# Network Service Mesh Webinar

## Agenda

- Housekeeping
- NSM Vision
- State of the NSM
- NSM Future
- Deep Dive:
    - How the Magic Works
    - Interdomain
    - HW NICs

## Housekeeping

https://networkservicemesh.io

NSMCon

These slides

Nov 18, 2019 | San Diego, California
Colocated with Kubecon+CloudNativeCon 2019

Network Service Mesh

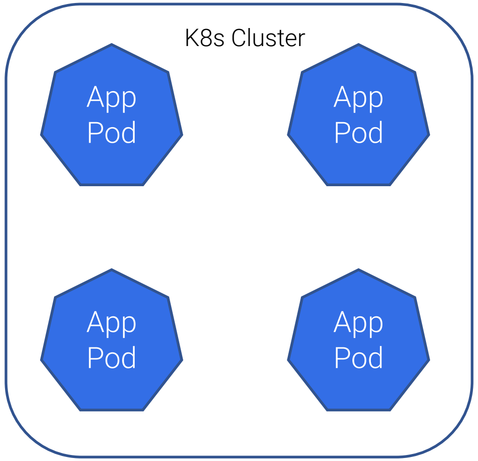
# NSM Vision

# NSM Vision

- The Problems
- The Non-Solutions
- The NSM Solutions

# Runtime Domain

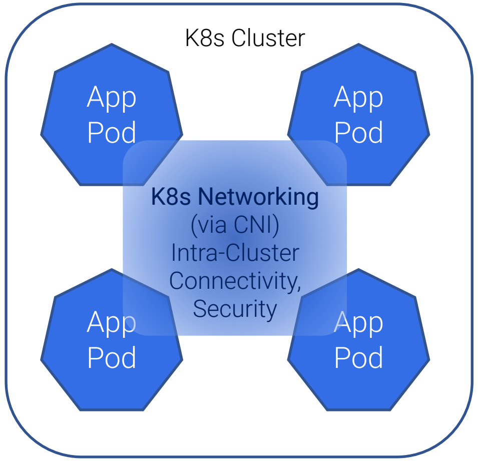
K8s is a ‘Runtime Domain’…



# Connectivity Domain

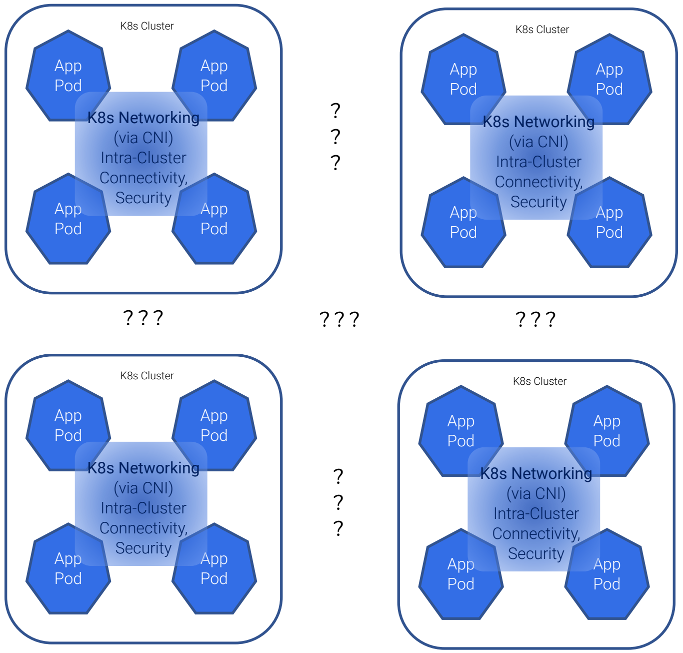
K8s is a 'Runtime Domain'...
With a 'Connectivity Domain'...

- Pure L3
- + **Service Discovery/Routing:** K8s Services
- + **Isolation:** K8s Network Policies (Isolation)
- (Optionally) + L7 Service Mesh(Istio etc)
- Intra Cluster



# The Problems

What about East/West traffic between workloads (Pods) in different clusters?

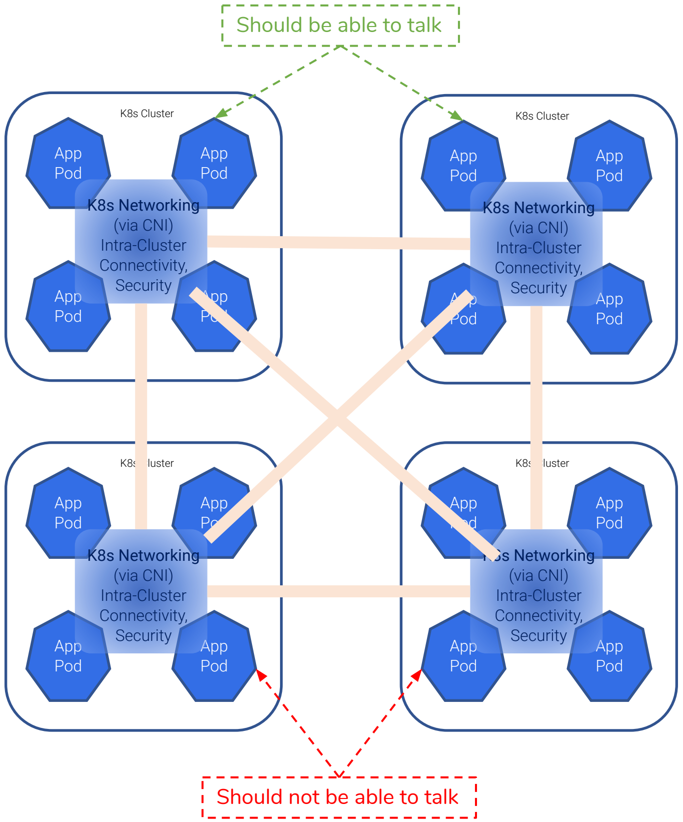

K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
? ? ?
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
? ? ?
? ? ?
? ? ?
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
? ? ?
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

# The Non-Solutions

All non-NSM attempts involve cluster-to-cluster networking:

Problems:

- **Inter-cluster Workload Isolation**

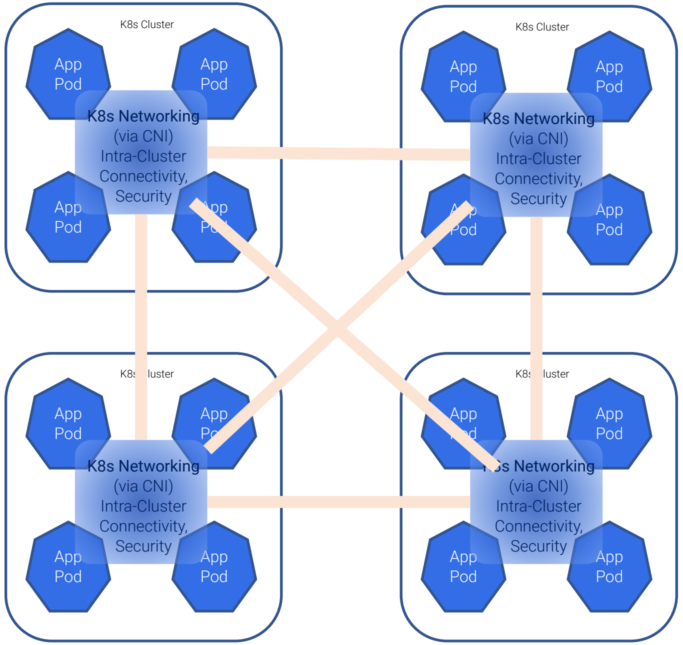

Should be able to talk
K8s Cluster
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
Should not be able to talk

# The Non-Solutions

All non-NSM attempts involve cluster to cluster networking:

Problems:

- Inter-cluster Workload Isolation
- **Full Mesh between clusters explodes combinatorics – number of links scales like number of clusters choose 2 (ie: factorially)**

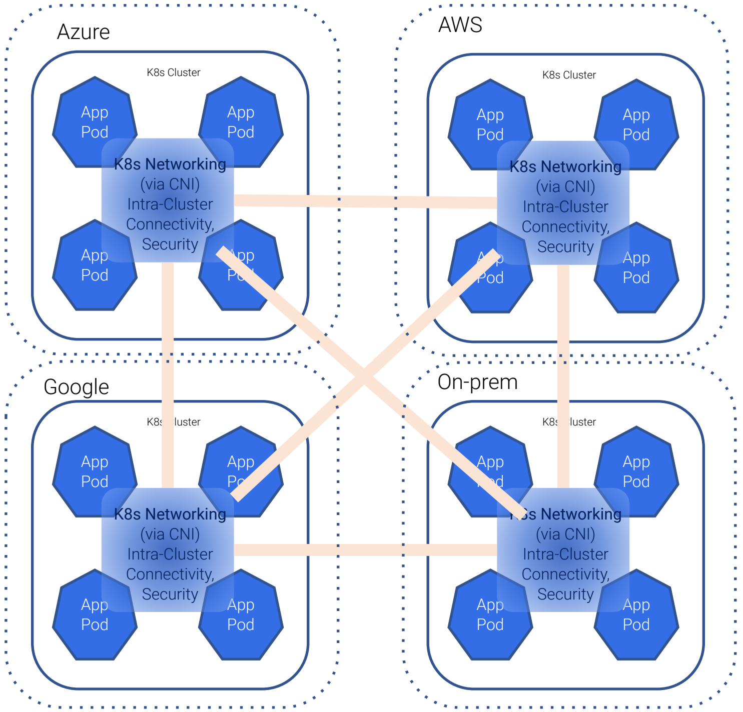

K8s Cluster
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security

# The Non-Solutions

All non-NSM attempts involve cluster to cluster networking:

Problems:

- Inter-cluster Workload Isolation
- Full Mesh between clusters explodes combinatorics – number of links scales like number of clusters choose 2 (ie: factorially)
- **Complex often manual cluster to cluster link setup between different public/private cloud providers (possibly involving complex firewall rules depending on how its done).**

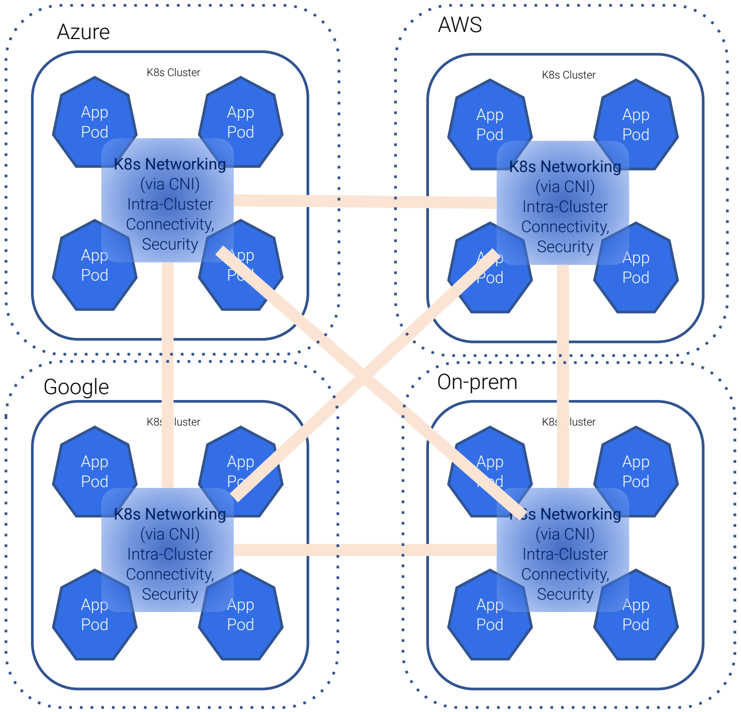

Azure
AWS
Google
On-prem
K8s Cluster
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security

# The Non-Solutions

All non-NSM attempts involve cluster to cluster networking:

Problems:

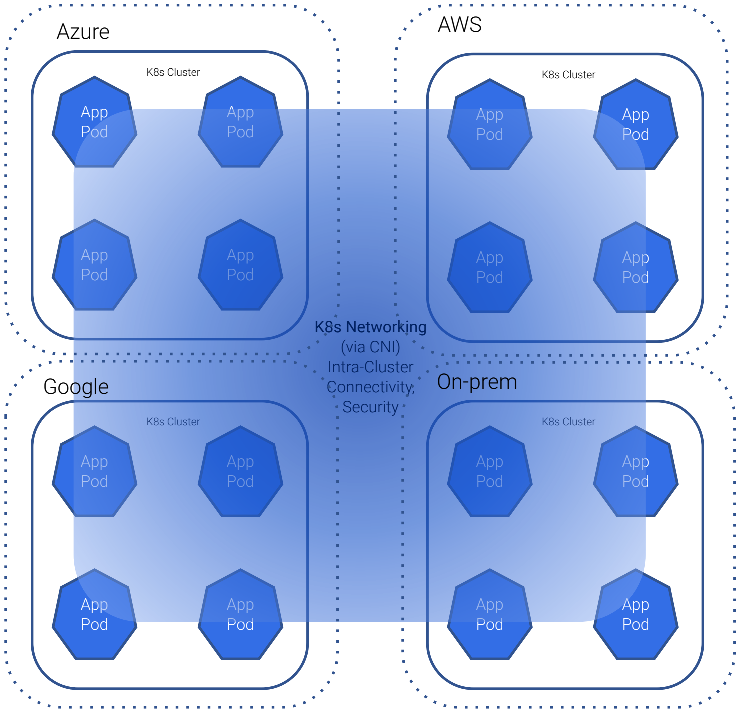
- Inter-cluster Workload Isolation
- Full Mesh between clusters explodes combinatorics – number of links scales like number of clusters choose 2 (ie: factorially)
- Complex often manual cluster to cluster link setup between different public/private cloud providers (possibly involving complex firewall rules depending on how its done).
- **Inter-cluster Service Discovery/Routing**


Azure
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
AWS
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
Google
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
On-prem
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

# The Federation non-Solution

Attempt to 'Federate' multiple clusters:

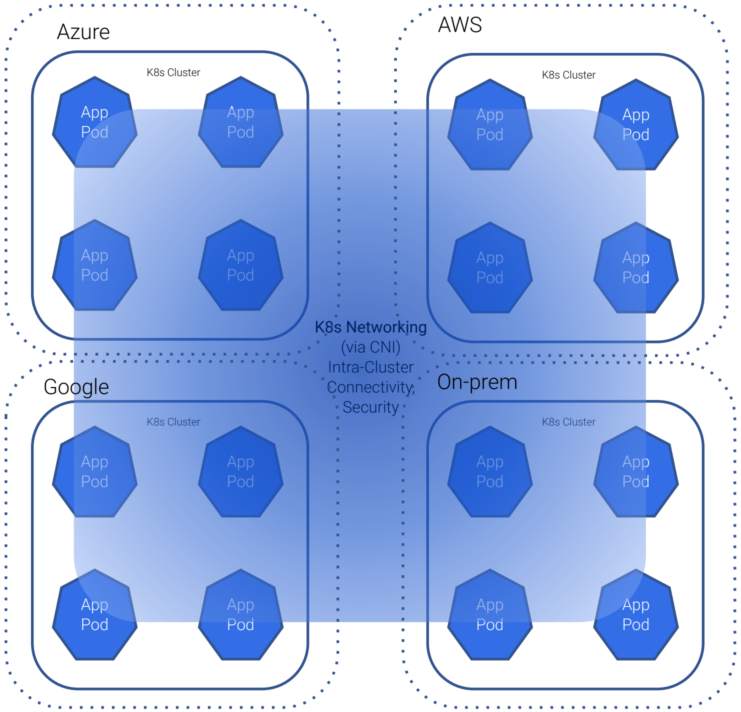
- Hides rather than fixes inter-cluster link combinatorics/complexity/manualness
- Doesn't scale:
  - Services/Network Policies have enough trouble scaling in a single cluster, with low latency updates

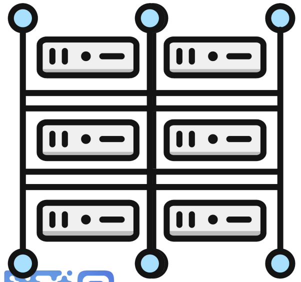


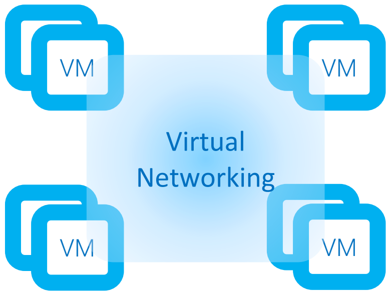
# The Federation non-Solution

Attempt to 'Federate' multiple clusters:

- Also incompatible with non-K8s runtime domains
  - Semantics of VM domain different than K8s
  - Semantics of on-prem server networking different than K8s



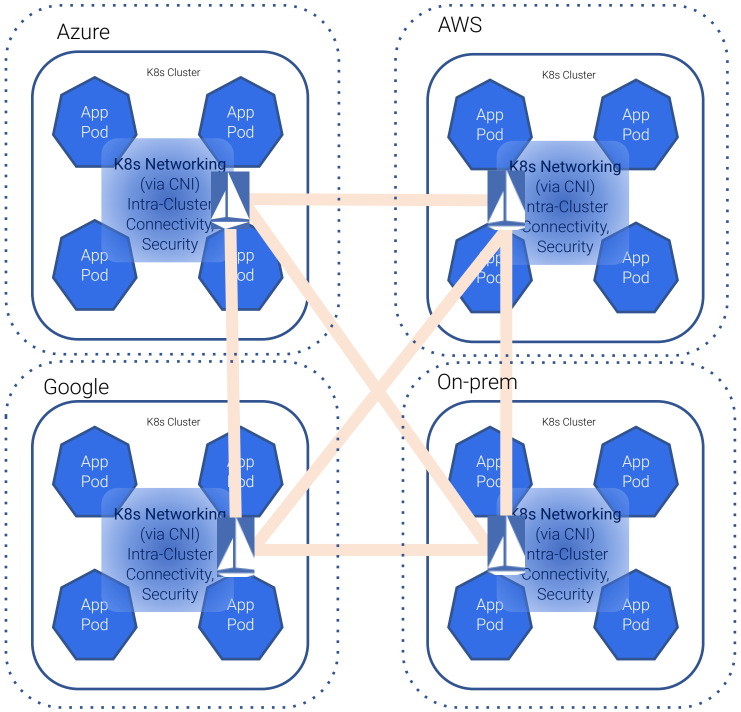




# The Service Mesh Intercluster Gateway non-Solution

Solving with Service Mesh(Istio etc)
Inter-cluster gateways
Problems:

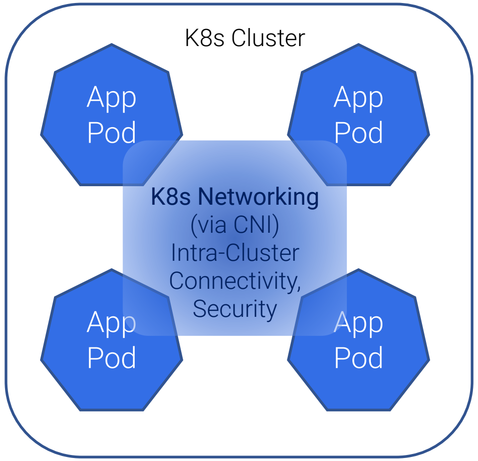
- Only works for L7, not L3
- Same full mesh combinatorics problems
- Same complex often manual cluster to cluster link setup between different public/private cloud providers (possibly involving complex firewall rules depending on how its done).



# The NSM Realization

**‘Connectivity Domain’ Independence:**

- **Welding your ’connectivity domain’ to your ‘runtime domain’ (cluster) is mistake**



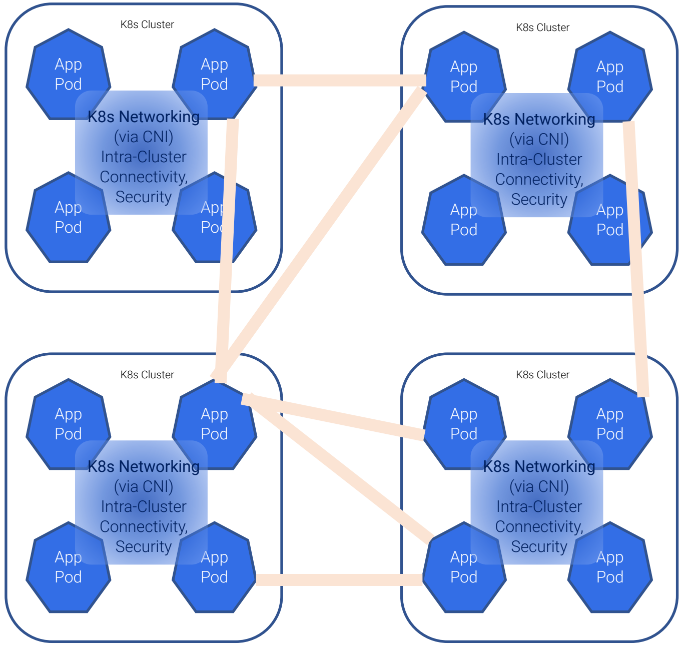
# The NSM Realizations

‘Connectivity Domain’ Independence:

- Welding your ’connectivity domain’ to your ‘runtime domain’ (cluster) is mistake

**What you really care about is**

**workload to workload connectivity: independent of runtime domain.**

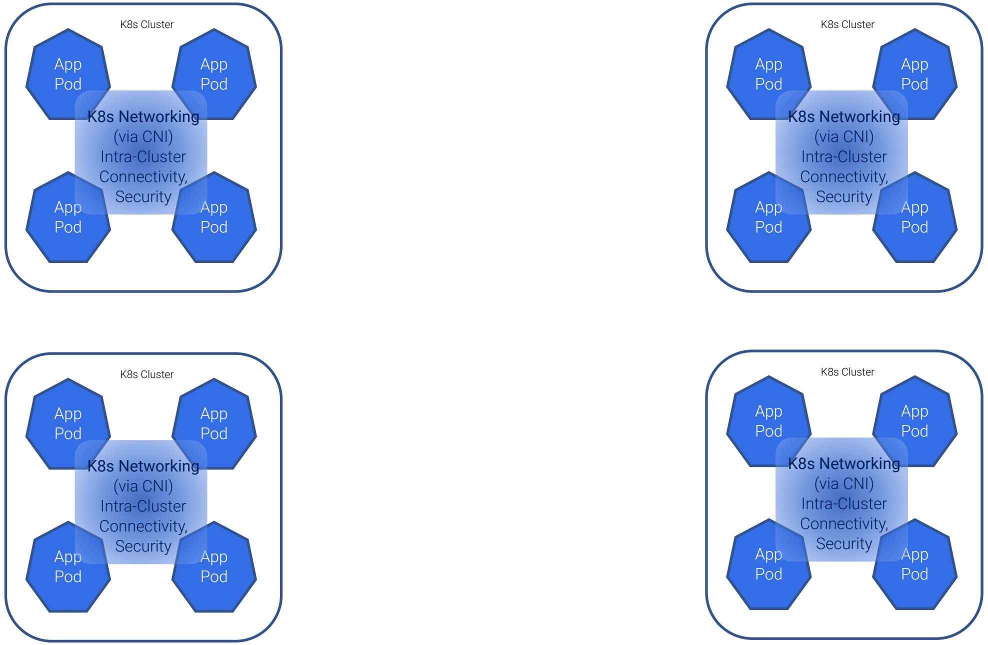


K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

# The NSM Solution

**Leave Intra-Cluster Networking Alone:**

- **Orthogonal to CNI**
- **Harmless to existing K8s Networking**


K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

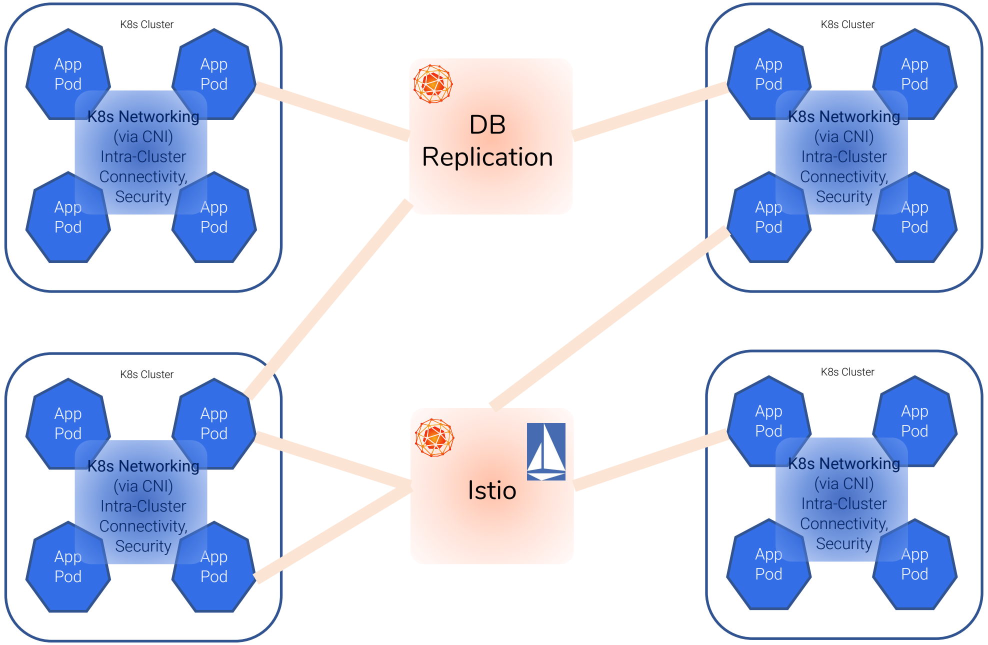
# The NSM Solution

Leave Intra-Cluster Networking Alone:

- Orthogonal to CNI
- Harmless to existing K8s Networking

**Allow workloads to connect to new 'connectivity domains':**

- **With the Connectivity/Security/Observability features needed in that connectivity domain**


K8s Cluster
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
DB Replication
Istio

# The NSM Solution

Leave IntraCluster Networking Alone:

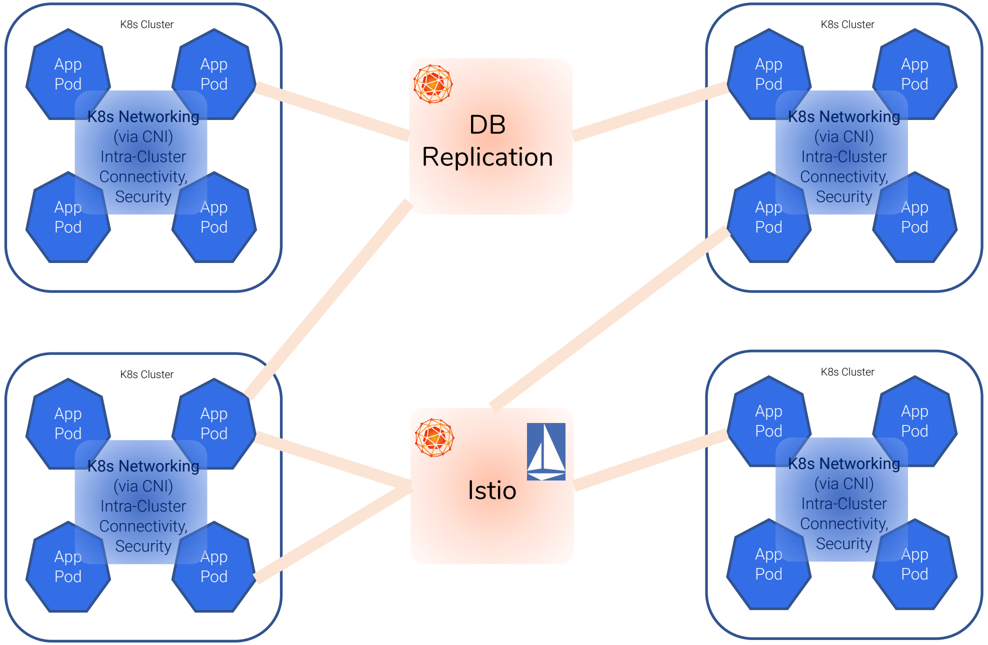
- Orthogonal to CNI
- Harmless to existing K8s Networking

Allow workloads to connect to new 'connectivity domains':

- With the **Connectivity/Security/Observability** features needed in that connectivity domain

**Examples:**

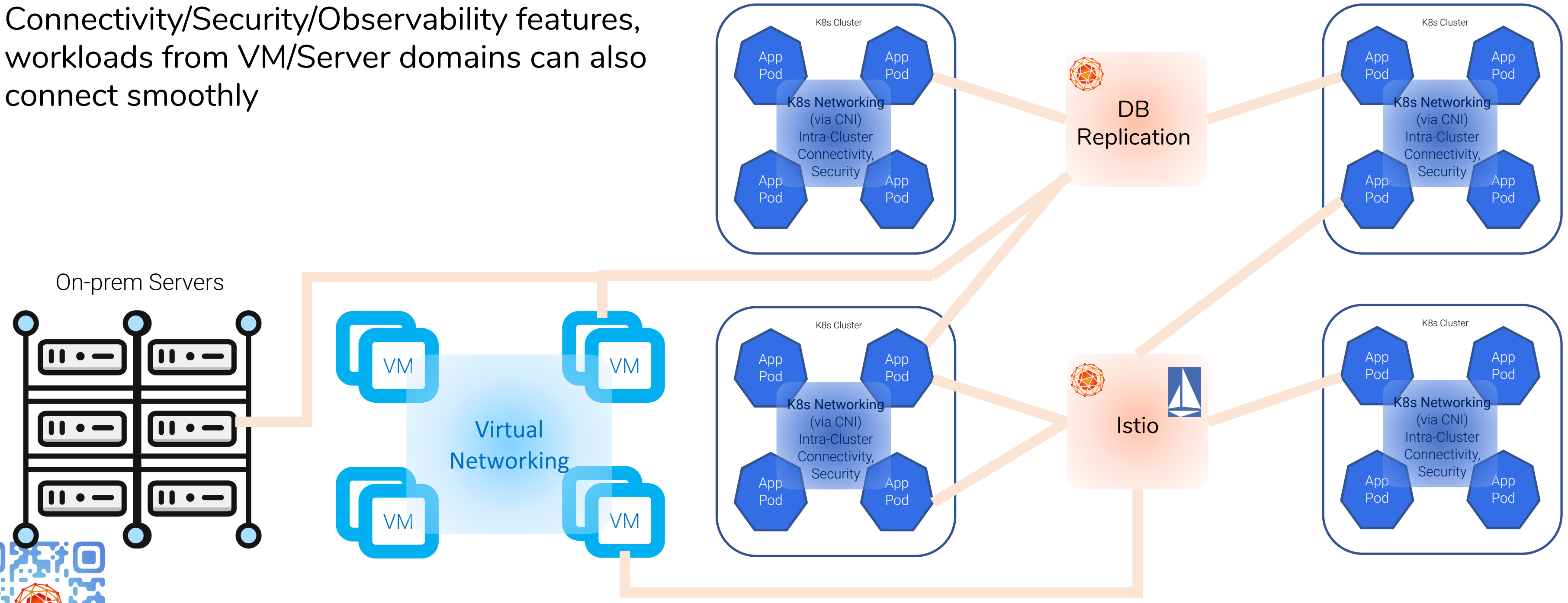
- **DB Replication Connectivity Domain**
  - **Pure vL3 domain between DB replicas: where-ever they may be**
- **Istio Connectivity Domain**
  - **Single Istio instance serving workloads wherever they may be over vL3 domain.**



K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
DB Replication
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
Istio
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

# The NSM Solution

Because connectivity domain has its own Connectivity/Security/Observability features, workloads from VM/Server domains can also connect smoothly



Network Service Mesh

## State of the NSM

## State of the NSM - Community

- **CNCF Project**: NSM is now a CNCF project

CLOUD NATIVE
COMPUTING FOUNDATION

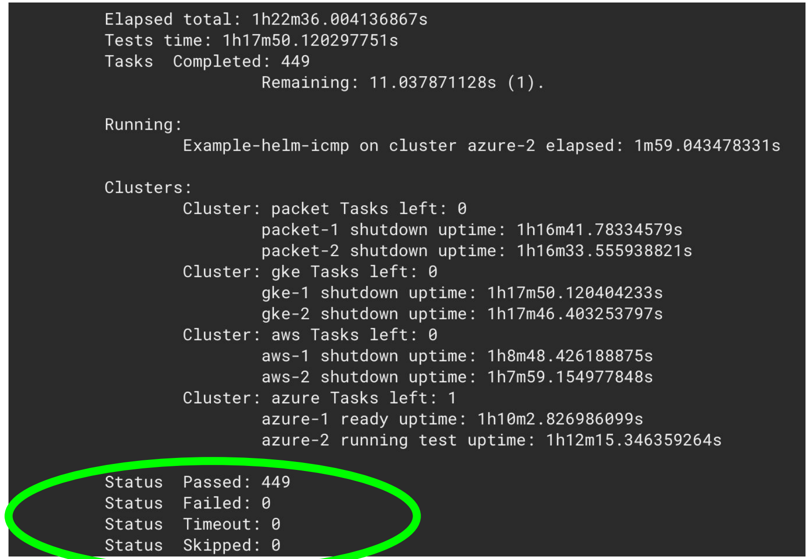
# State of the NSM - Early Use

- **CNF Testbed**: NSM is used in the CNF Testbed project for Cloud Native NFV for Telco.

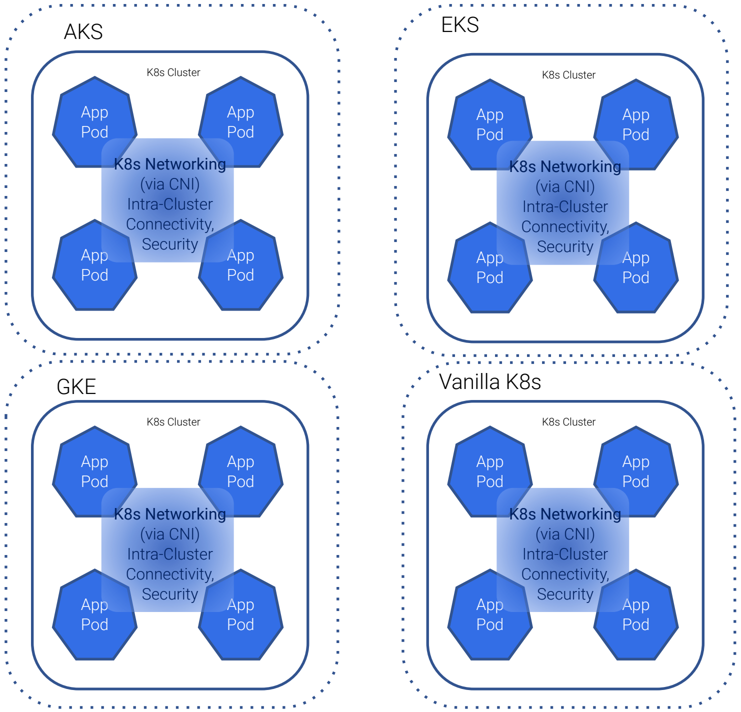
CNF Testbed

# State of the NSM

- CNCF Project
- **Multi-cloud CI:** NSM runs CI on AKS/EKS/GKE/Vanilla K8s (100+ tests each)

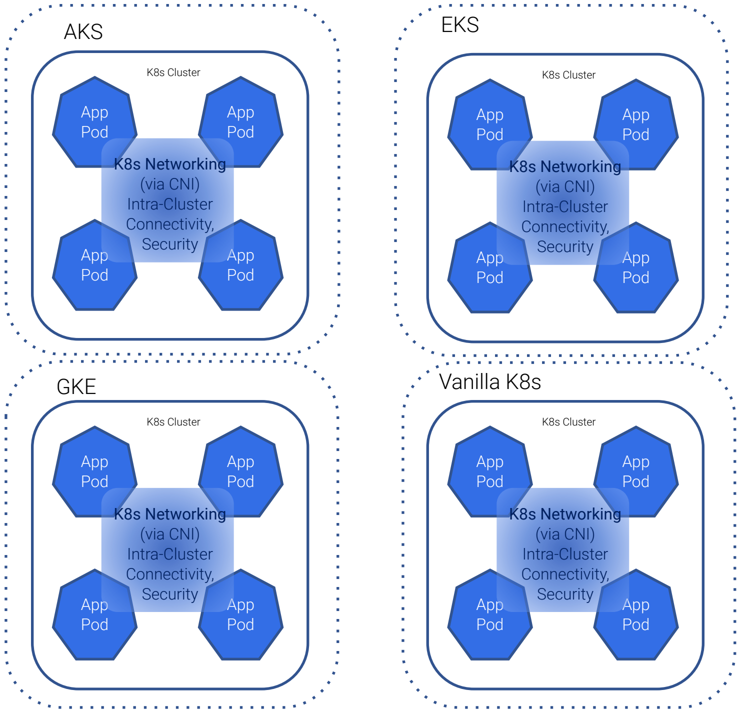


```
Elapsed total: 1h22m36.004136867s
Tests time: 1h17m50.120297751s
Tasks  Completed: 449
                Remaining: 11.037871128s (1).

Running:
        Example-helm-icmp on cluster azure-2 elapsed: 1m59.043478331s

Clusters:
        Cluster: packet Tasks left: 0
                packet-1 shutdown uptime: 1h16m41.78334579s
                packet-2 shutdown uptime: 1h16m33.555938821s
        Cluster: gke Tasks left: 0
                gke-1 shutdown uptime: 1h17m50.120404233s
                gke-2 shutdown uptime: 1h17m46.403253797s
        Cluster: aws Tasks left: 0
                aws-1 shutdown uptime: 1h8m48.426188875s
                aws-2 shutdown uptime: 1h7m59.154977848s
        Cluster: azure Tasks left: 1
                azure-1 ready uptime: 1h10m2.826986099s
                azure-2 running test uptime: 1h12m15.346359264s

Status  Passed: 449
Status  Failed: 0
Status  Timeout: 0
Status  Skipped: 0
```


AKS
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
EKS
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
GKE
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
Vanilla K8s
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

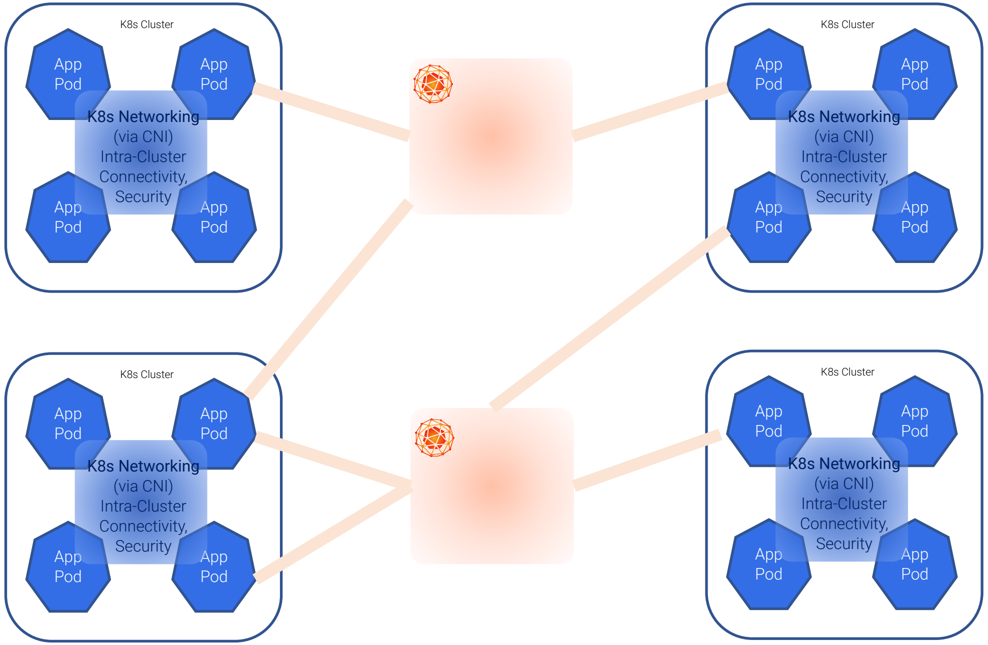
# State of the NSM

- CNCF Project
- **Multi-cloud CI**
- **Resiliencyv1 (AutoHealing)**: Can auto heal connections between Pods and Network Services if various system elements restart or NSE providing Network Service dies without disturbing client Pod.


AKS
EKS
GKE
Vanilla K8s
K8s Cluster
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security

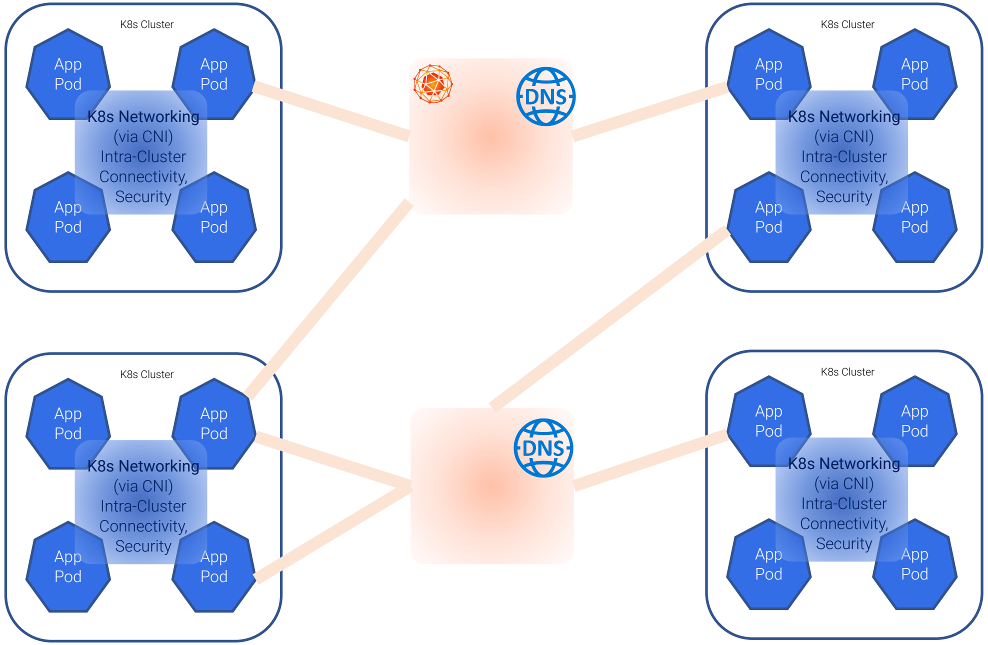
# State of the NSM

- CNCF Project
- Multi-cloud CI
- Resiliencyv1 (AutoHealing)
- **Inter-domain:** Initial Inter-domain support merged


K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

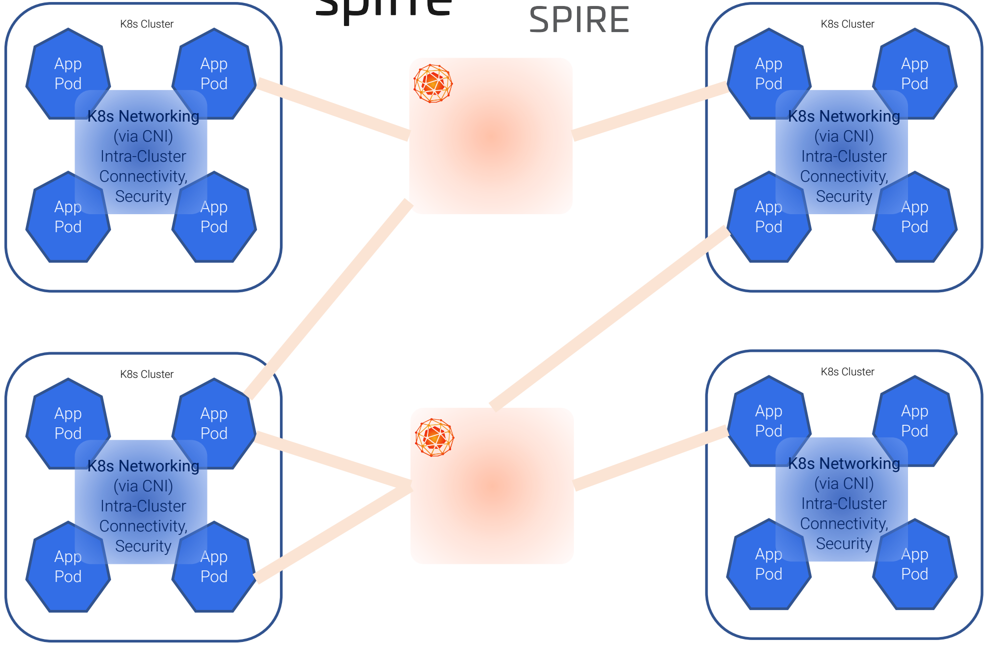
# State of the NSM

- CNCF Project
- Multi-cloud CI
- Resiliencyv1 (AutoHealing)
- Inter-domain
- **DNS:** Each Network Service (Connectivity Domain) can provide DNS to workload additivitly (ie: without breaking K8s DNS).


K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
DNS
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
DNS
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

# State of the NSM

- CNCF Project
- Multi-cloud CI
- Resiliencyv1 (AutoHealing)
- Inter-domain
- DNS
- **Security:** Spiffe/Spire based security – initial work done.

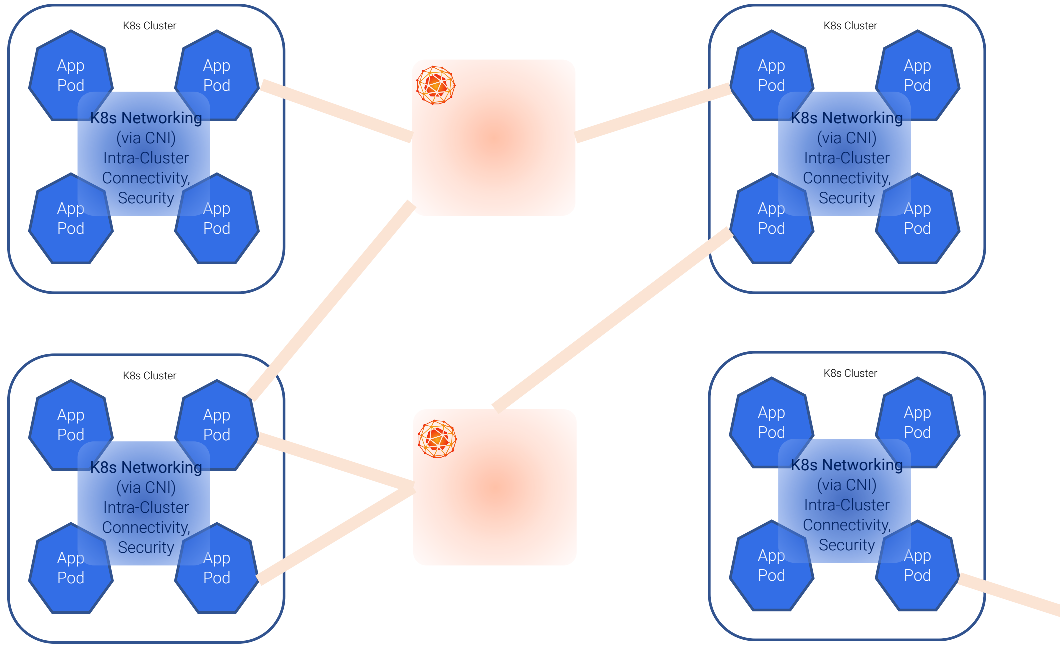

K8s Cluster
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
K8s Cluster
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
K8s Cluster
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
K8s Cluster
K8s Networking (via CNI) Intra-Cluster Connectivity, Security

Network Service Mesh

## NSM Future

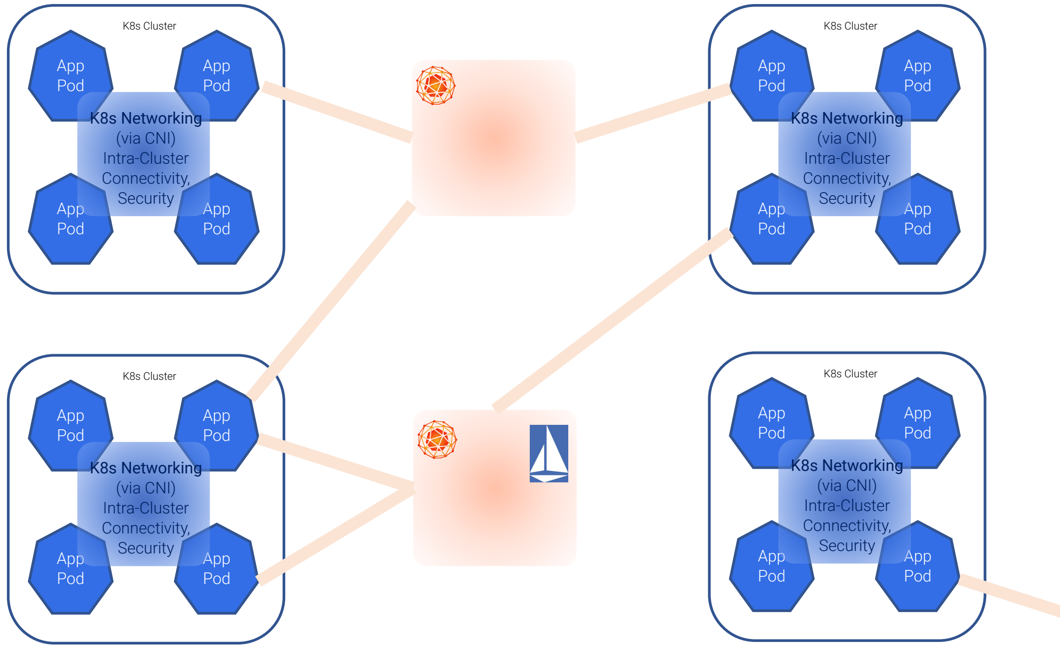
# Resiliencyv2

Autoheal connections from Client Pods to Network Service Endpoints (NSEs) they are connected to even if *all* non-client elements of the system restart simultaneously and NSE dies without impacting Client.


K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

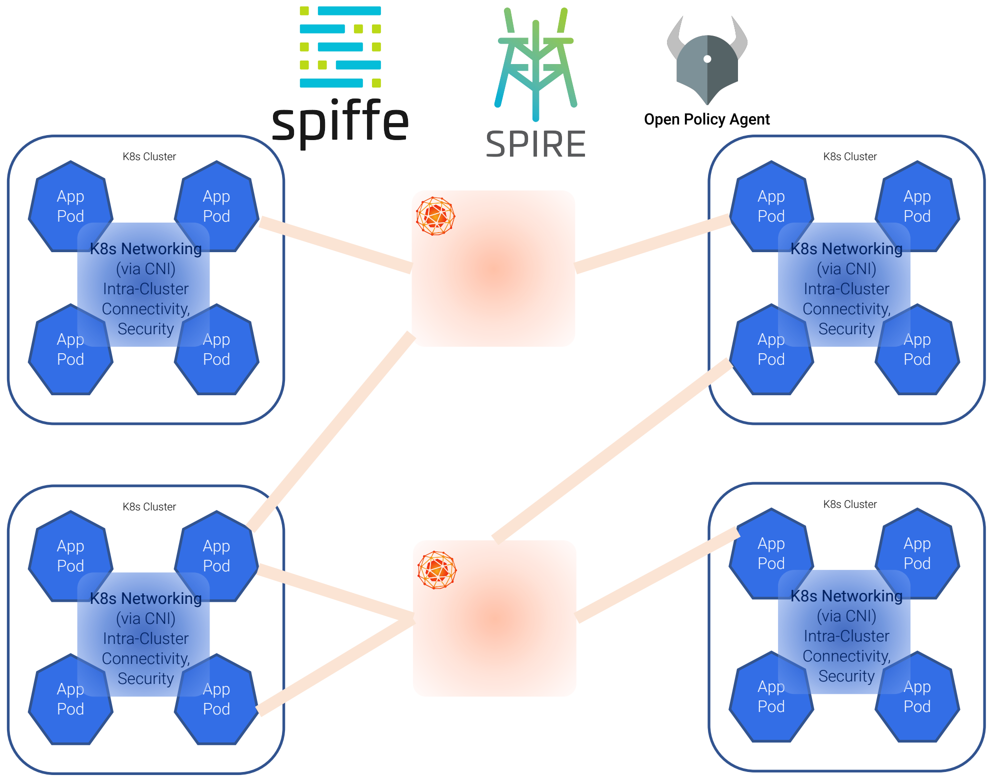
# Istio on NSM

Run an Istio domain over an NSM Network Service


K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

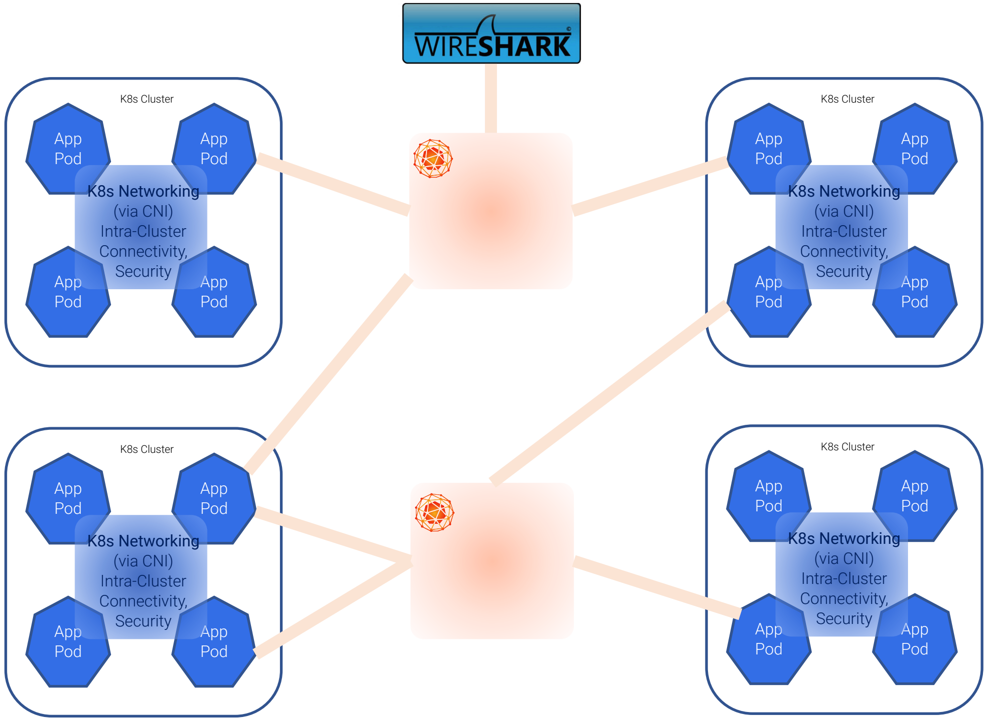
# Open Policy Agent: Authz

Using OPA to allow the Network Service Mesh to enforce admissions policy based on Spiffe/Spire identities


spiffe
SPIRE
Open Policy Agent
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod
K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

# Packet Capture Observability

Make it simple for Network Services (Connectivity Domains) to allow developers to securely get packet capture observability at per workload granularity.

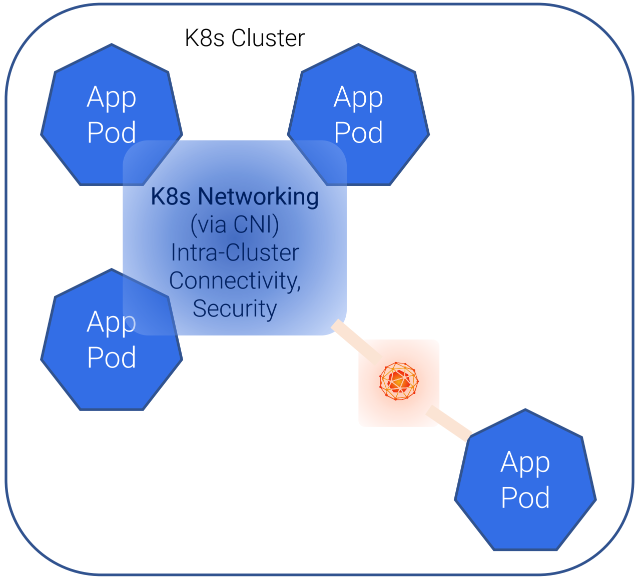

WIRESHARK
K8s Cluster
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security

# CNI Intercept

Allow (safe) insertion of Network Service between Pod and its CNI interface.

Would allow adding features to IntraCluster Networking with any CNI

Could be used for inserting Envoy Sidecar for Istio via NSM.

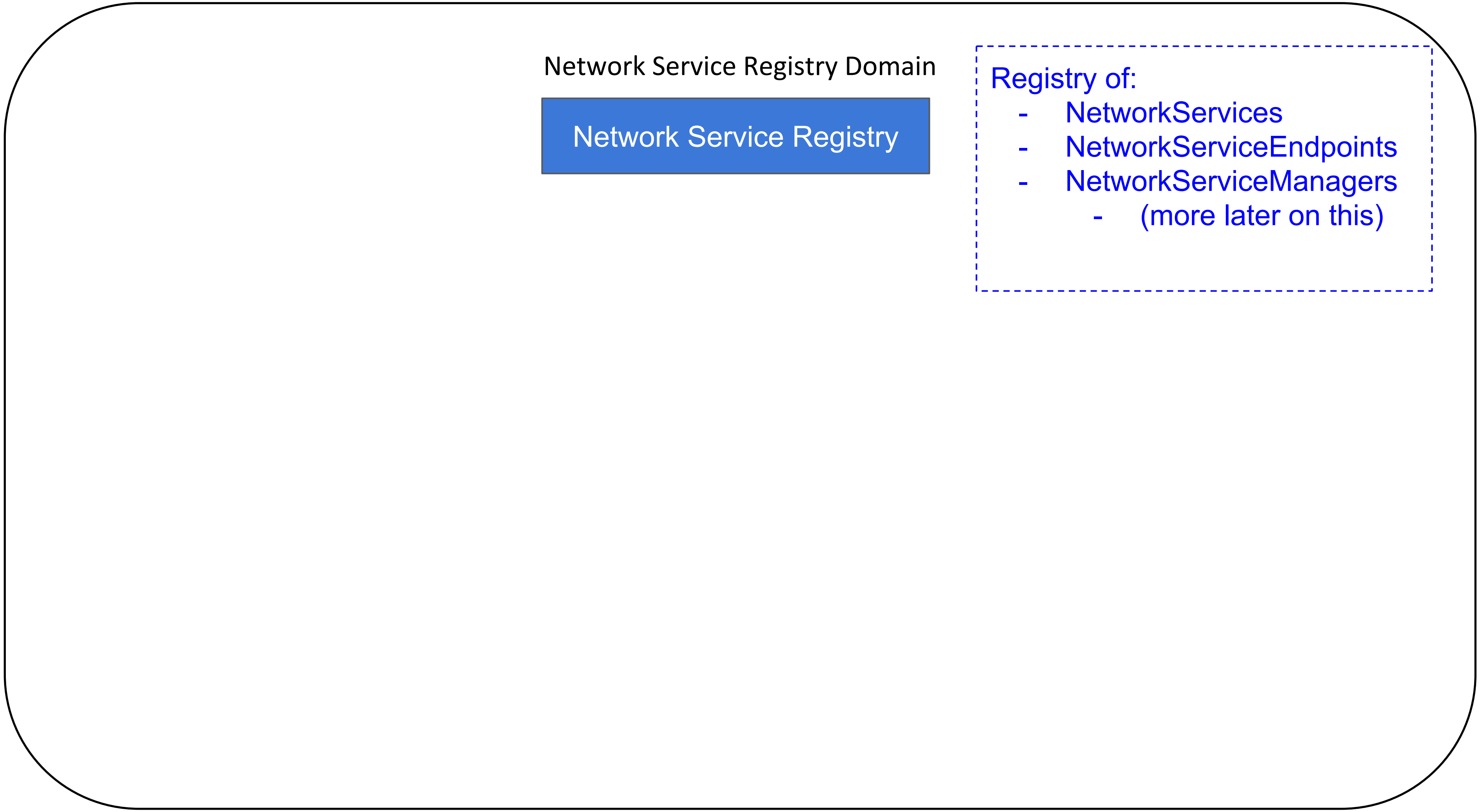

K8s Cluster
App Pod
App Pod
K8s Networking (via CNI) Intra-Cluster Connectivity, Security
App Pod
App Pod

# NSMCon @Kubecon NA

First NSMCon Nov 18 @KubeCon NA in San Diego

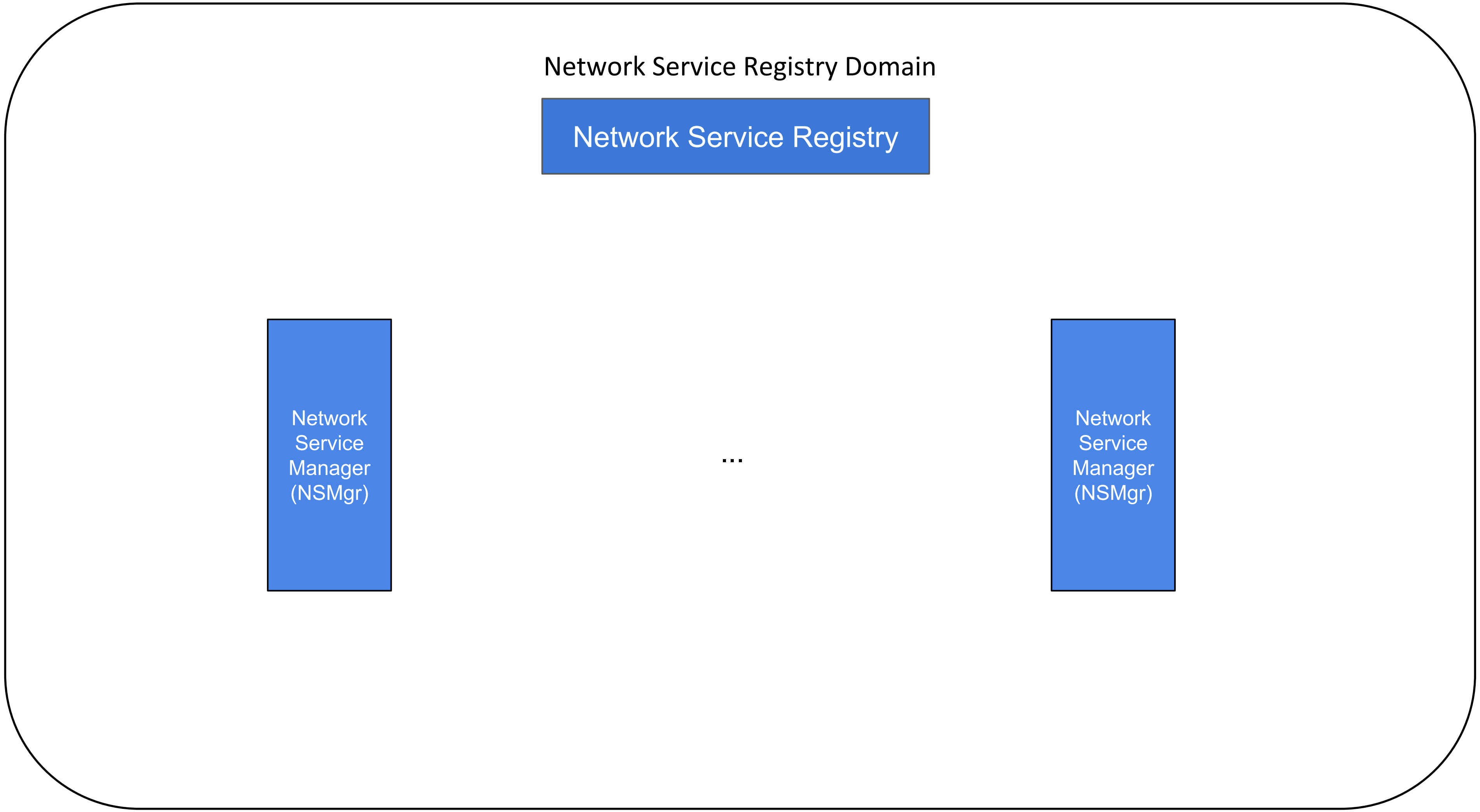
Network Service Mesh

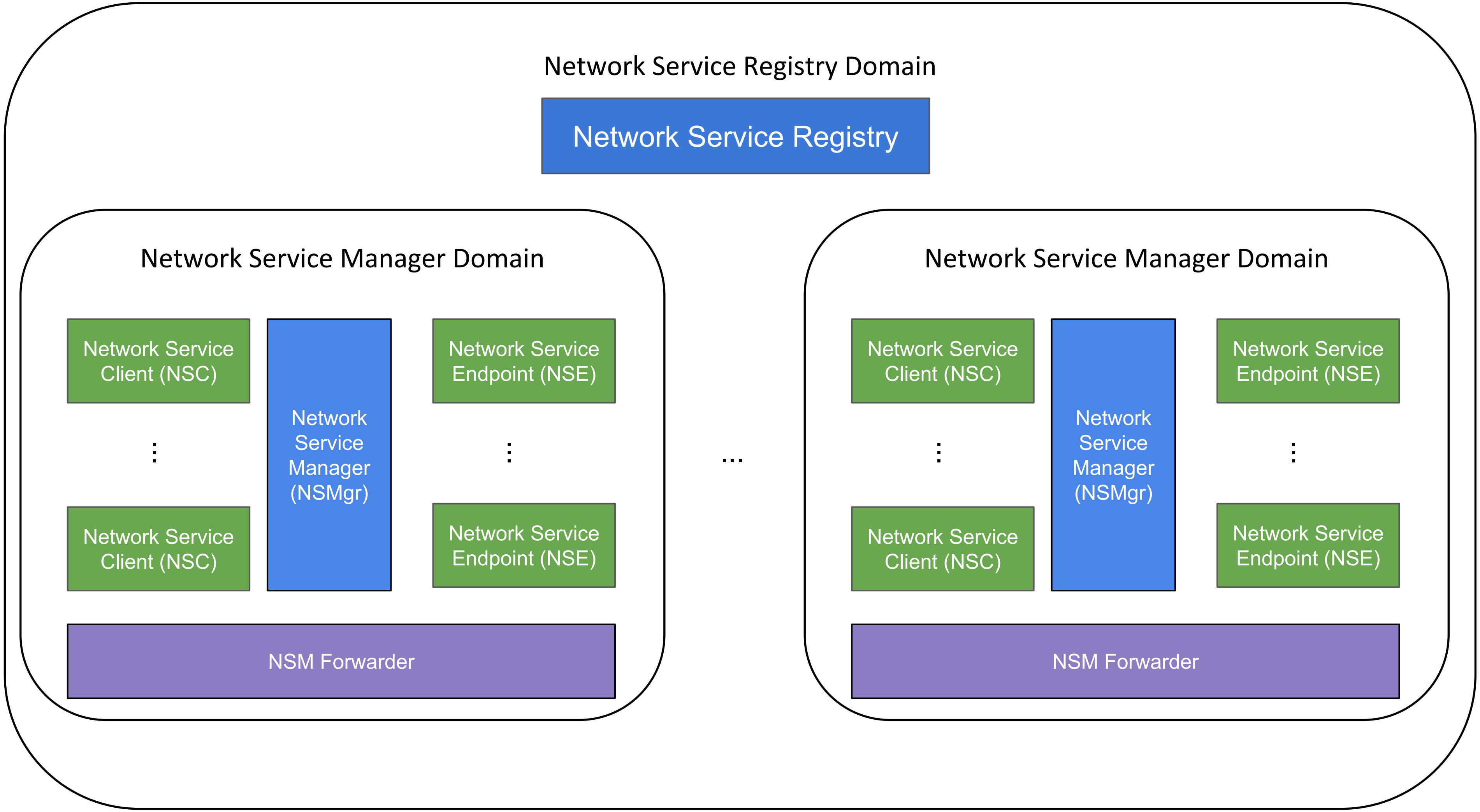
## How the Magic Works

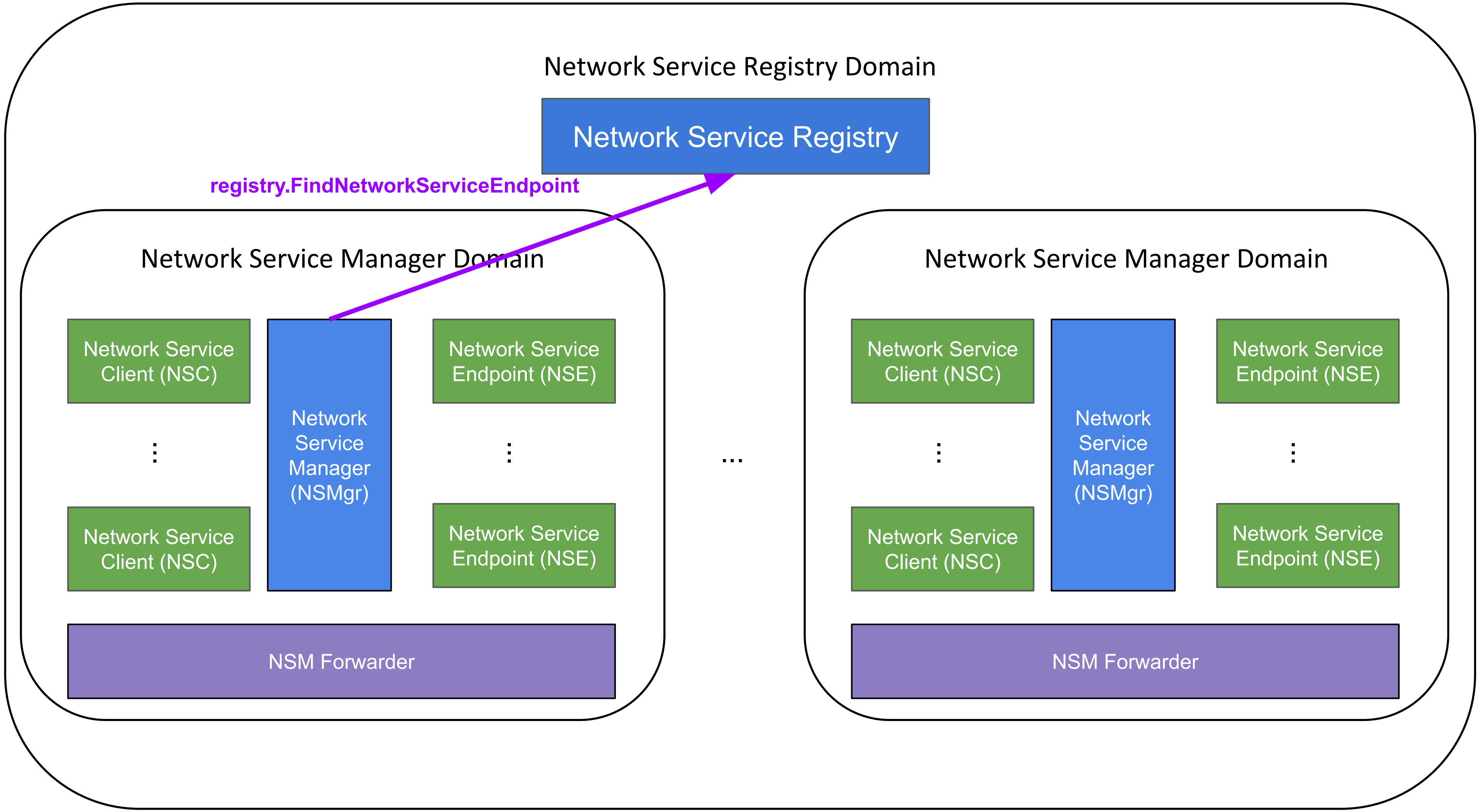


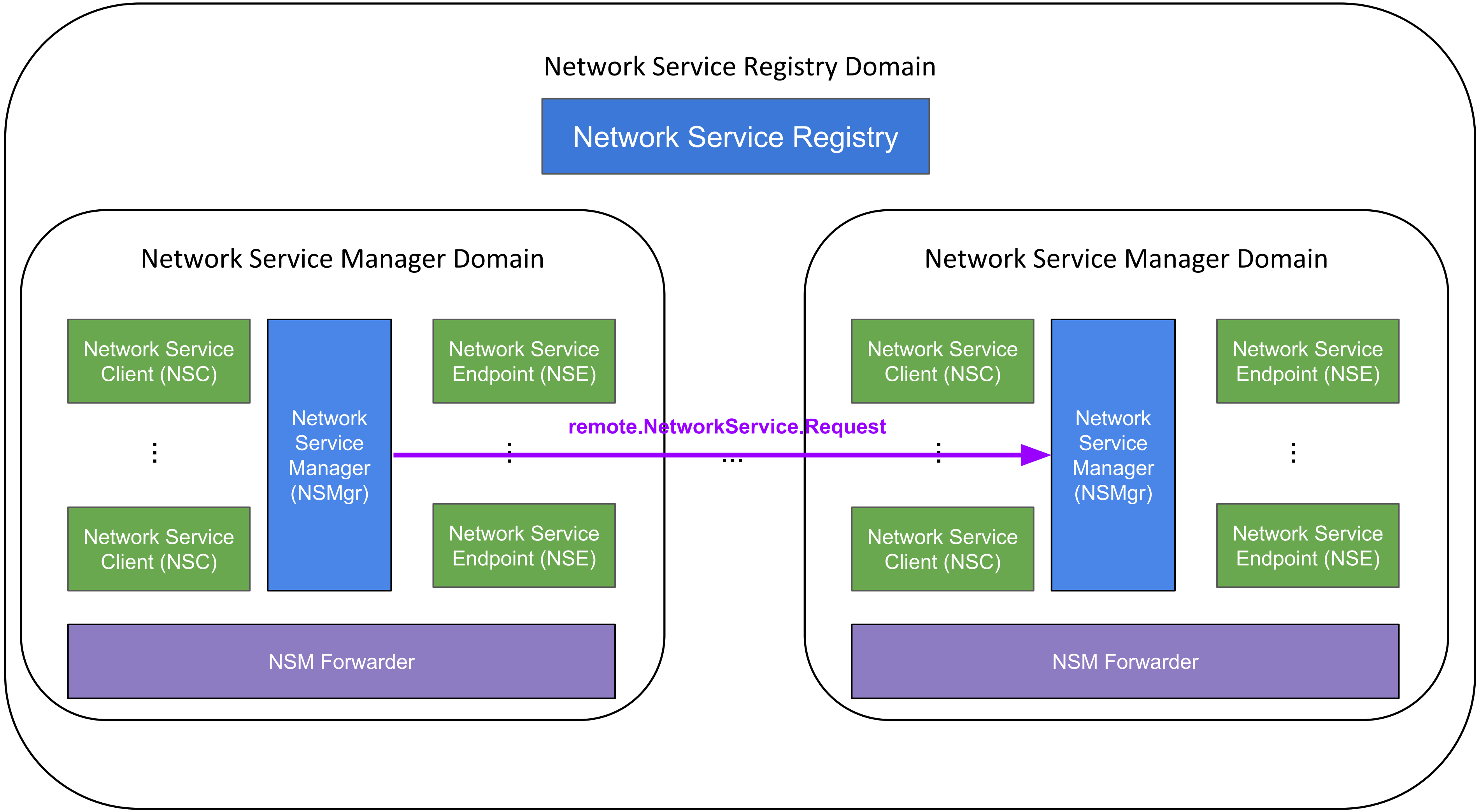
Registry of:

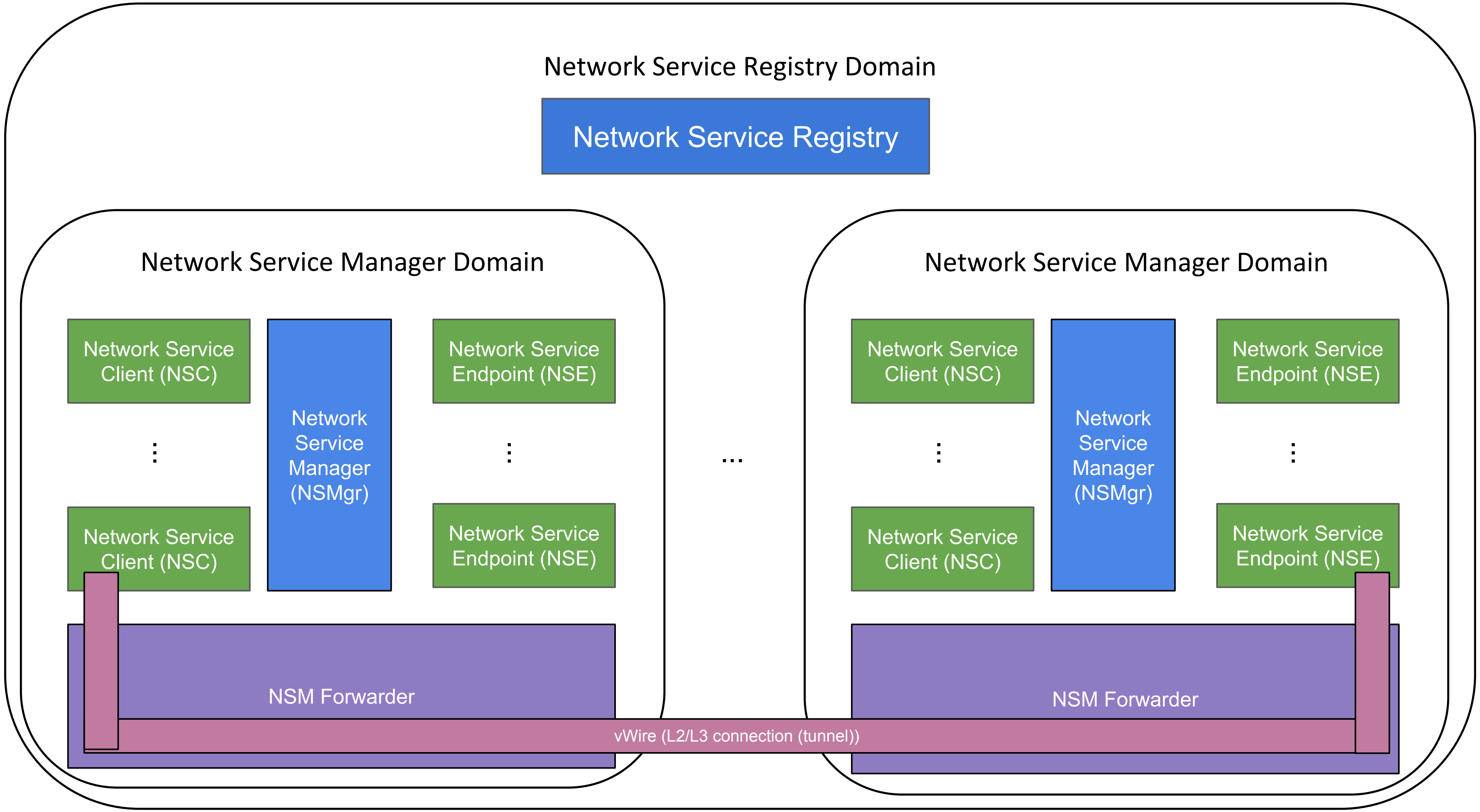
- NetworkServices
- NetworkServiceEndpoints
- NetworkServiceManagers
    - (more later on this)


Network Service Registry Domain
Network Service Registry
Network Service Manager (NSMgr)
...
Network Service Manager (NSMgr)


Network Service Registry Domain
Network Service Registry
Network Service Manager Domain
Network Service Client (NSC)
Network Service Manager (NSMgr)
Network Service Endpoint (NSE)
Network Service Client (NSC)
Network Service Endpoint (NSE)
NSM Forwarder
...
Network Service Manager Domain
Network Service Client (NSC)
Network Service Manager (NSMgr)
Network Service Endpoint (NSE)
Network Service Client (NSC)
Network Service Endpoint (NSE)
NSM Forwarder


Network Service Registry Domain
Network Service Registry
registry.FindNetworkServiceEndpoint
Network Service Manager Domain
Network Service Client (NSC)
Network Service Manager (NSMgr)
Network Service Endpoint (NSE)
Network Service Client (NSC)
Network Service Endpoint (NSE)
NSM Forwarder
...
Network Service Manager Domain
Network Service Client (NSC)
Network Service Manager (NSMgr)
Network Service Endpoint (NSE)
Network Service Client (NSC)
Network Service Endpoint (NSE)
NSM Forwarder

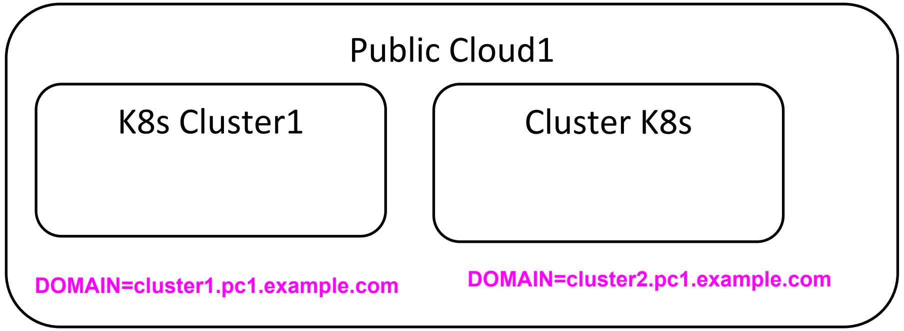

Network Service Registry Domain
Network Service Registry
Network Service Manager Domain
Network Service Client (NSC)
Network Service Client (NSC)
Network Service Manager (NSMgr)
Network Service Endpoint (NSE)
Network Service Endpoint (NSE)
NSM Forwarder
remote.NetworkService.Request
...
Network Service Manager Domain
Network Service Client (NSC)
Network Service Client (NSC)
Network Service Manager (NSMgr)
Network Service Endpoint (NSE)
Network Service Endpoint (NSE)
NSM Forwarder


Network Service Registry Domain
Network Service Registry
Network Service Manager Domain
Network Service Client (NSC)
Network Service Manager (NSMgr)
Network Service Endpoint (NSE)
Network Service Client (NSC)
Network Service Endpoint (NSE)
NSM Forwarder
...
Network Service Manager Domain
Network Service Client (NSC)
Network Service Manager (NSMgr)
Network Service Endpoint (NSE)
Network Service Client (NSC)
Network Service Endpoint (NSE)
NSM Forwarder
vWire (L2/L3 connection (tunnel))

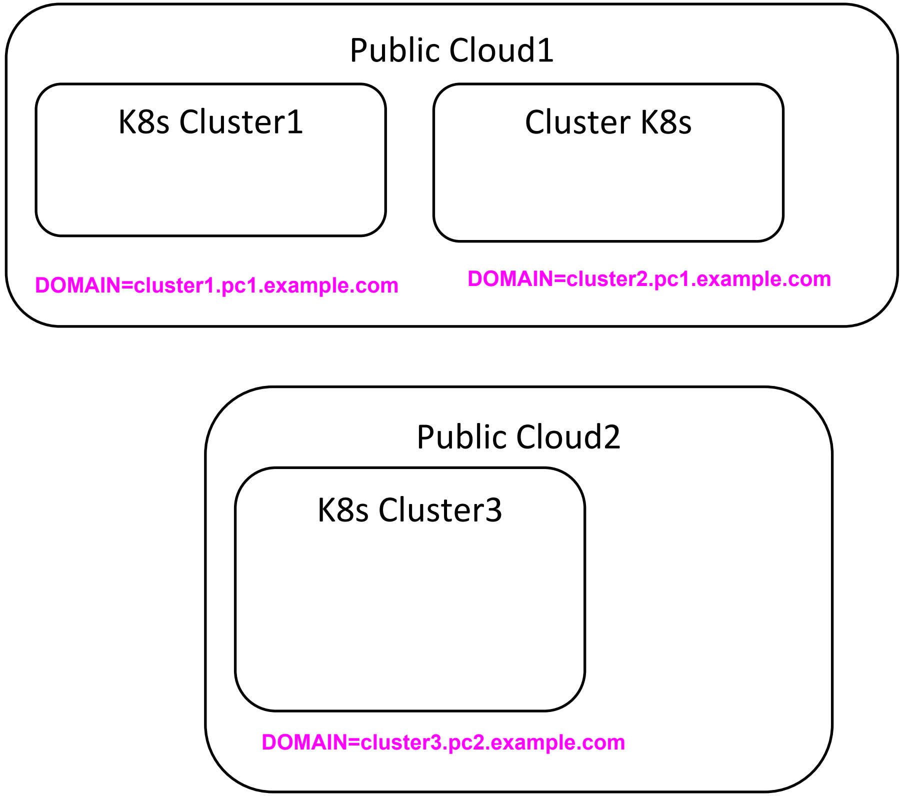
Network Service Mesh

## Interdomain

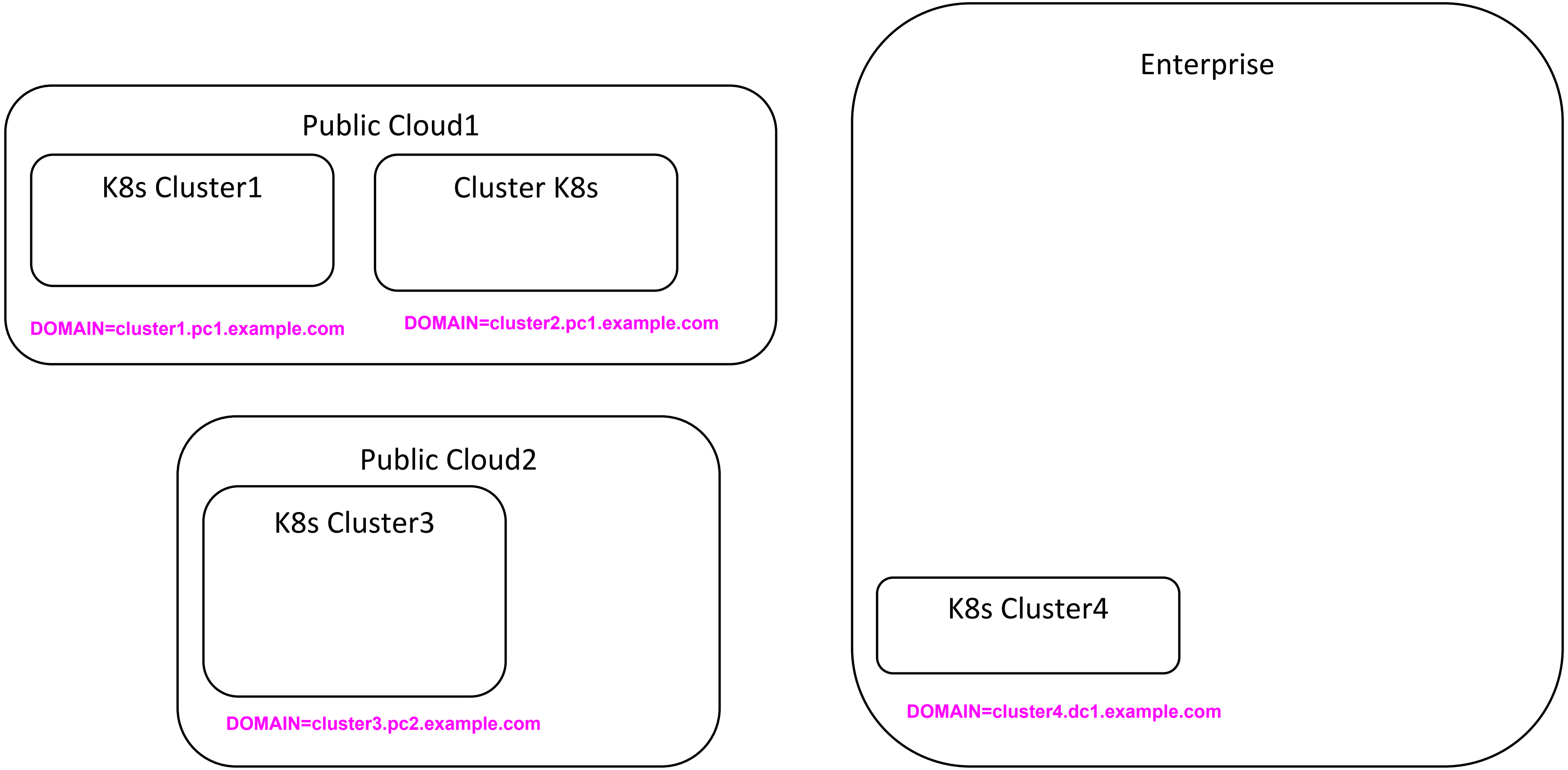
# Examples of 'Domains'


Public Cloud1
K8s Cluster1
Cluster K8s
DOMAIN=cluster1.pc1.example.com
DOMAIN=cluster2.pc1.example.com

# Examples of ‘Domains’

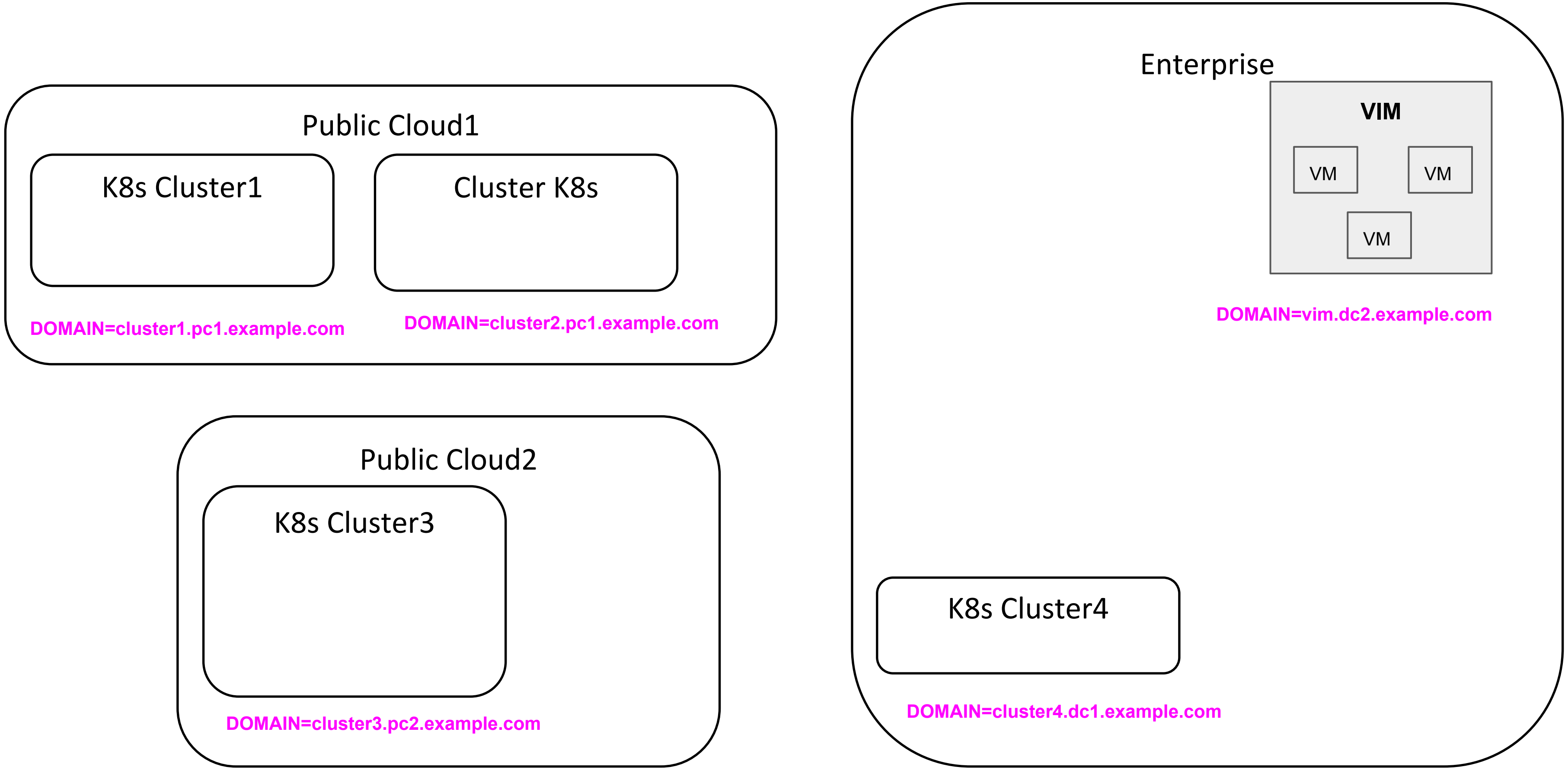

Public Cloud1
K8s Cluster1
Cluster K8s
DOMAIN=cluster1.pc1.example.com
DOMAIN=cluster2.pc1.example.com
Public Cloud2
K8s Cluster3
DOMAIN=cluster3.pc2.example.com

# Examples of ‘Domains’

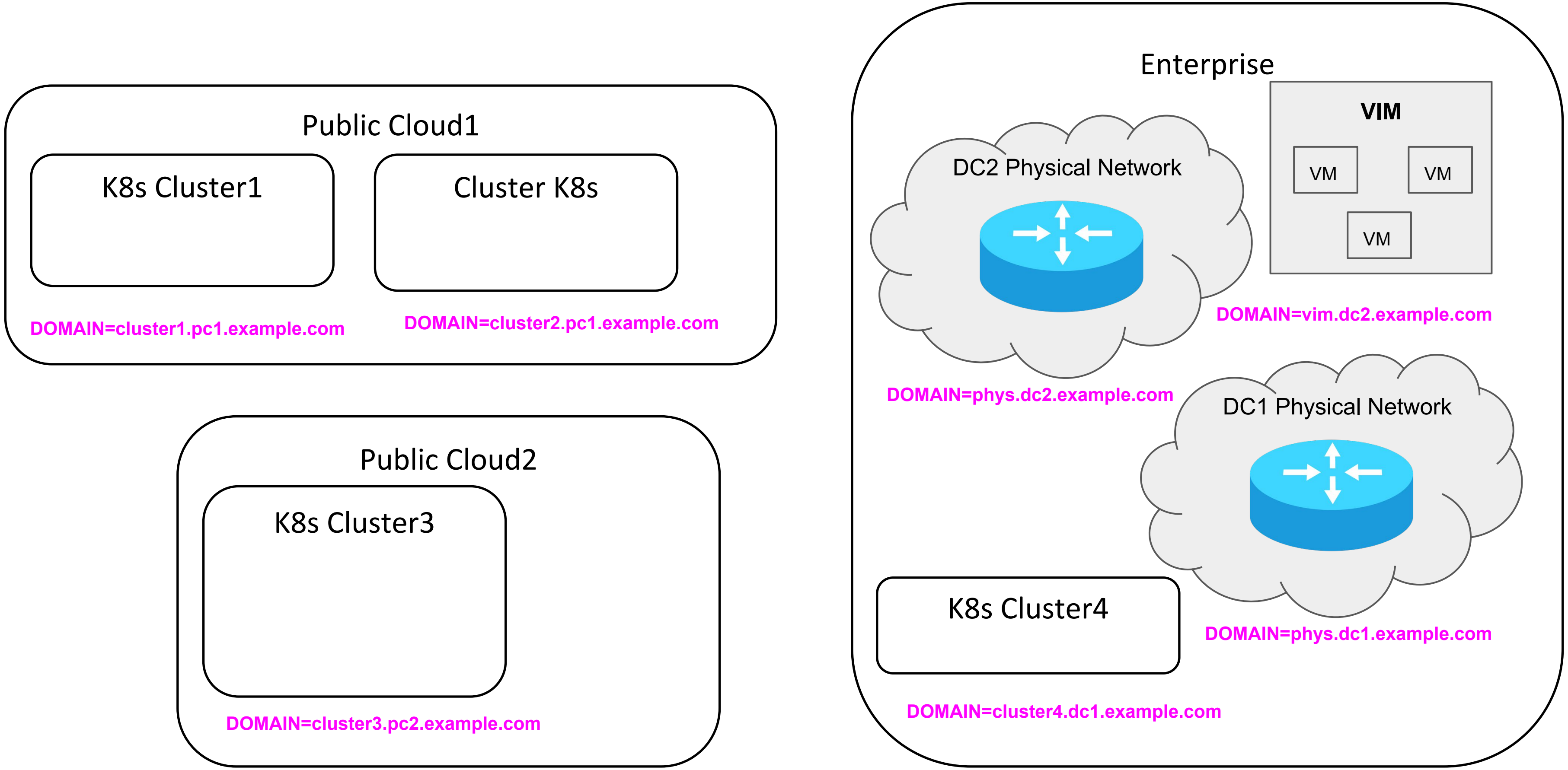


Enterprise
Public Cloud1
K8s Cluster1
Cluster K8s
DOMAIN=cluster1.pc1.example.com
DOMAIN=cluster2.pc1.example.com
Public Cloud2
K8s Cluster3
DOMAIN=cluster3.pc2.example.com
K8s Cluster4
DOMAIN=cluster4.dc1.example.com

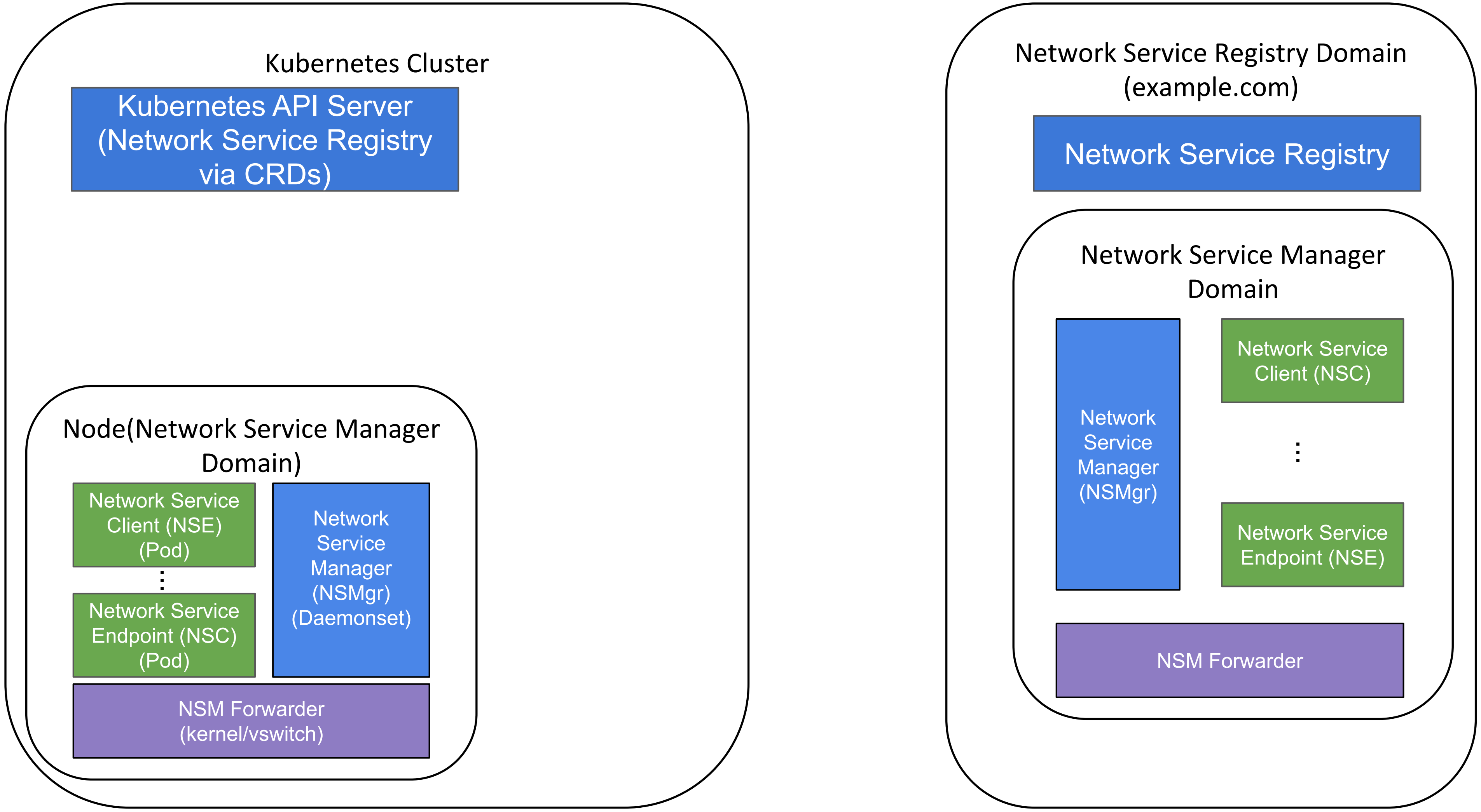
# Examples of ‘Domains’

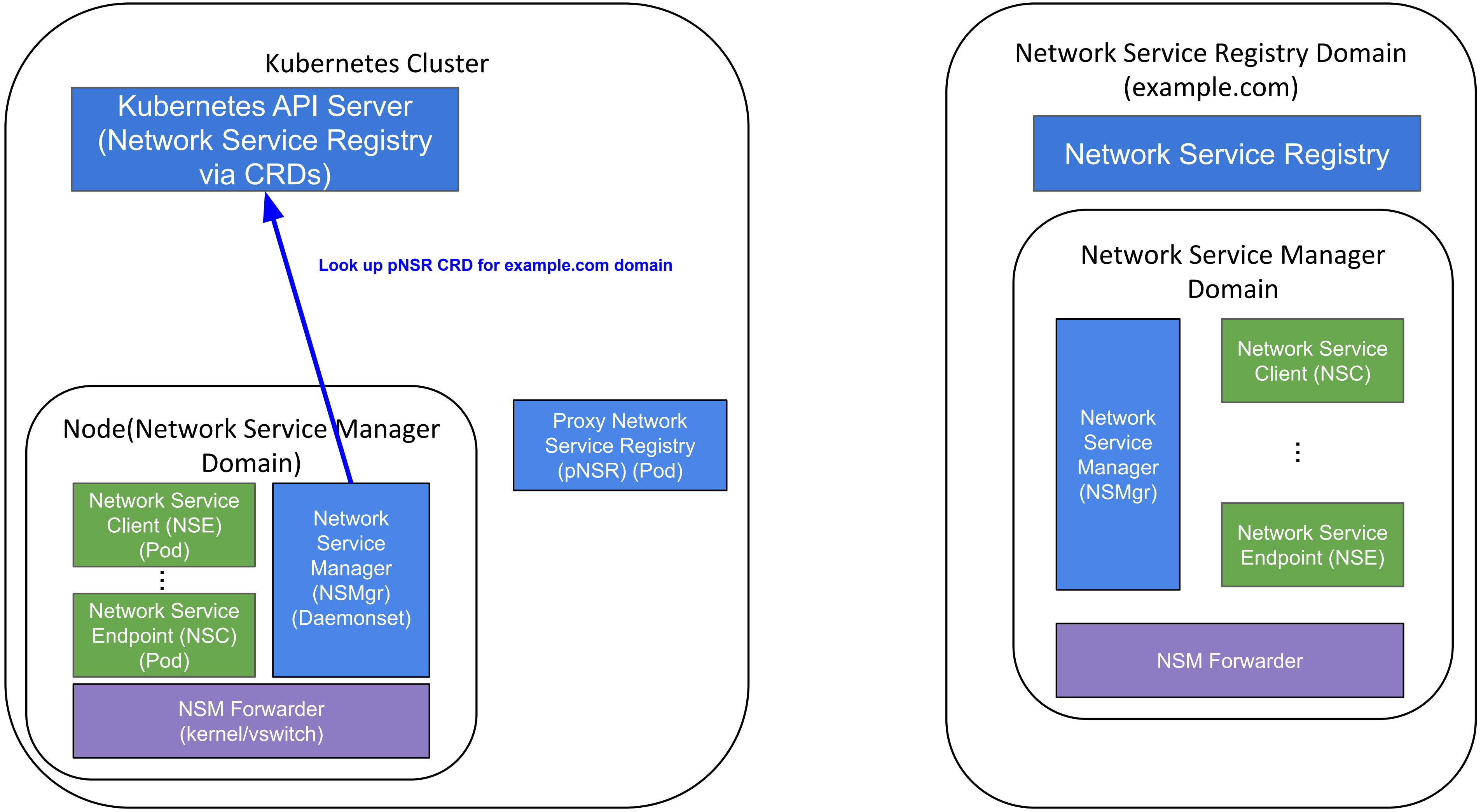

Public Cloud1
K8s Cluster1
Cluster K8s
DOMAIN=cluster1.pc1.example.com
DOMAIN=cluster2.pc1.example.com
Public Cloud2
K8s Cluster3
DOMAIN=cluster3.pc2.example.com
Enterprise
VIM
VM
VM
VM
DOMAIN=vim.dc2.example.com
K8s Cluster4
DOMAIN=cluster4.dc1.example.com

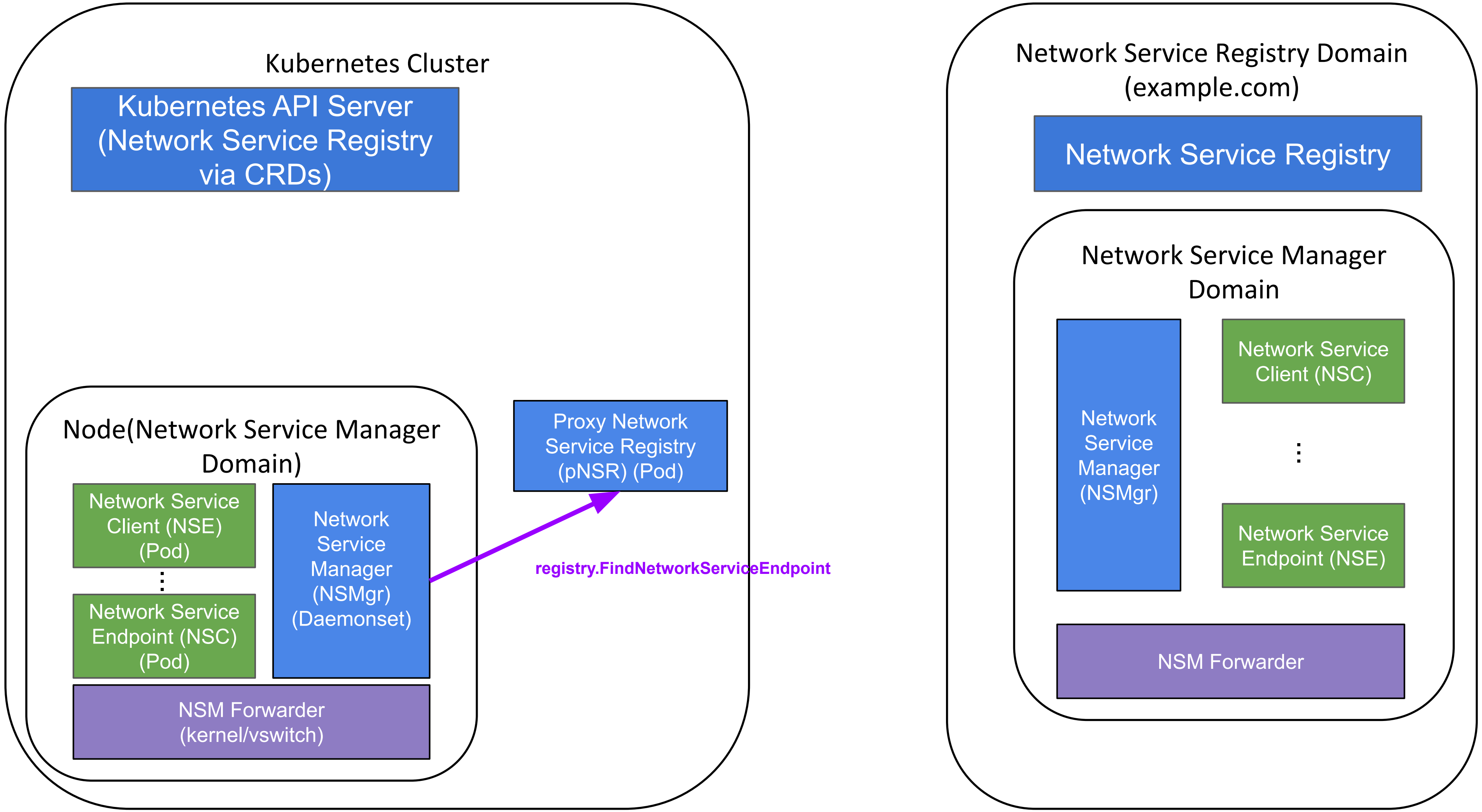
# Examples of ‘Domains’

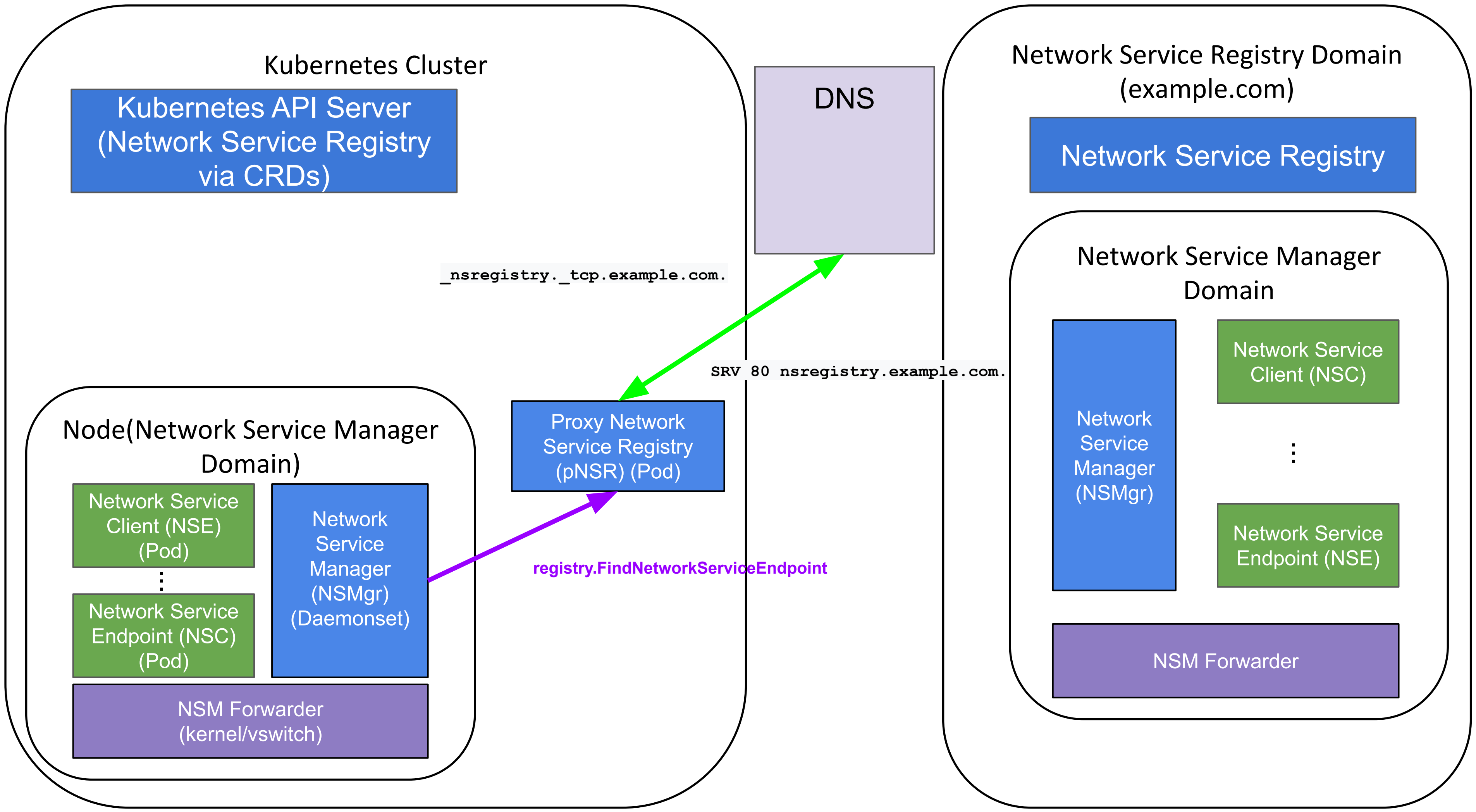


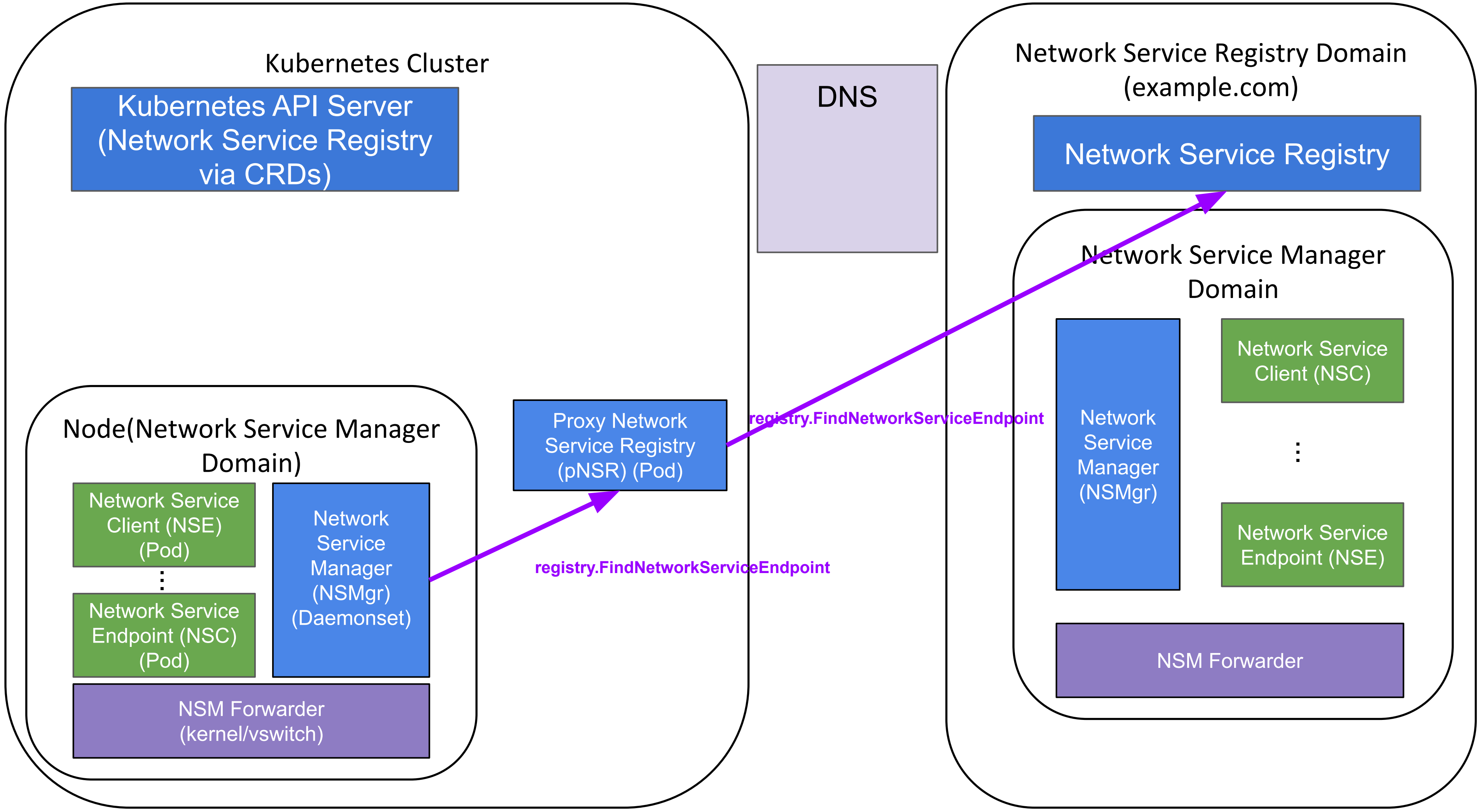
Public Cloud1
K8s Cluster1
Cluster K8s
DOMAIN=cluster1.pc1.example.com
DOMAIN=cluster2.pc1.example.com
Public Cloud2
K8s Cluster3
DOMAIN=cluster3.pc2.example.com
Enterprise
DC2 Physical Network
VIM
VM
VM
VM
DOMAIN=vim.dc2.example.com
DOMAIN=phys.dc2.example.com
DC1 Physical Network
K8s Cluster4
DOMAIN=phys.dc1.example.com
DOMAIN=cluster4.dc1.example.com

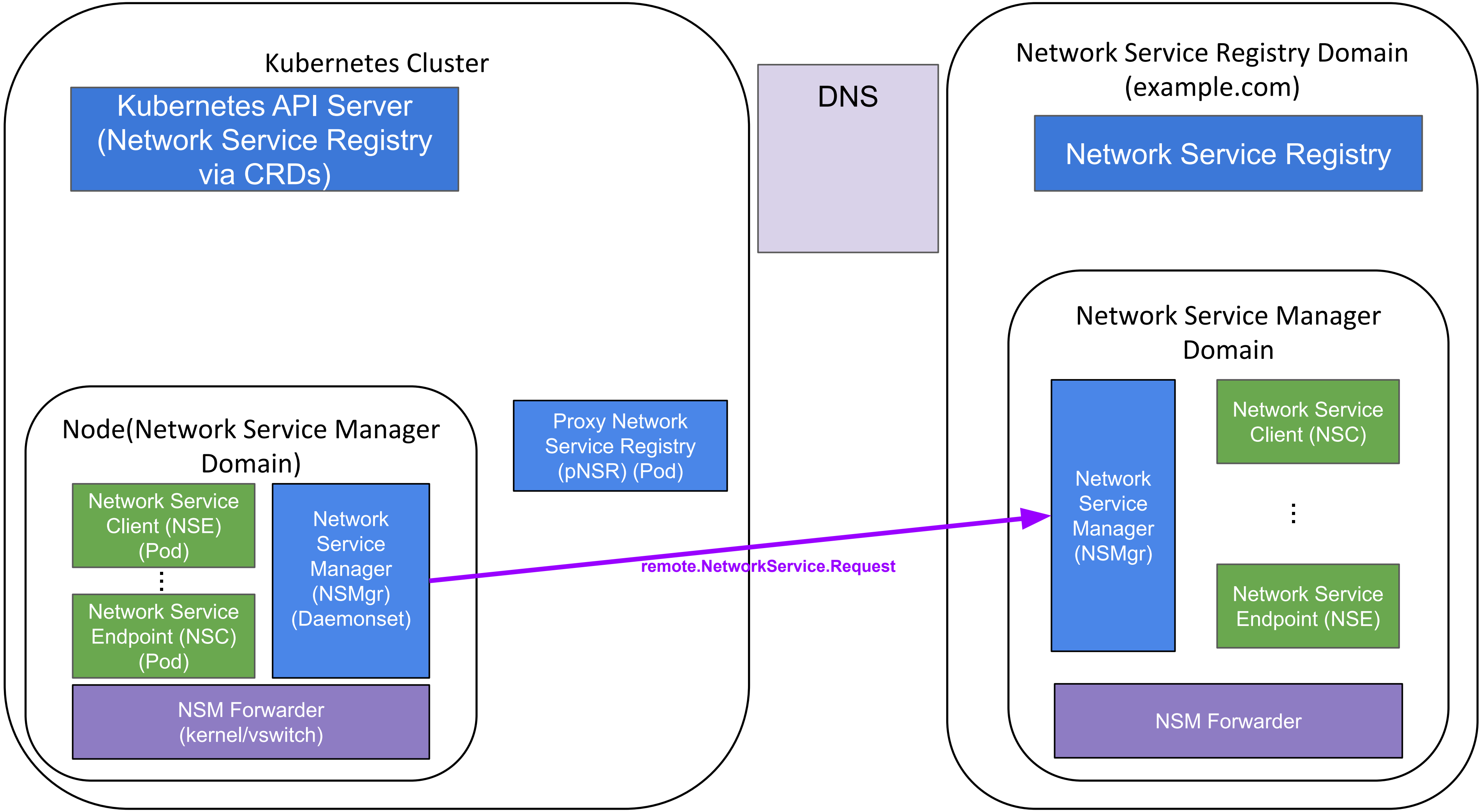

Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
Node(Network Service Manager Domain)
Network Service Client (NSE) (Pod)
⋮
Network Service Endpoint (NSC) (Pod)
Network Service Manager (NSMgr) (Daemonset)
NSM Forwarder (kernel/vswitch)
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Network Service Manager (NSMgr)
Network Service Client (NSC)
⋮
Network Service Endpoint (NSE)
NSM Forwarder

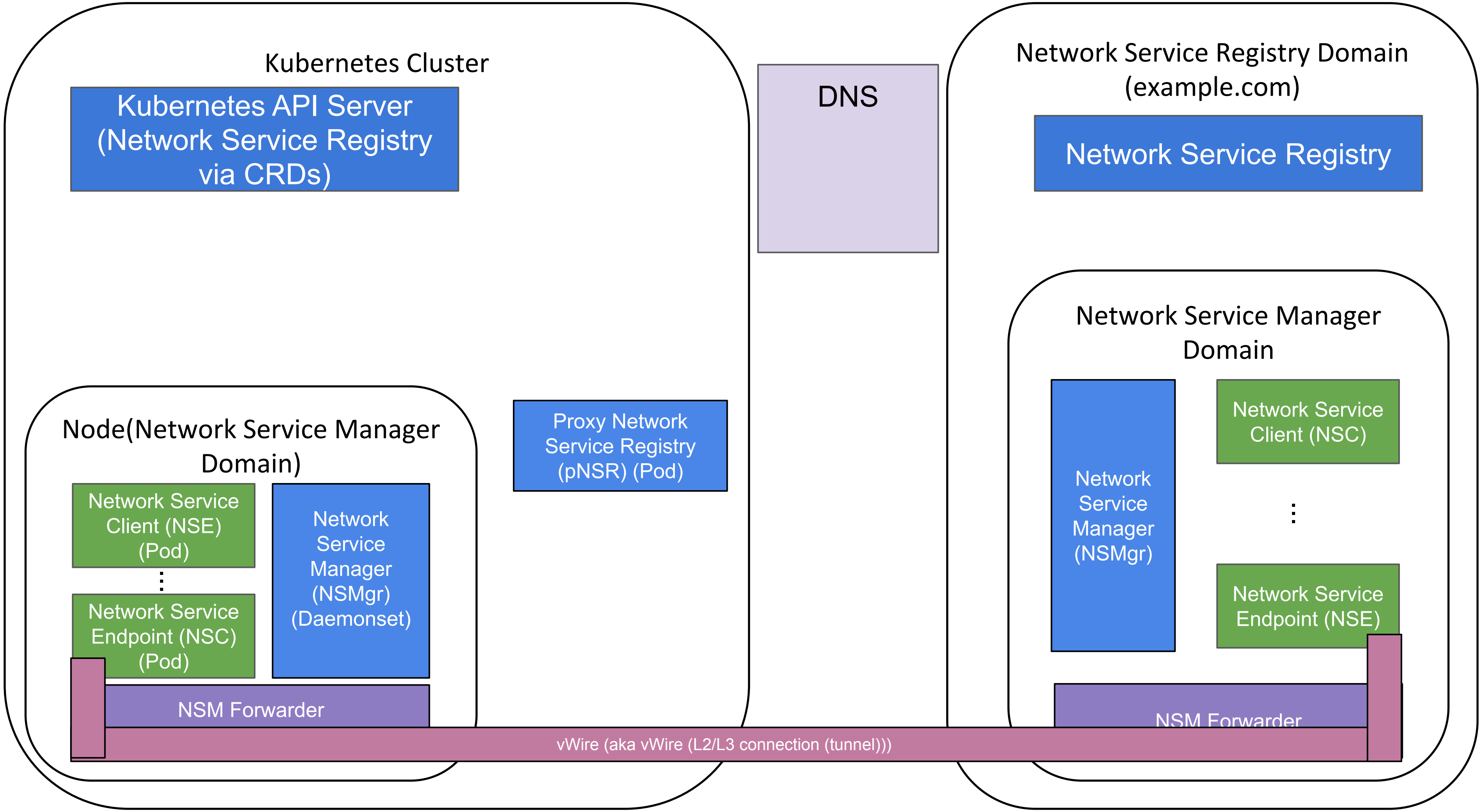

Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
Look up pNSR CRD for example.com domain
Node(Network Service Manager Domain)
Network Service Client (NSE) (Pod)
⋮
Network Service Endpoint (NSC) (Pod)
Network Service Manager (NSMgr) (Daemonset)
NSM Forwarder (kernel/vswitch)
Proxy Network Service Registry (pNSR) (Pod)
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Network Service Manager (NSMgr)
Network Service Client (NSC)
⋮
Network Service Endpoint (NSE)
NSM Forwarder


Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
Node(Network Service Manager Domain)
Network Service Client (NSE) (Pod)
⋮
Network Service Endpoint (NSC) (Pod)
Network Service Manager (NSMgr) (Daemonset)
NSM Forwarder (kernel/vswitch)
Proxy Network Service Registry (pNSR) (Pod)
registry.FindNetworkServiceEndpoint
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Network Service Manager (NSMgr)
Network Service Client (NSC)
⋮
Network Service Endpoint (NSE)
NSM Forwarder


Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
DNS
Network Service Registry Domain (example.com)
Network Service Registry
_nsregistry._tcp.example.com.
Network Service Manager Domain
SRV 80 nsregistry.example.com.
Network Service Client (NSC)
Network Service Manager (NSMgr)
Node(Network Service Manager Domain)
Proxy Network Service Registry (pNSR) (Pod)
Network Service Client (NSE) (Pod)
Network Service Manager (NSMgr) (Daemonset)
Network Service Endpoint (NSE)
registry.FindNetworkServiceEndpoint
Network Service Endpoint (NSC) (Pod)
NSM Forwarder
NSM Forwarder (kernel/vswitch)

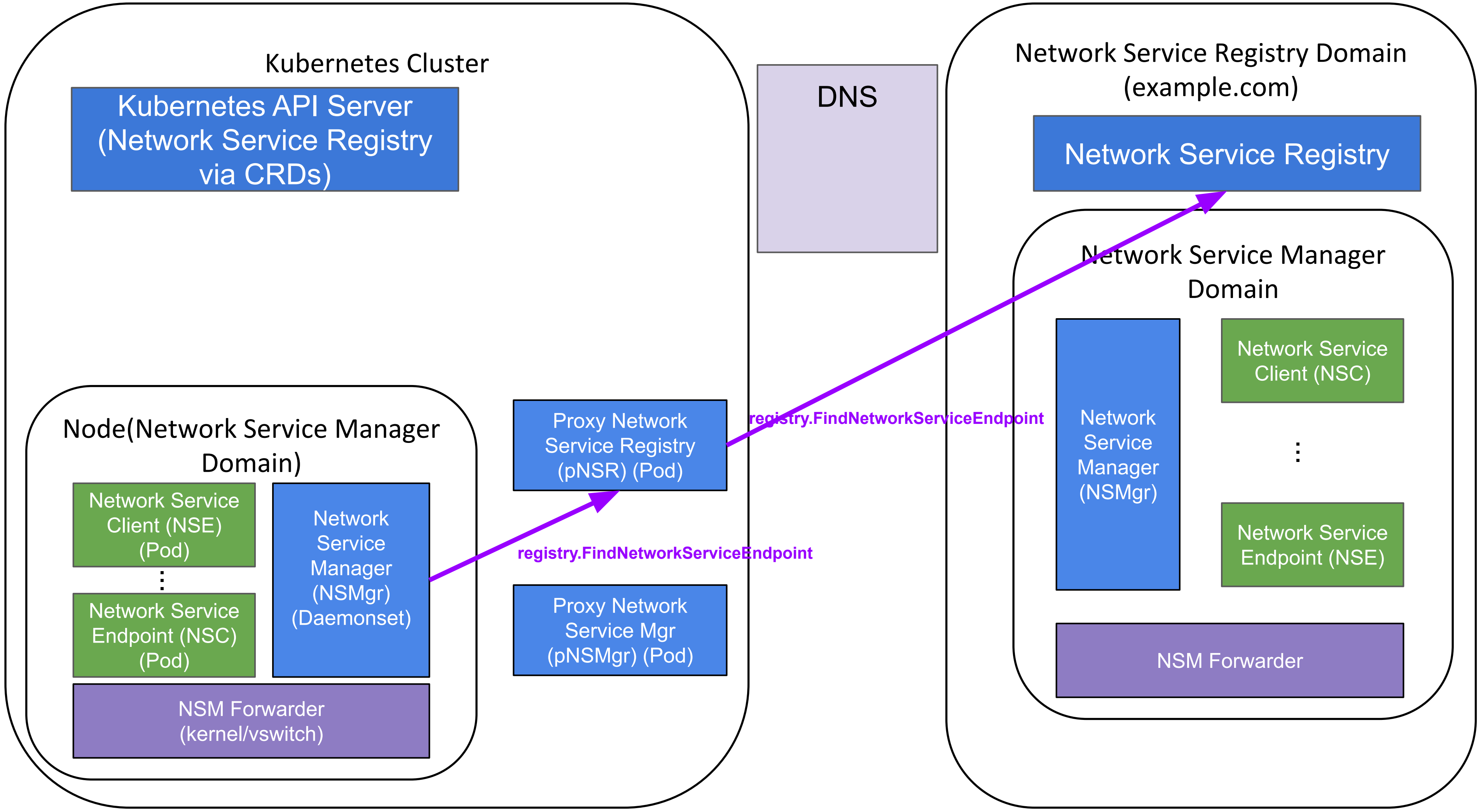

Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
DNS
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Network Service Manager (NSMgr)
Network Service Client (NSC)
⋮
Network Service Endpoint (NSE)
NSM Forwarder
registry.FindNetworkServiceEndpoint
Proxy Network Service Registry (pNSR) (Pod)
Node(Network Service Manager Domain)
Network Service Client (NSE) (Pod)
⋮
Network Service Endpoint (NSC) (Pod)
Network Service Manager (NSMgr) (Daemonset)
registry.FindNetworkServiceEndpoint
NSM Forwarder (kernel/vswitch)

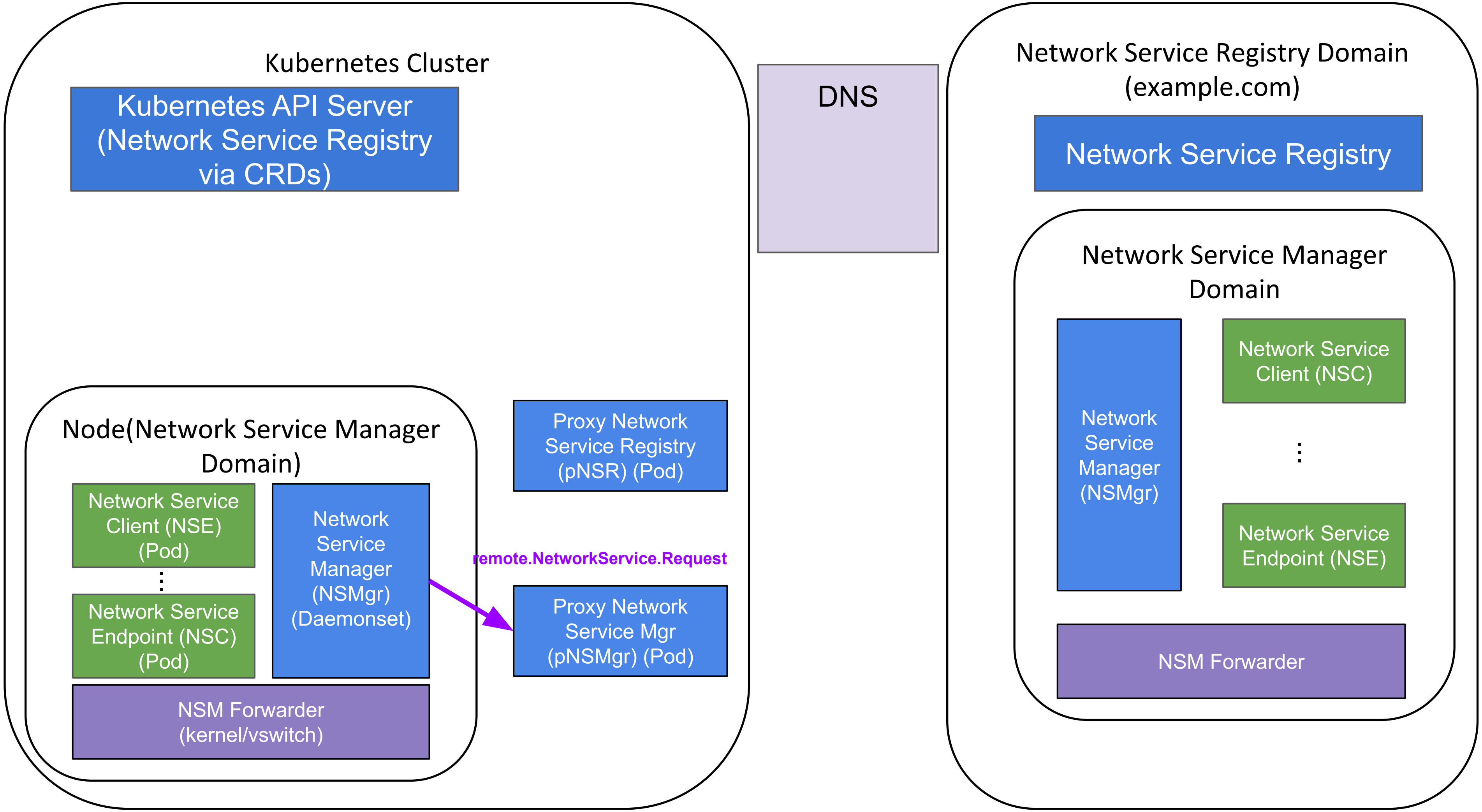

Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
Node(Network Service Manager Domain)
Network Service Client (NSE) (Pod)
⋮
Network Service Endpoint (NSC) (Pod)
Network Service Manager (NSMgr) (Daemonset)
NSM Forwarder (kernel/vswitch)
Proxy Network Service Registry (pNSR) (Pod)
DNS
remote.NetworkService.Request
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Network Service Manager (NSMgr)
Network Service Client (NSC)
⋮
Network Service Endpoint (NSE)
NSM Forwarder

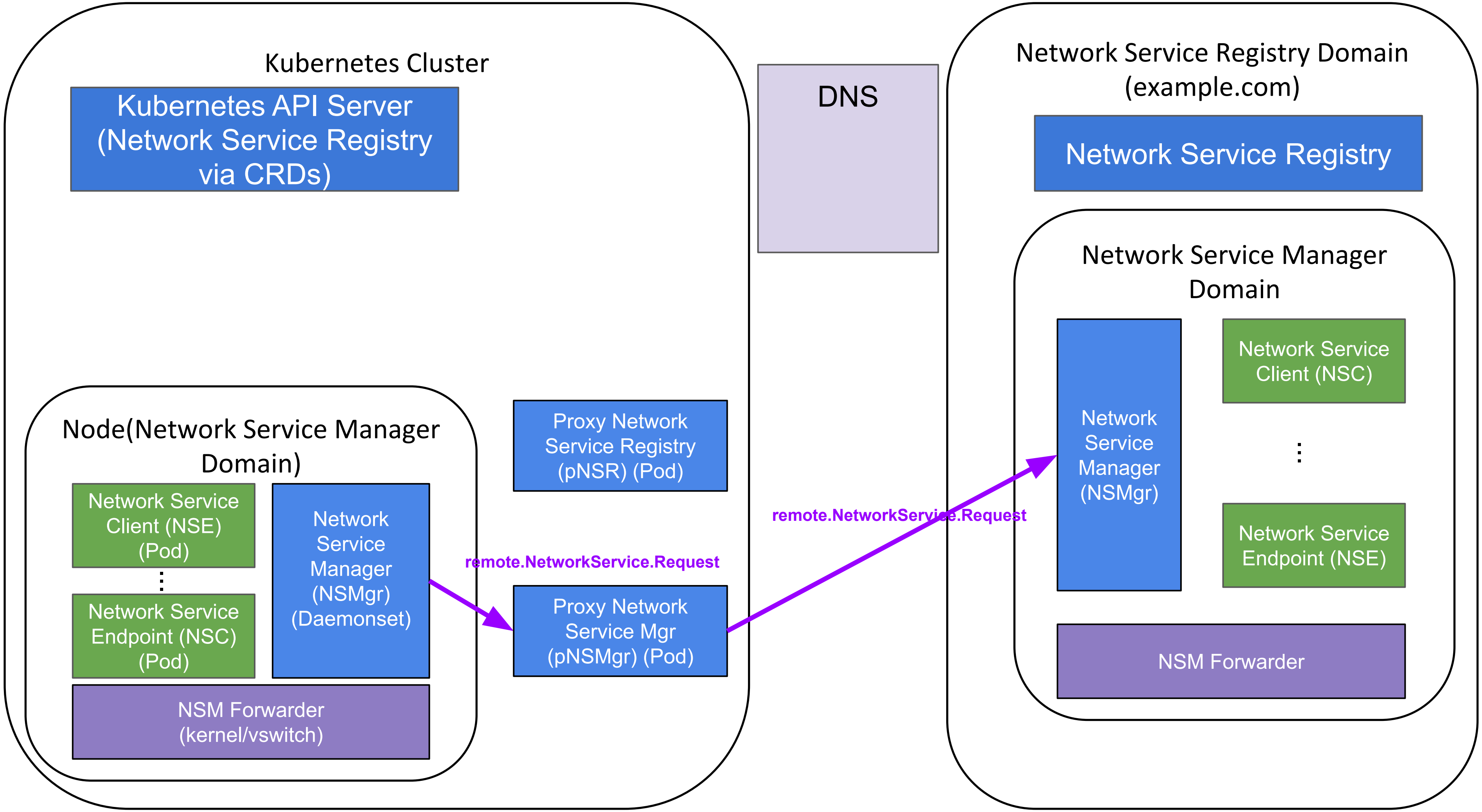

Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
DNS
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Node(Network Service Manager Domain)
Proxy Network Service Registry (pNSR) (Pod)
Network Service Client (NSE) (Pod)
Network Service Endpoint (NSC) (Pod)
Network Service Manager (NSMgr) (Daemonset)
Network Service Manager (NSMgr)
Network Service Client (NSC)
Network Service Endpoint (NSE)
NSM Forwarder
NSM Forwarder
vWire (aka vWire (L2/L3 connection (tunnel)))

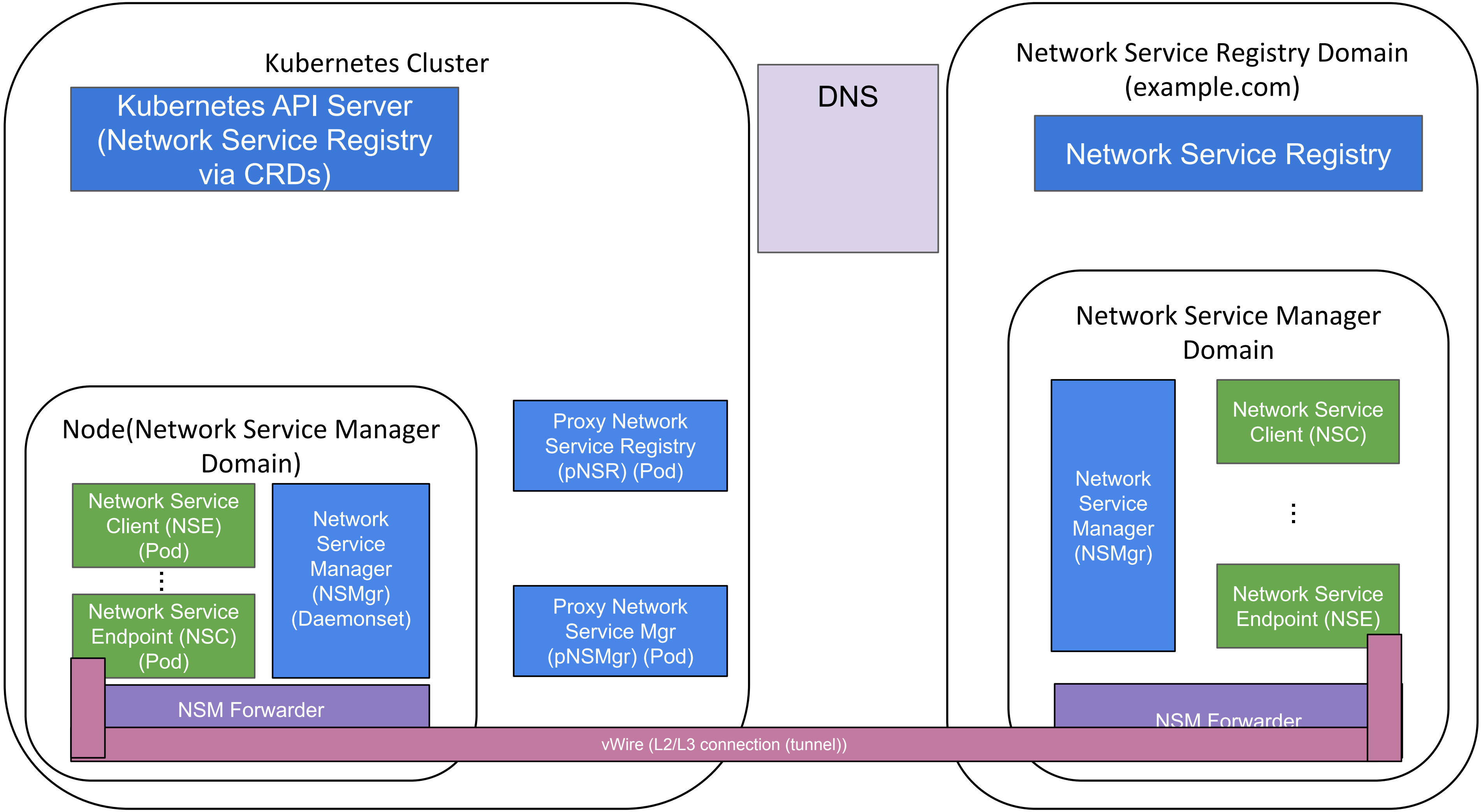
Network Service Mesh

## Interdomain w/pNSMgr


Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
Node(Network Service Manager Domain)
Network Service Client (NSE) (Pod)
⋮
Network Service Endpoint (NSC) (Pod)
Network Service Manager (NSMgr) (Daemonset)
NSM Forwarder (kernel/vswitch)
Proxy Network Service Registry (pNSR) (Pod)
Proxy Network Service Mgr (pNSMgr) (Pod)
registry.FindNetworkServiceEndpoint
registry.FindNetworkServiceEndpoint
DNS
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Network Service Manager (NSMgr)
Network Service Client (NSC)
⋮
Network Service Endpoint (NSE)
NSM Forwarder


Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
Node(Network Service Manager Domain)
Network Service Client (NSE) (Pod)
⋮
Network Service Endpoint (NSC) (Pod)
Network Service Manager (NSMgr) (Daemonset)
NSM Forwarder (kernel/vswitch)
Proxy Network Service Registry (pNSR) (Pod)
remote.NetworkService.Request
Proxy Network Service Mgr (pNSMgr) (Pod)
DNS
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Network Service Manager (NSMgr)
Network Service Client (NSC)
⋮
Network Service Endpoint (NSE)
NSM Forwarder


Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
DNS
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Network Service Manager (NSMgr)
Network Service Client (NSC)
Network Service Endpoint (NSE)
NSM Forwarder
Node(Network Service Manager Domain)
Network Service Client (NSE) (Pod)
Network Service Endpoint (NSC) (Pod)
Network Service Manager (NSMgr) (Daemonset)
NSM Forwarder (kernel/vswitch)
Proxy Network Service Registry (pNSR) (Pod)
Proxy Network Service Mgr (pNSMgr) (Pod)
remote.NetworkService.Request
remote.NetworkService.Request

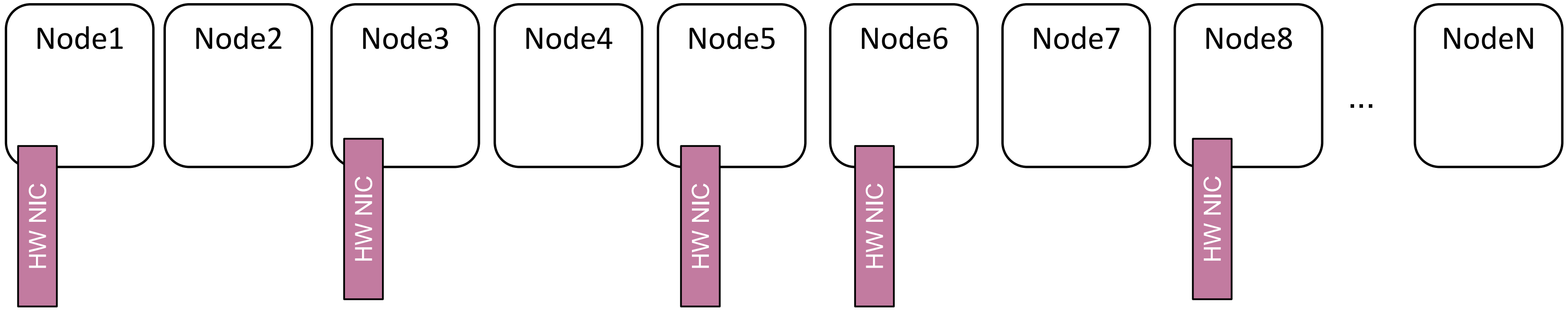

Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
DNS
Network Service Registry Domain (example.com)
Network Service Registry
Network Service Manager Domain
Node(Network Service Manager Domain)
Proxy Network Service Registry (pNSR) (Pod)
Network Service Client (NSE) (Pod)
Network Service Manager (NSMgr) (Daemonset)
Network Service Endpoint (NSC) (Pod)
Proxy Network Service Mgr (pNSMgr) (Pod)
Network Service Manager (NSMgr)
Network Service Client (NSC)
Network Service Endpoint (NSE)
NSM Forwarder
NSM Forwarder
vWire (L2/L3 connection (tunnel))

Network Service Mesh

## HW NICs

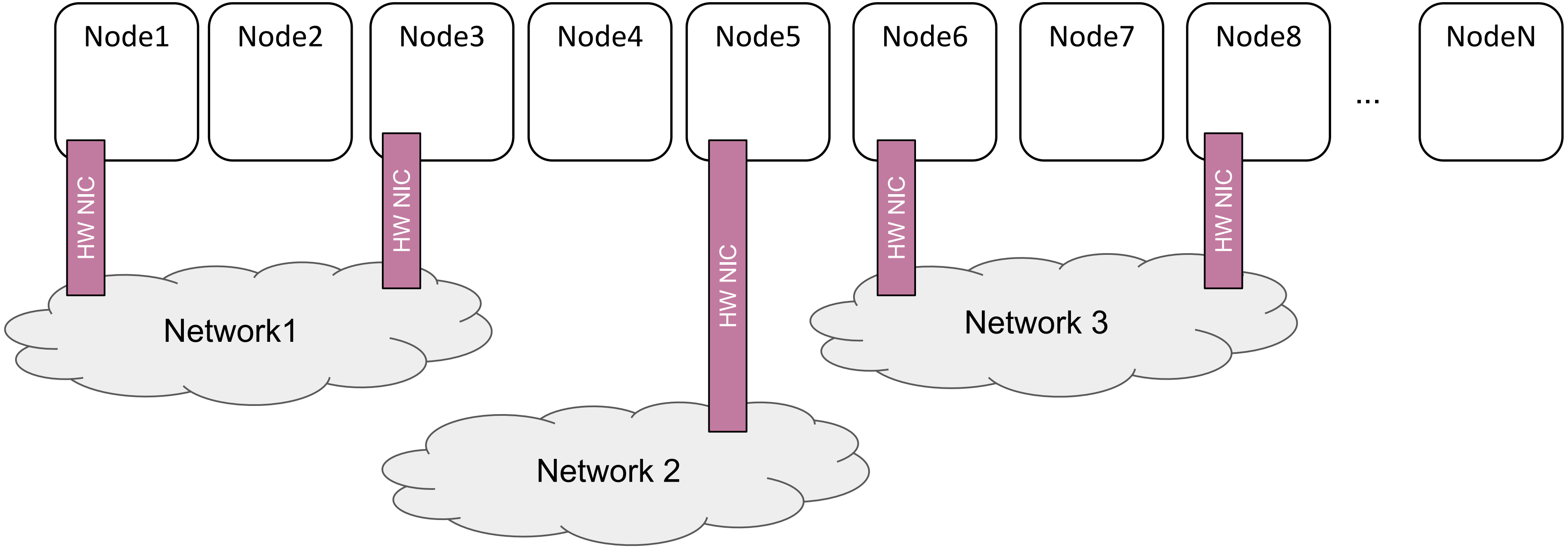
# The Problem: NICs

A Kubernetes Cluster may have special NICs in some but not all Nodes:

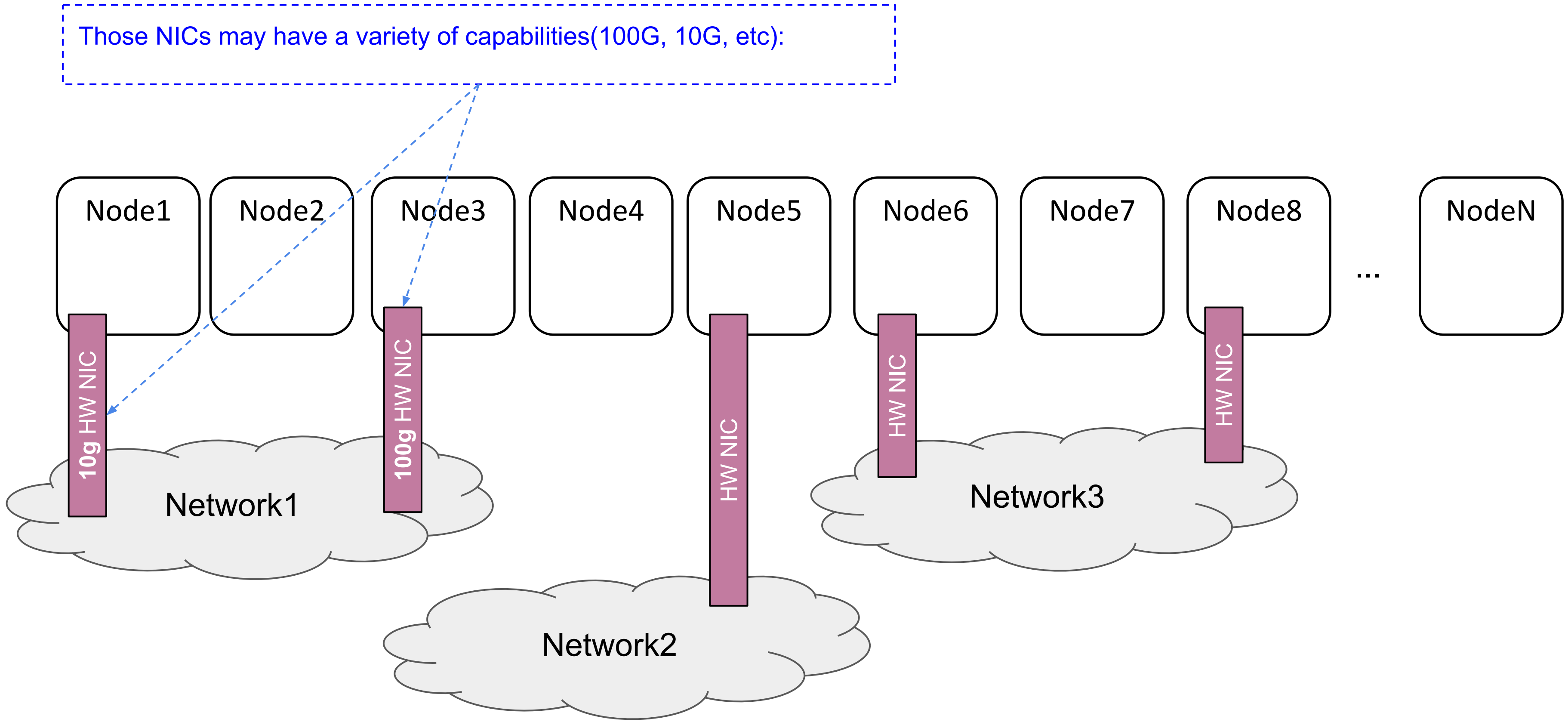


# Not all NICs are on the same Network

Those NICs may be plugged into a variety of different Networks:

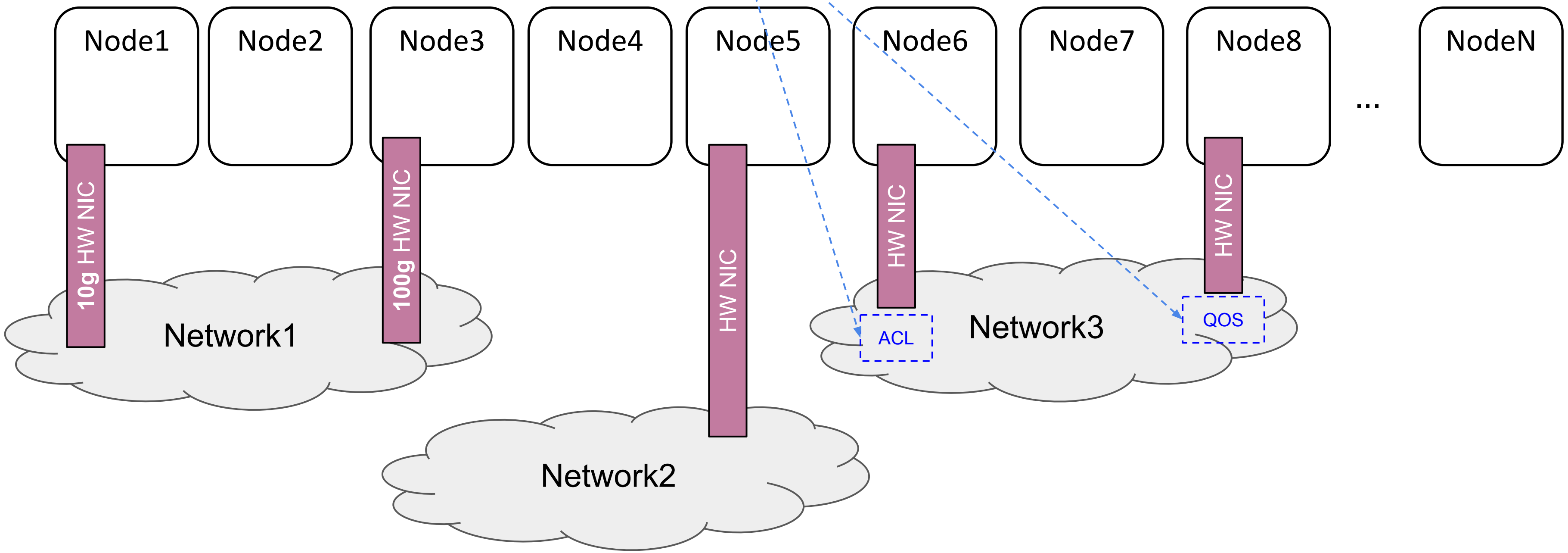


# Not all NICs have the same capabilities



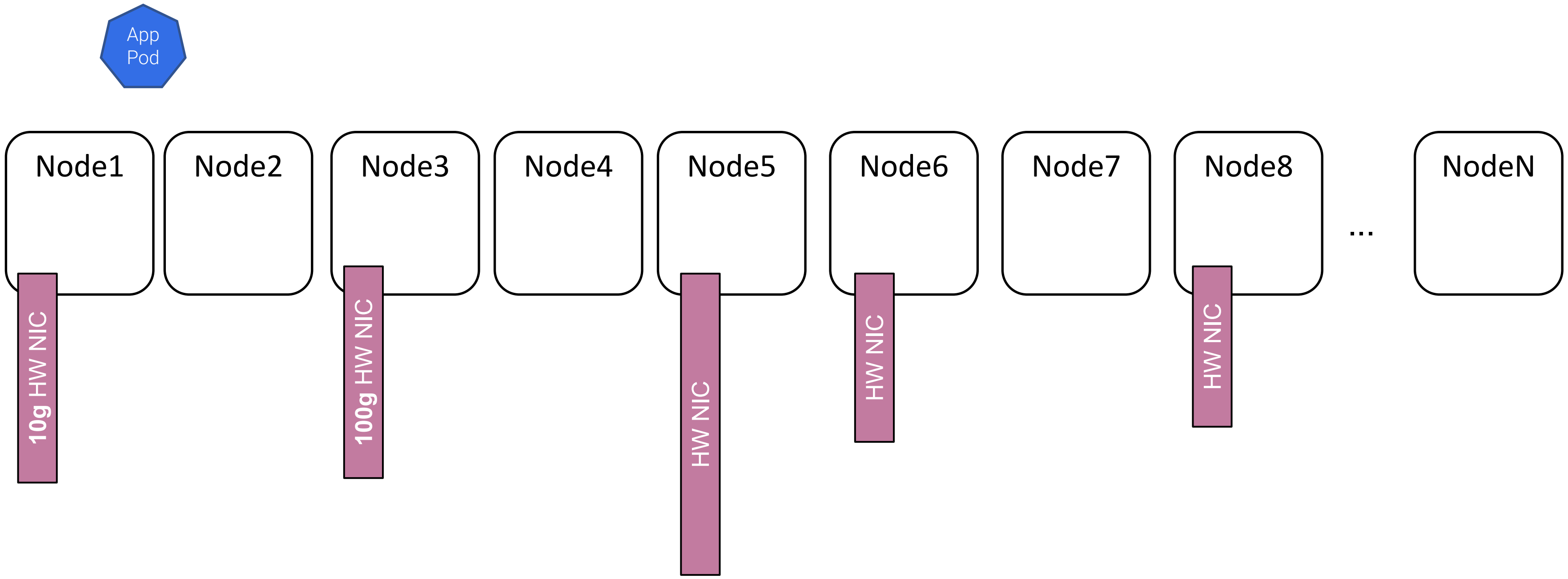
# Not all NICs can access the same Network Service

And two NICs plugged into the *same* Network may have different treatment (ACLs, QoS etc) (Network Service):



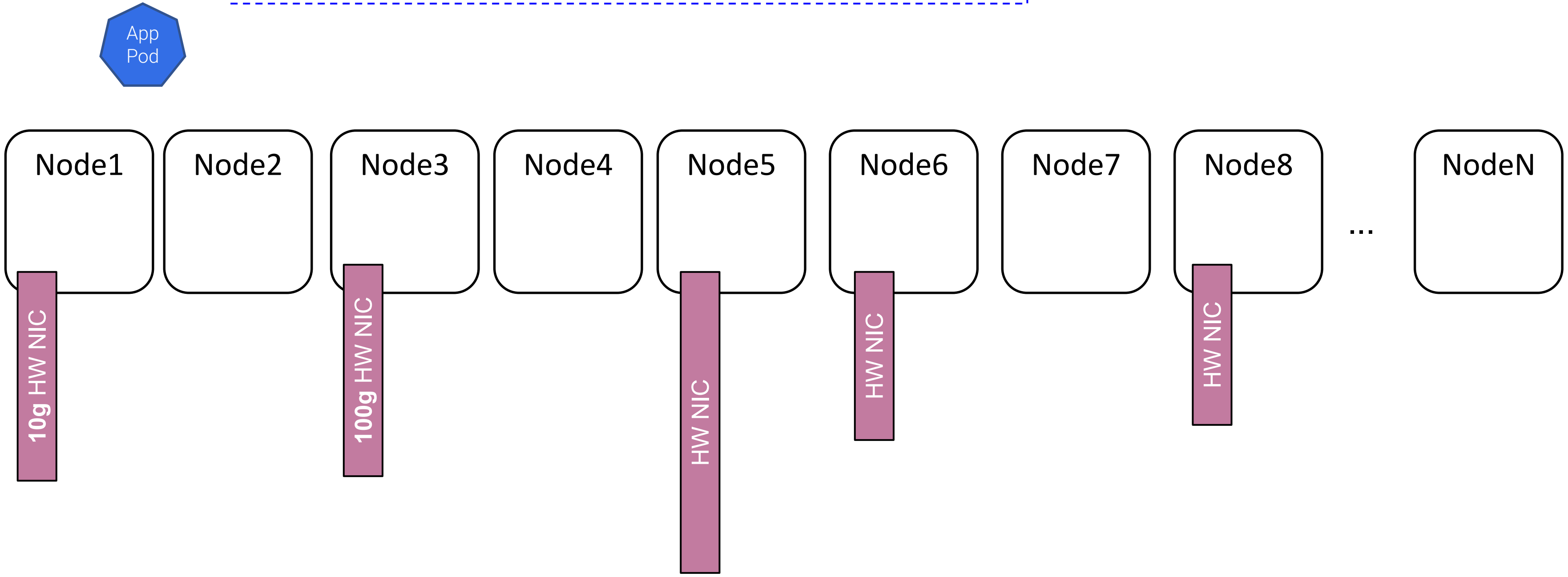
## Scheduling a Pod

So what do we need when deploying a Pod?



# Scheduling a Pod

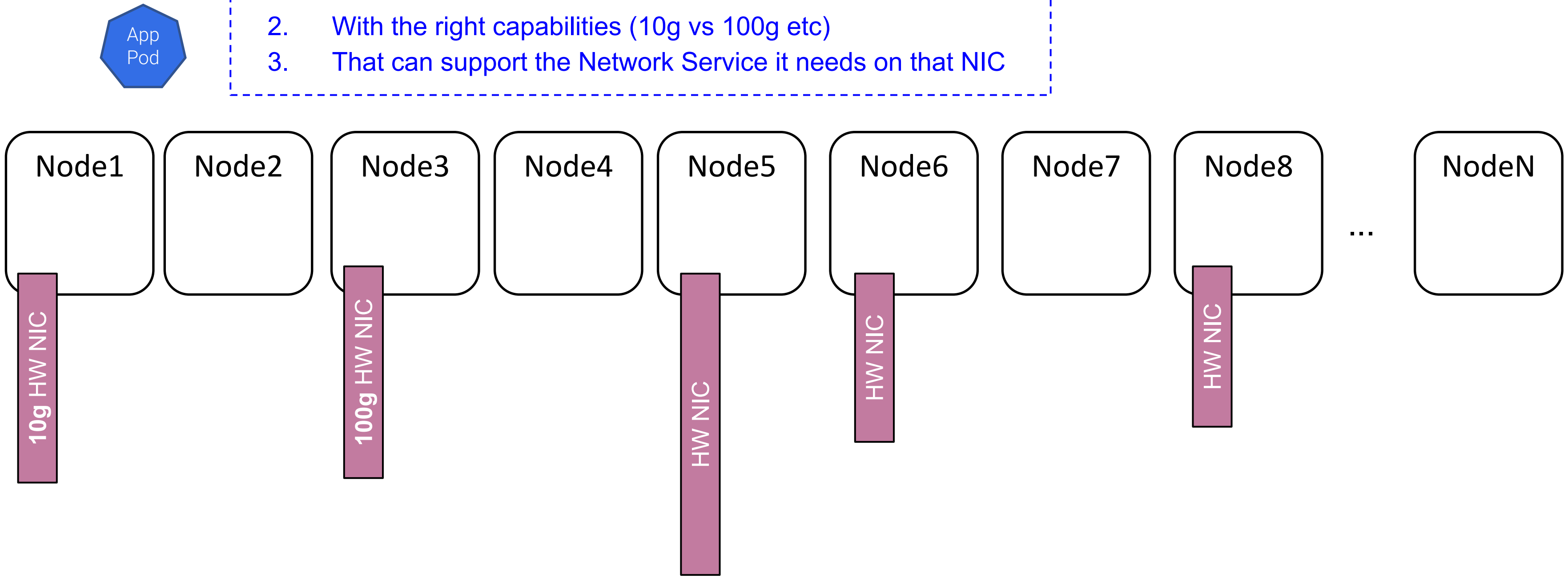
The K8s Scheduler needs to decide which Node to deploy it to...



# Criteria for scheduling a Pod

That Node needs to have:

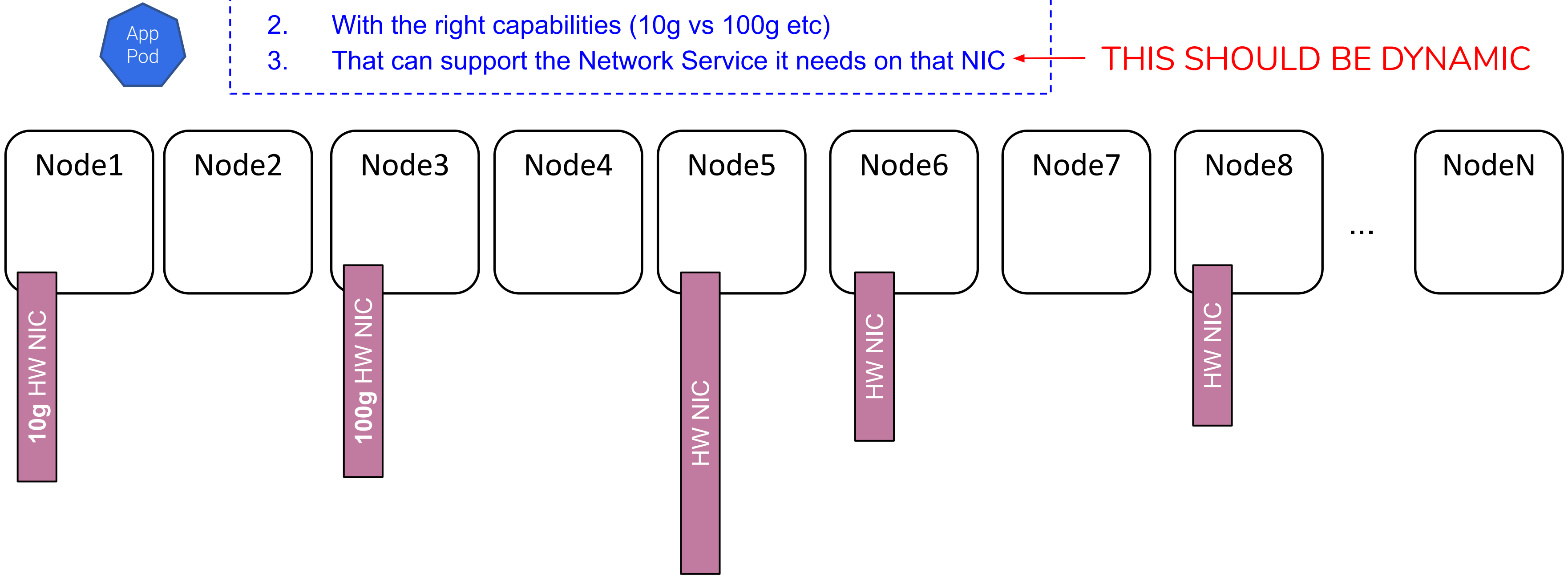
1. A HW NIC
2. With the right capabilities (10g vs 100g etc)
3. That can support the Network Service it needs on that NIC



# Network Service Should be Dynamic

That Node needs to have:

1. A HW NIC
2. With the right capabilities (10g vs 100g etc)
3. That can support the Network Service it needs on that NIC ← THIS SHOULD BE DYNAMIC



# How to ask for it?

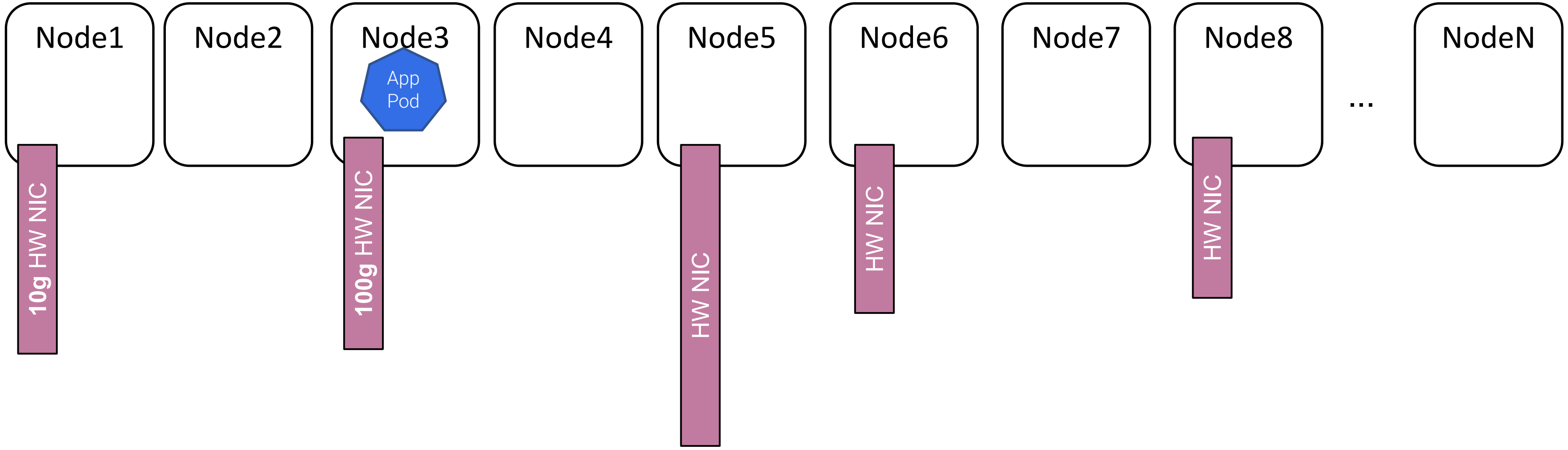
```
apiVersion: v1
kind: Pod
metadata:
 name: cnf-1
 annotations:
   ns.networkservicemesh.io: myns.example.com
spec:
   …
   resources:
      ...
      requires:
         example.com/100g: 1
```

Ask for ‘Network Service’

Ask for resource that can provide Network Services from the example.com domain with 100g capability

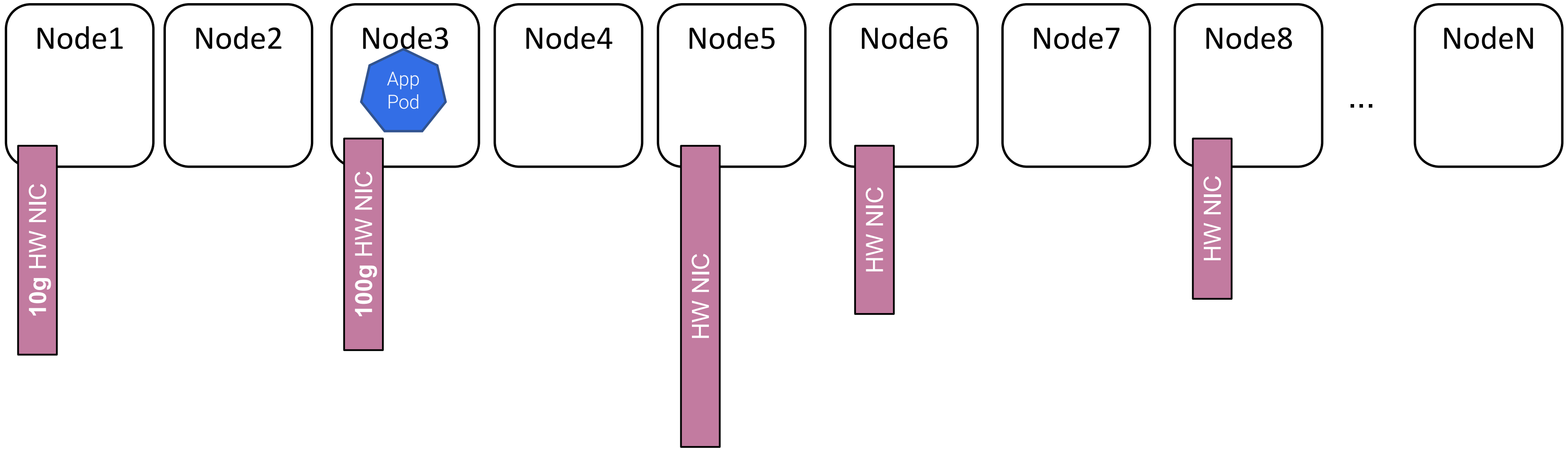
# Scheduling...

Device Plugin schedules us to Node with an available 'example.com/100g' resource where we can get any Network Service in the example.com domain with a NIC with 100g capabilities.



# After scheduling...

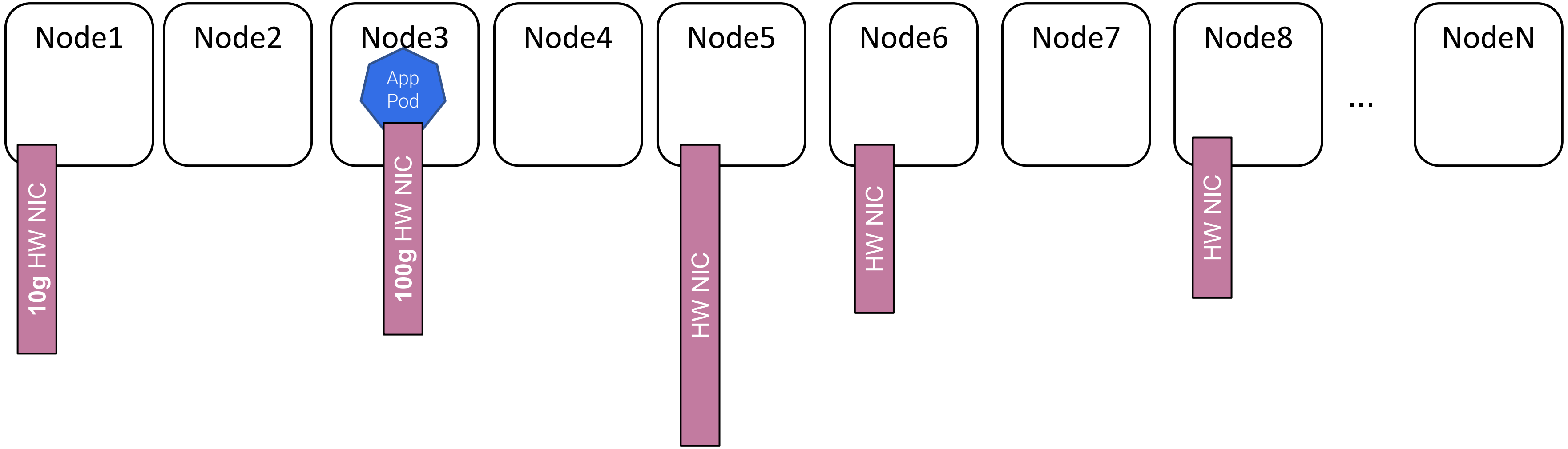
Once Scheduled… we need to:



# Plug NIC into the Pod

Once Scheduled… we need to:

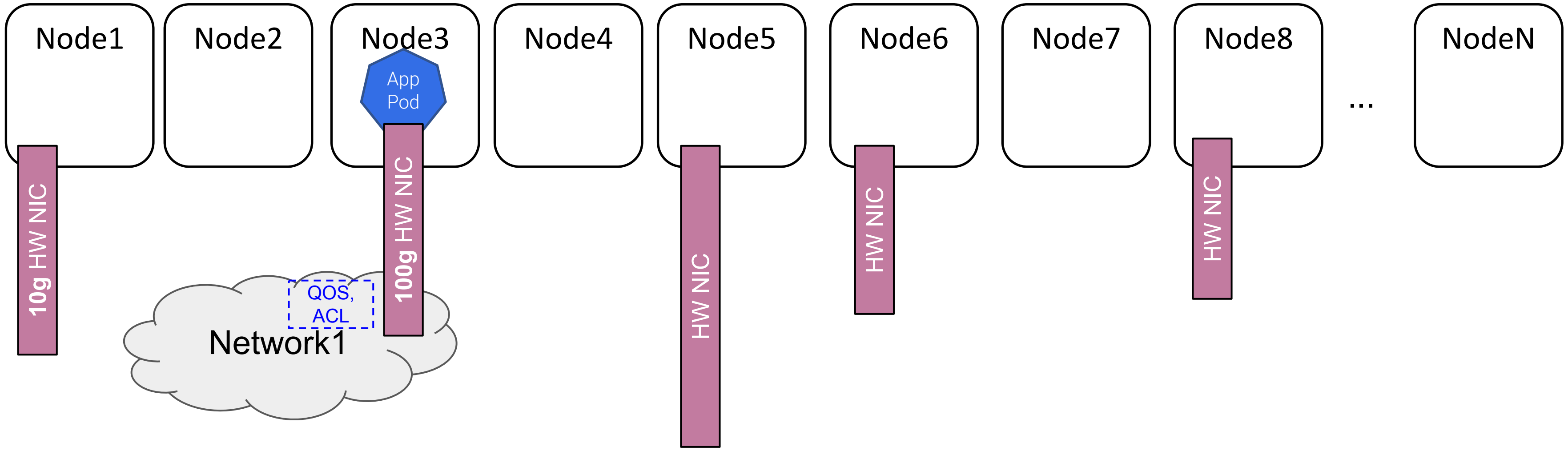
1. Plug the NIC with the correct capabilities into the Pod



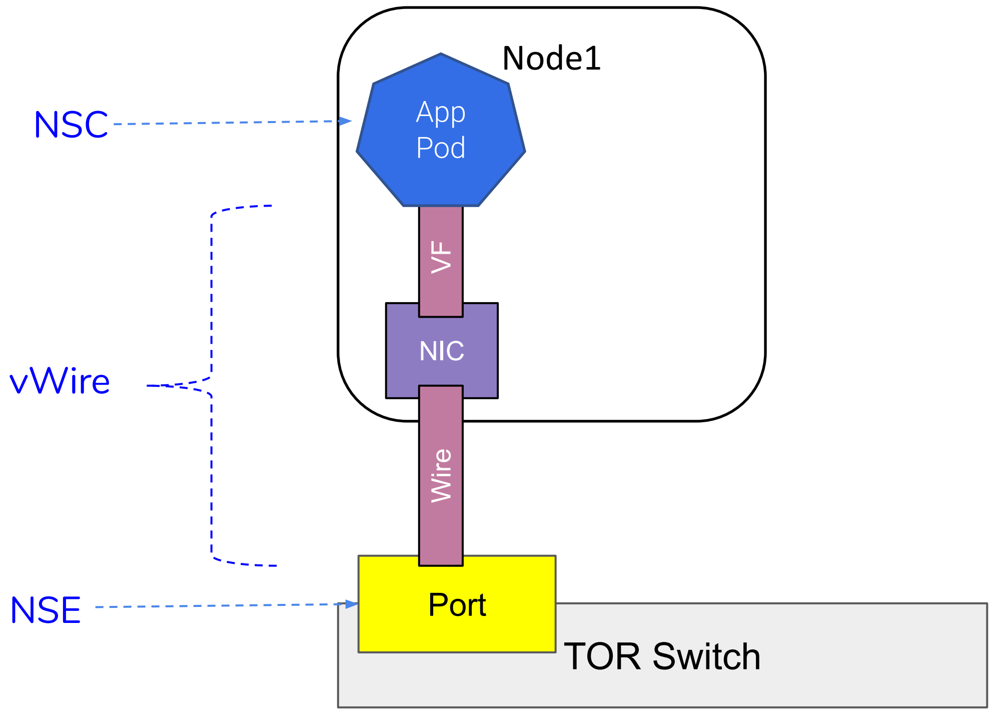
# Configure the Network Service

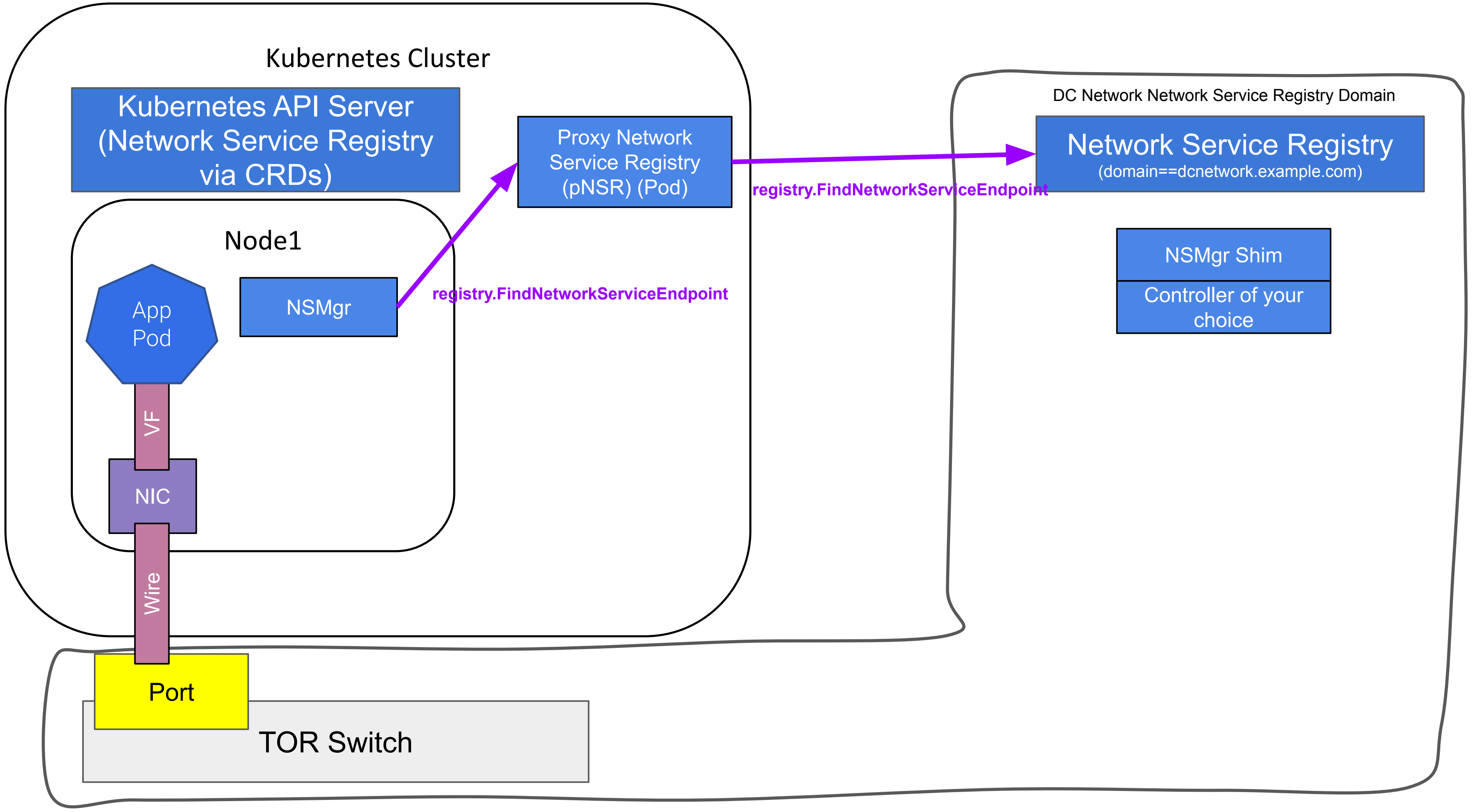
Once Scheduled… we need to:

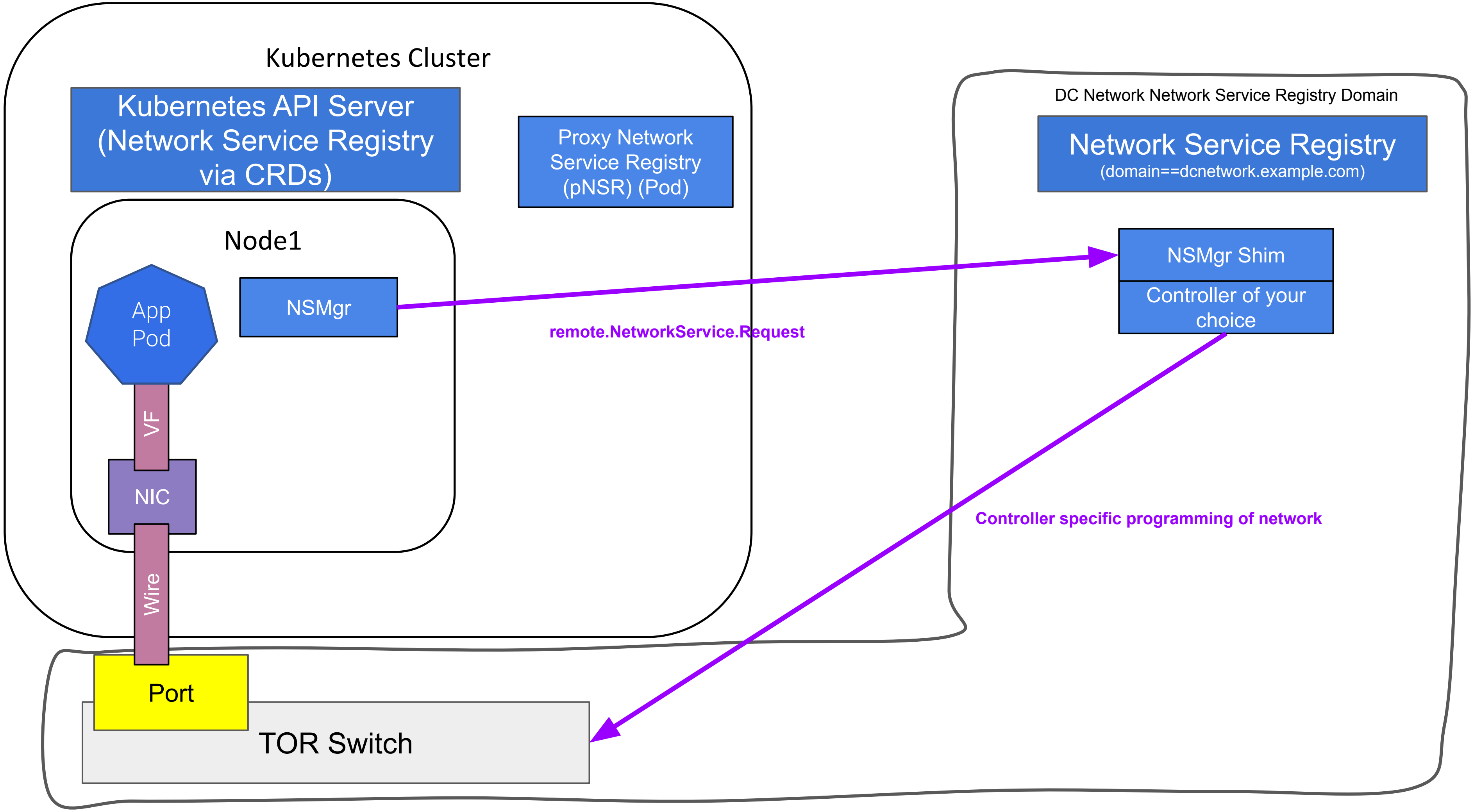
1. Plug the NIC into the Pod
2. Configure the other end of the NIC (TOR port) to provide the Network Service the Pod requested



# Relationship to NSM model




Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
Proxy Network Service Registry (pNSR) (Pod)
Node1
App Pod
NSMgr
VF
NIC
Wire
registry.FindNetworkServiceEndpoint
registry.FindNetworkServiceEndpoint
DC Network Network Service Registry Domain
Network Service Registry
(domain==dcnetwork.example.com)
NSMgr Shim
Controller of your choice
Port
TOR Switch


Kubernetes Cluster
Kubernetes API Server (Network Service Registry via CRDs)
Proxy Network Service Registry (pNSR) (Pod)
Node1
App Pod
NSMgr
VF
NIC
Wire
Port
TOR Switch
remote.NetworkService.Request
DC Network Network Service Registry Domain
Network Service Registry
(domain==dcnetwork.example.com)
NSMgr Shim
Controller of your choice
Controller specific programming of network

## Housekeeping

https://networkservicemesh.io

NSMCon

Nov 18, 2019 | San Diego, California
Colocated with Kubecon+CloudNativeCon 2019

These slides