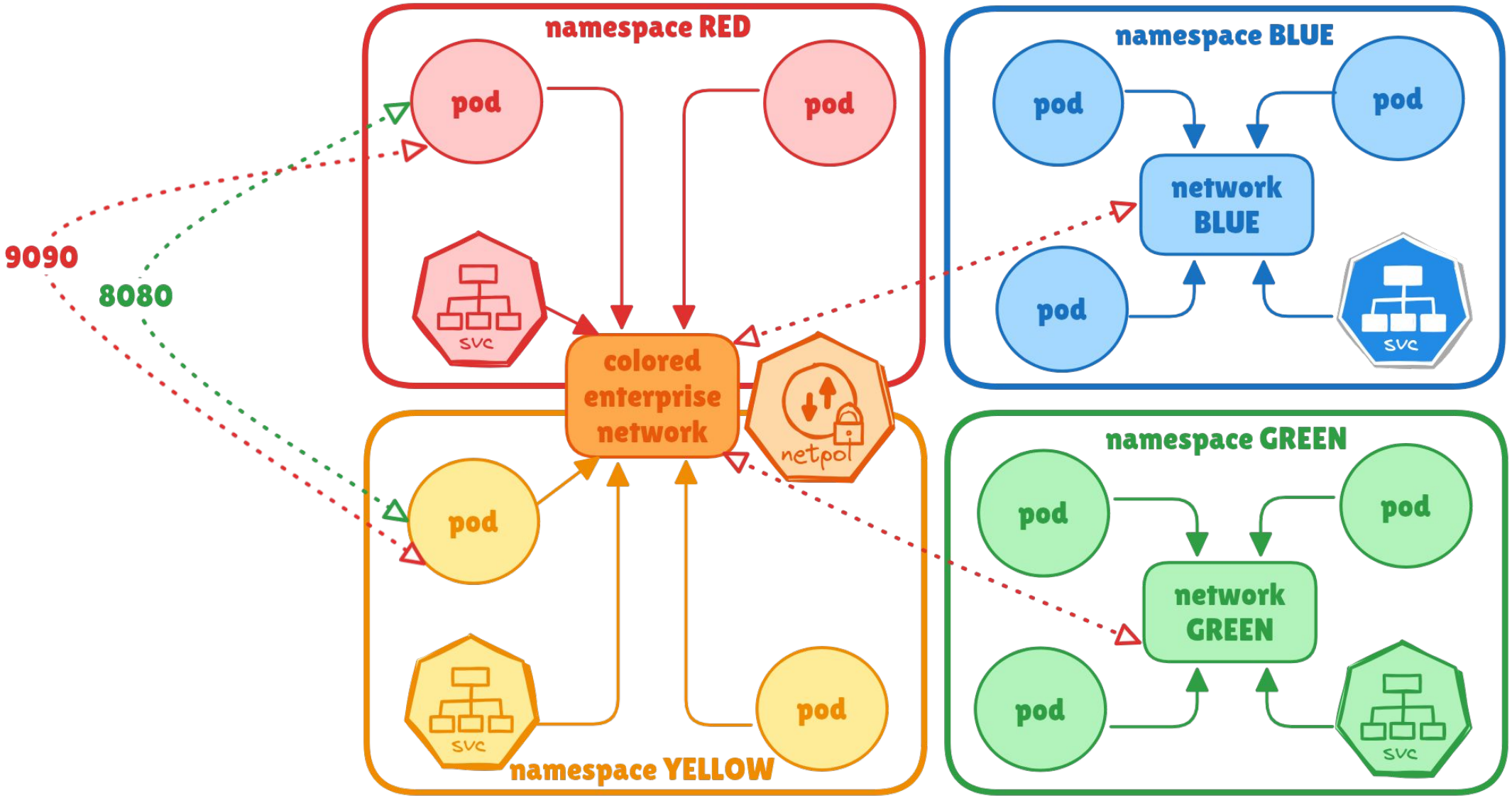


QUIZ: Should a NetworkPolicy allow rule defined between two UDNs work?

Options:

1. Yes! (Network Policies should take precedence over UDN Segmentation)
2. No! (UDN Segmentation should take precedence over Network Policies)
3. Neither, I have a better idea!
4. Meh.. too confusing and complex!

Network Policies on UDNs



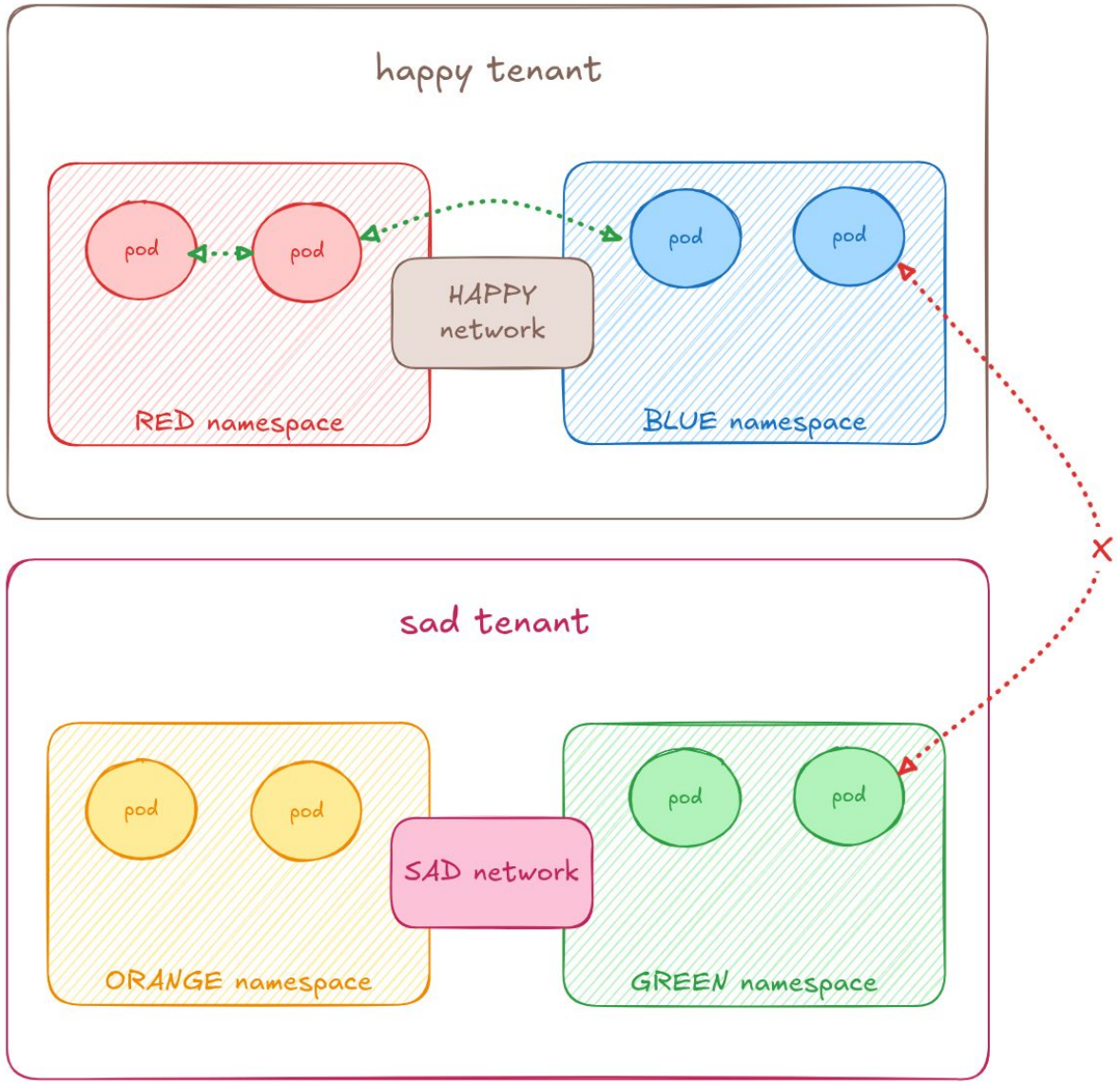
KubeCon | CloudNativeCon
Europe 2025

Workshop Virt Section

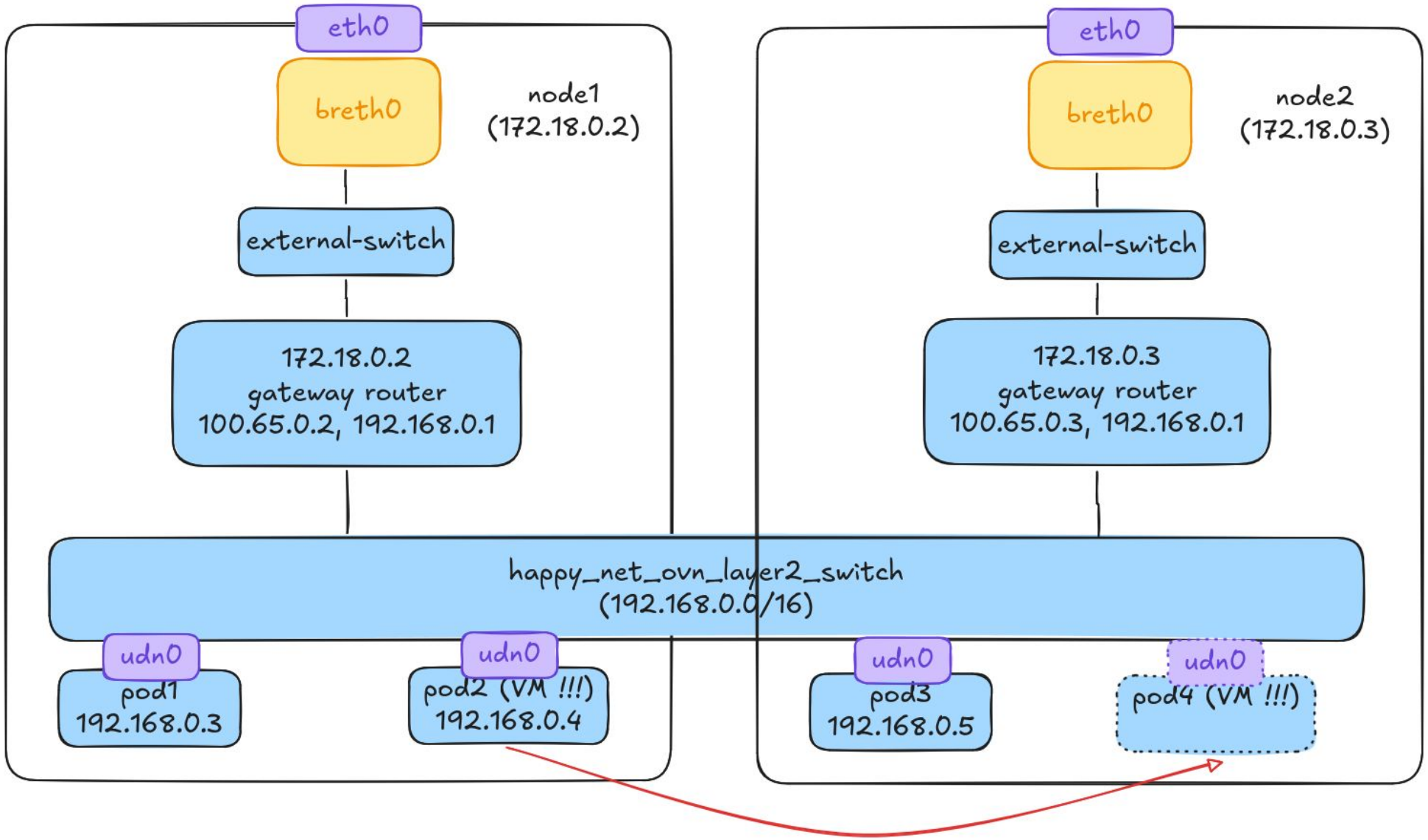
Demo Agenda for Part-2

- Create CUDN
- Create VMs in each of the CUDN namespaces
- Ensure egress to the www works as expected
- Ensure east/west traffic works as expected
- Ensure VM live-migration does not break established TCP connections

Demo setup for VMs on UDNs



VM live-migration

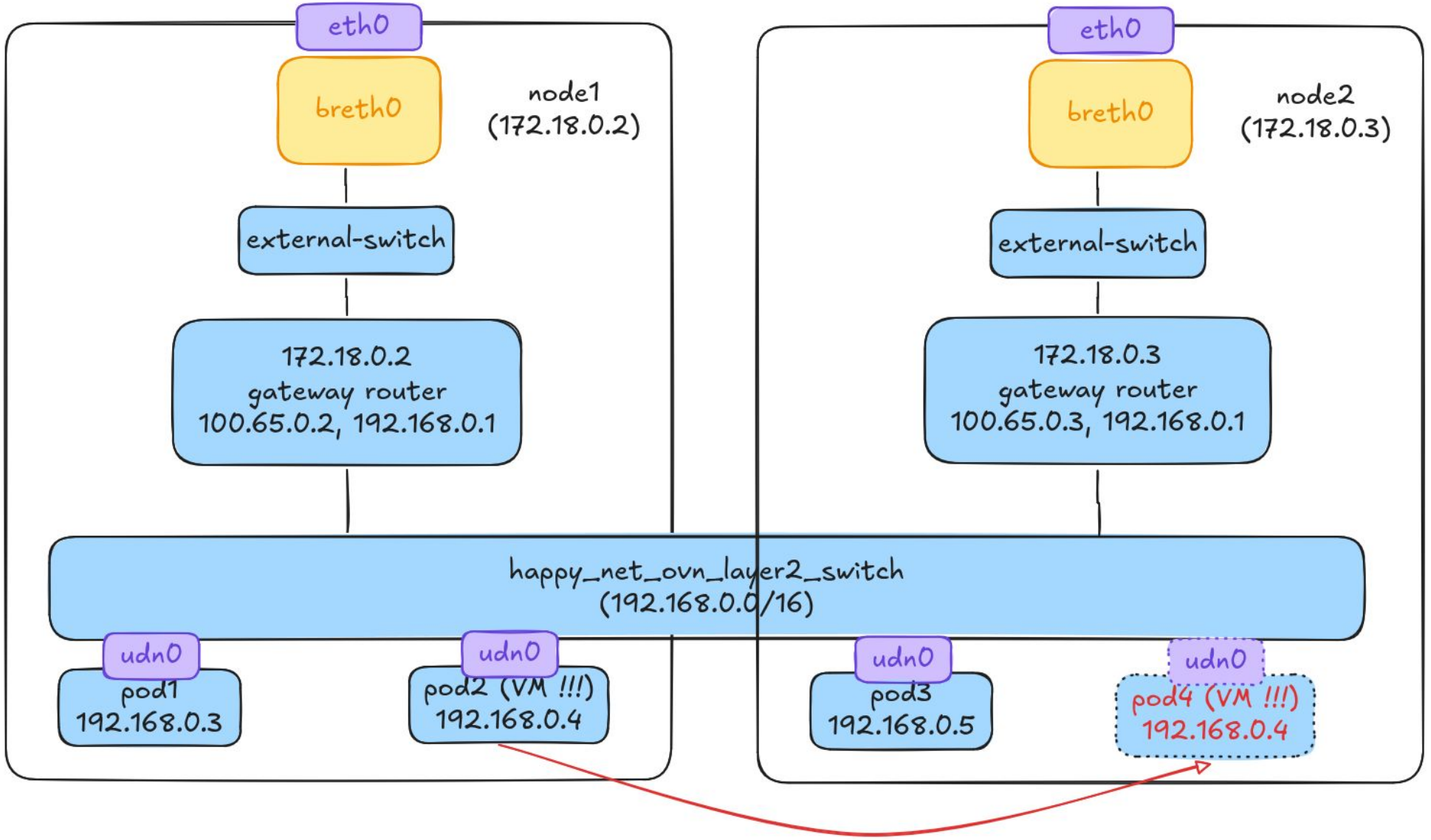


QUIZ: What do you think will happen to the VM's IP during live-migration ?

Options:

1. It'll change !
2. It'll stay the same !
3. The VM will crash and burn

QUIZ answer !!!


eth0
breth0
node1
(172.18.0.2)
external-switch
172.18.0.2
gateway router
100.65.0.2, 192.168.0.1
eth0
breth0
node2
(172.18.0.3)
external-switch
172.18.0.3
gateway router
100.65.0.3, 192.168.0.1
happy_net_ovn_layer2_switch
(192.168.0.0/16)
udn0
pod1
192.168.0.3
udn0
pod2 (VM !!!)
192.168.0.4
udn0
pod3
192.168.0.5
udn0
pod4 (VM !!!)
192.168.0.4

VM live-migration requirements

- VM’s IPAM configuration must follow the VM’s lifecycle
  - IP addresses
  - Gateways
  - DNS configuration

Persisting the IP addresses (1/2)

KubeCon CloudNativeCon Europe 2025

```
[root@fc40 kubecon-eu-2025-london-udn-workshop]# kubectl get vmi -nred-namespace red -ojsonpath="{@.status.interfaces[]}" | jq
{
  "infoSource": "domain, guest-agent",
  "interfaceName": "eth0",
  "ipAddress": "192.168.0.5",
  "ipAddresses": [
    "192.168.0.5"
  ],
  "linkState": "up",
  "mac": "0a:58:c0:a8:00:05",
  "name": "happy",
  "podInterfaceName": "ovn-udn1",
  "queueCount": 1
}
```

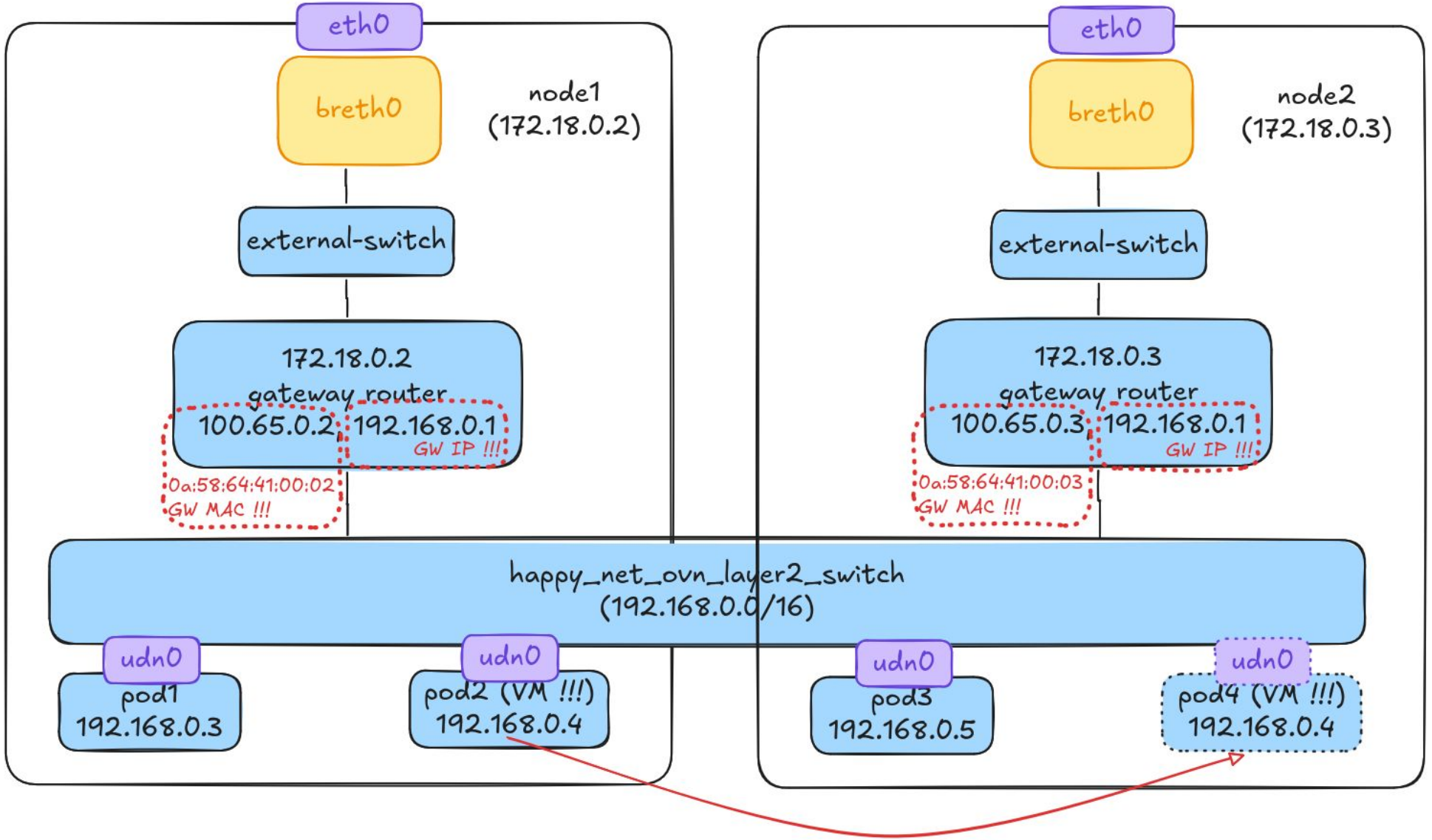
Persisting the IP addresses (2/2)

KubeCon CloudNativeCon Europe 2025

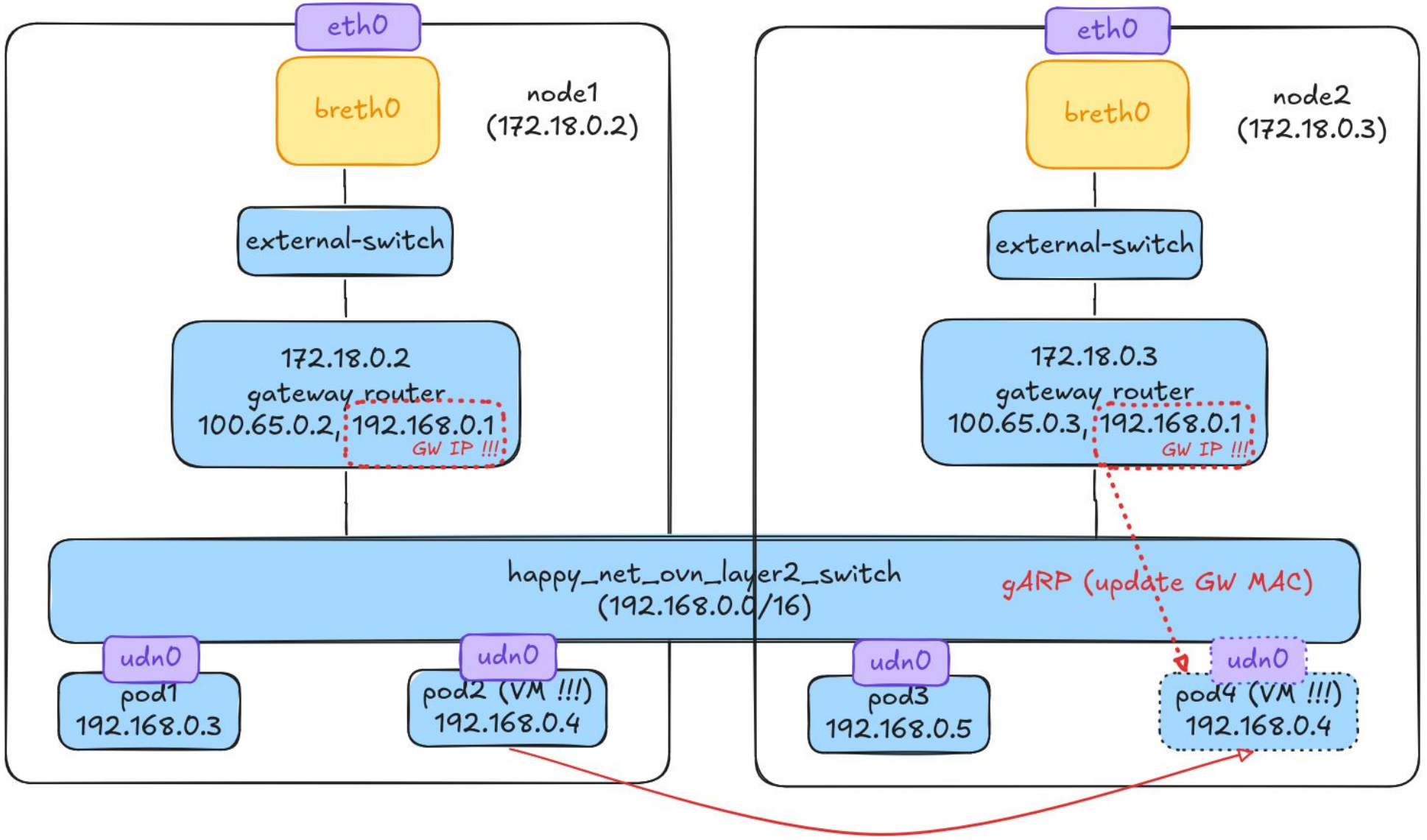
```
[root@fc40 kubecon-eu-2025-london-udn-workshop]# kubectl get vmi -nred-namespace red -ojsonpath="{@.status.interfaces[]}" | jq
{
  "infoSource": "domain, guest-agent",
  "interfaceName": "eth0",
  "ipAddress": "192.168.0.5",
  "ipAddresses": [
    "192.168.0.5"
  ],
  "linkState": "up",
  "mac": "0a:58:c0:a8:00:05",
  "name": "happy",
  "podInterfaceName": "ovn-udn1",
  "queueCount": 1
}
```

```
[root@fc40 kubecon-eu-2025-london-udn-workshop]# kubectl get ipamclaim -nred-namespace red.happy -ojsonpath="{@.status}" | jq
{
  "ips": [
    "192.168.0.5/16"
  ]
}
```

Persisting the Gateway (1/2)



Persisting the Gateway (2/2)



VM live-migration implementation

- VM's IPAM configuration must follow the VM's lifecycle
  - IP addresses
  - Gateways
  - DNS configuration
- IP addresses are persistent
  - Using **`IPAMClaim`** CRs
- Gateways are updated by the SDN
  - Gratuitous ARPs are sent - IPv4
  - RouterAdvertisements are sent - IPv6
- DNS is … tricky
  - Access to kubeDNS
  - No DNS resolution within the primary UDN

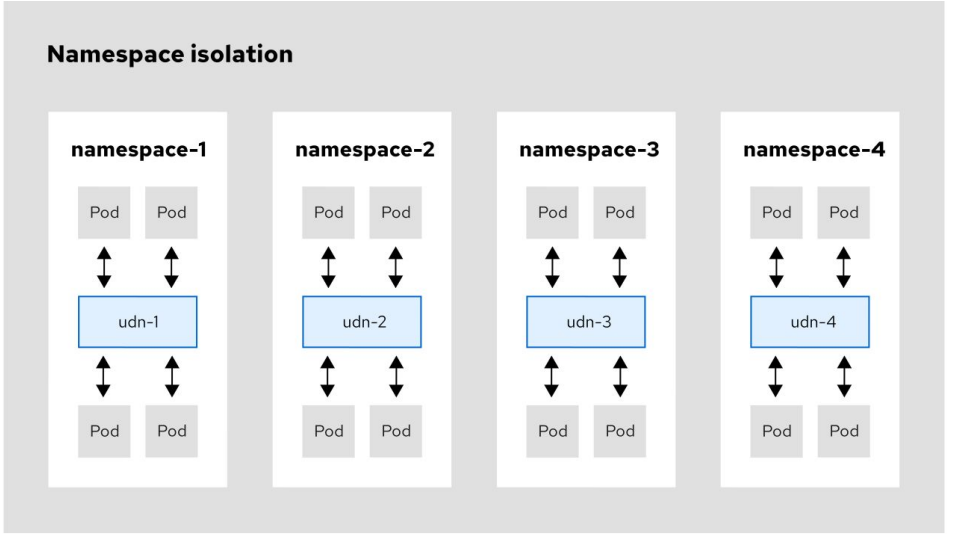
QUIZ: DNS + UDNs! What should the future look like?

Options:

1. The cluster (core) DNS be multi-network aware
2. We should add a core DNS per UDN
3. Neither, I have a better idea I'd like to share!

Conclusion

```
apiVersion: k8s.ovn.org/v1
kind: UserDefinedNetwork
metadata:
  creationTimestamp: "2024-08-28T17:18:47Z"
  finalizers:
  - k8s.ovn.org/user-defined-network-protection
  generation: 1
  name: udn-1
  namespace: some-custom-namespace
  resourceVersion: "53313"
  uid: f483626d-6846-48a1-b88e-6bbeb8bcde8c
spec:
  layer2:
    role: Primary
    subnets:
    - 10.0.0.0/24
    - 2001:db8::/60
  topology: Layer2
status:
  conditions:
  - lastTransitionTime: "2024-08-28T17:18:47Z"
    message: NetworkAttachmentDefinition has been created
    reason: NetworkAttachmentDefinitionReady
    status: "True"
    type: NetworkReady
```



KubeCon | CloudNativeCon
Europe 2025

THANK YOU!

QUESTIONS?

FEEDBACK