



**Hewlett Packard**  
Enterprise

# HPE Slingshot in the Kubernetes Ecosystem



**Caio Davi, Jesse Treger**

May, 2025



# Agenda

---

**Intro**

**Components**

**Device Plugin**

**CNI Plugins**

**Deployment**

**Roadmap**



# Introduction

- Concepts



**Admission  
Policies**

**CNI**

**K8s Device Plugin**

**K8s Controller**

**Helm Charts**



# Slingshot Kubernetes Components

## CXI Device Plugin

- New plug in to advertise NIC hardware resources to the Kubelet



## CNI Plugin

- The configuration details for existing components that manage the network plumbing



## K8s Operator

- Manage CXI resources on the host/pod.

COMING SOON

## Helm Chart

- Installation assistance

COMING SOON

Core configuration components to run kubernetes with RDMA on Cassini NICs

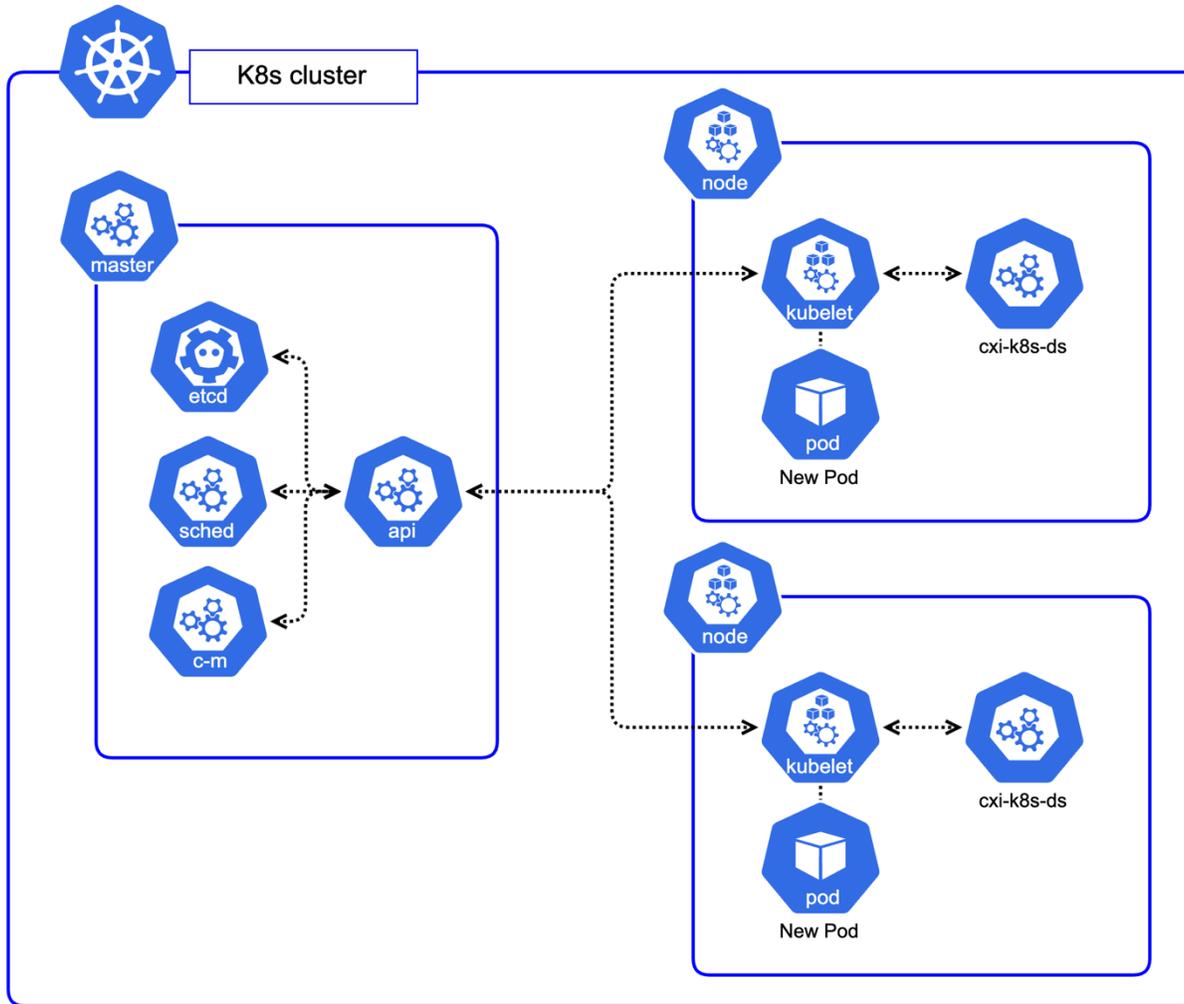
Future investment for more elegant management of the NIC resources (akin to SLURM plug-in)

Future investment simpler installation and configuration

Cxi-k8s-device-plugin available at:

<https://github.com/HewlettPackard/cxi-k8s-device-plugin>

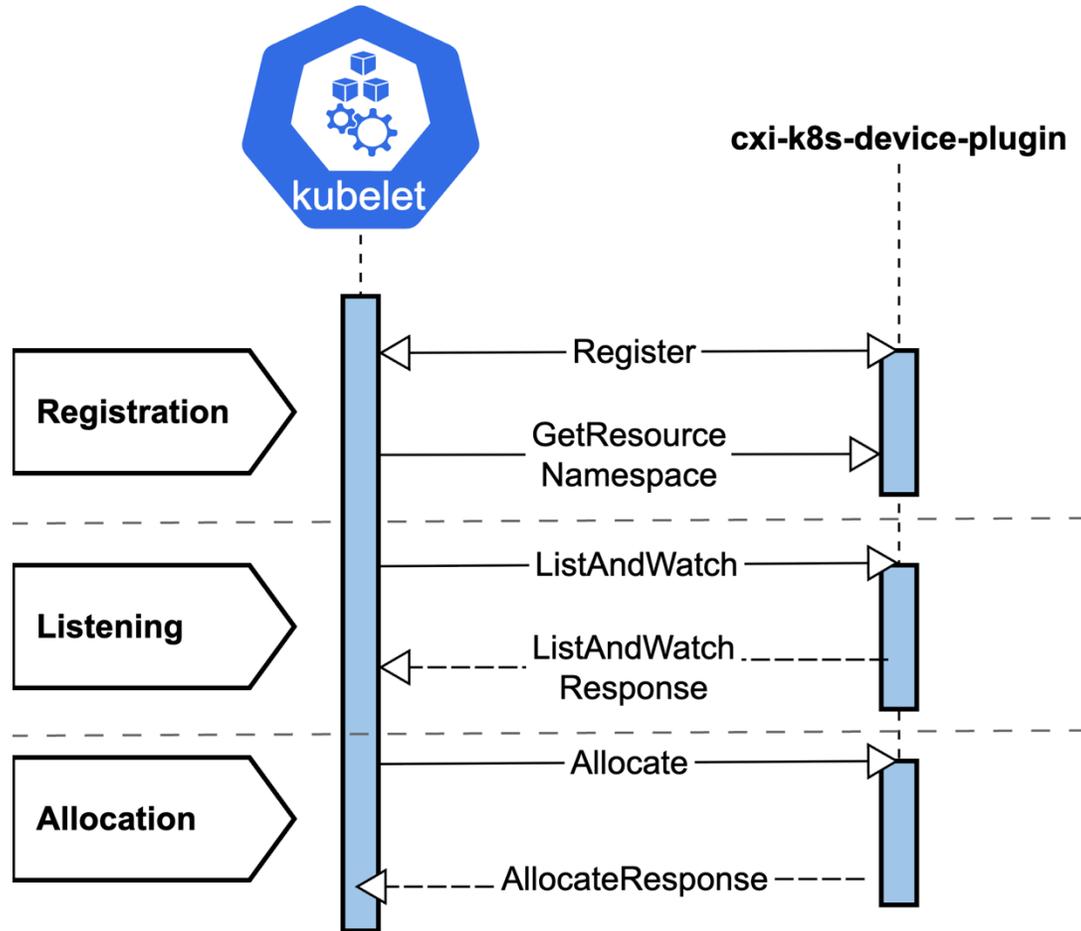
# Device plugin



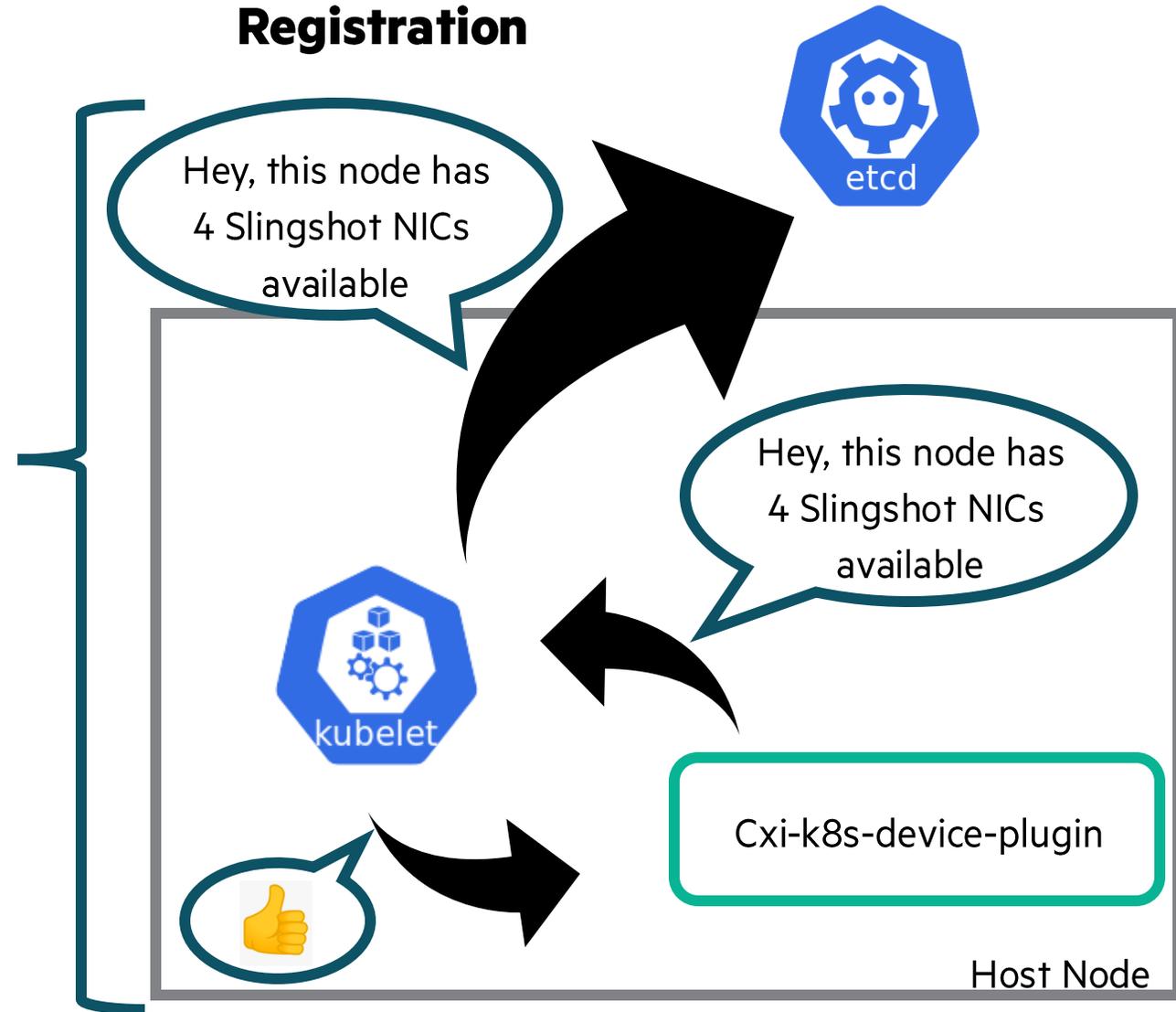
- The cxi-k8s-device-plugin is a daemon that may run in the host or in a pod, as a daemon-set.
- It assists Kubernetes management of physical devices.
- Additionally, it provides the Slingshot-specific software stack to the pod



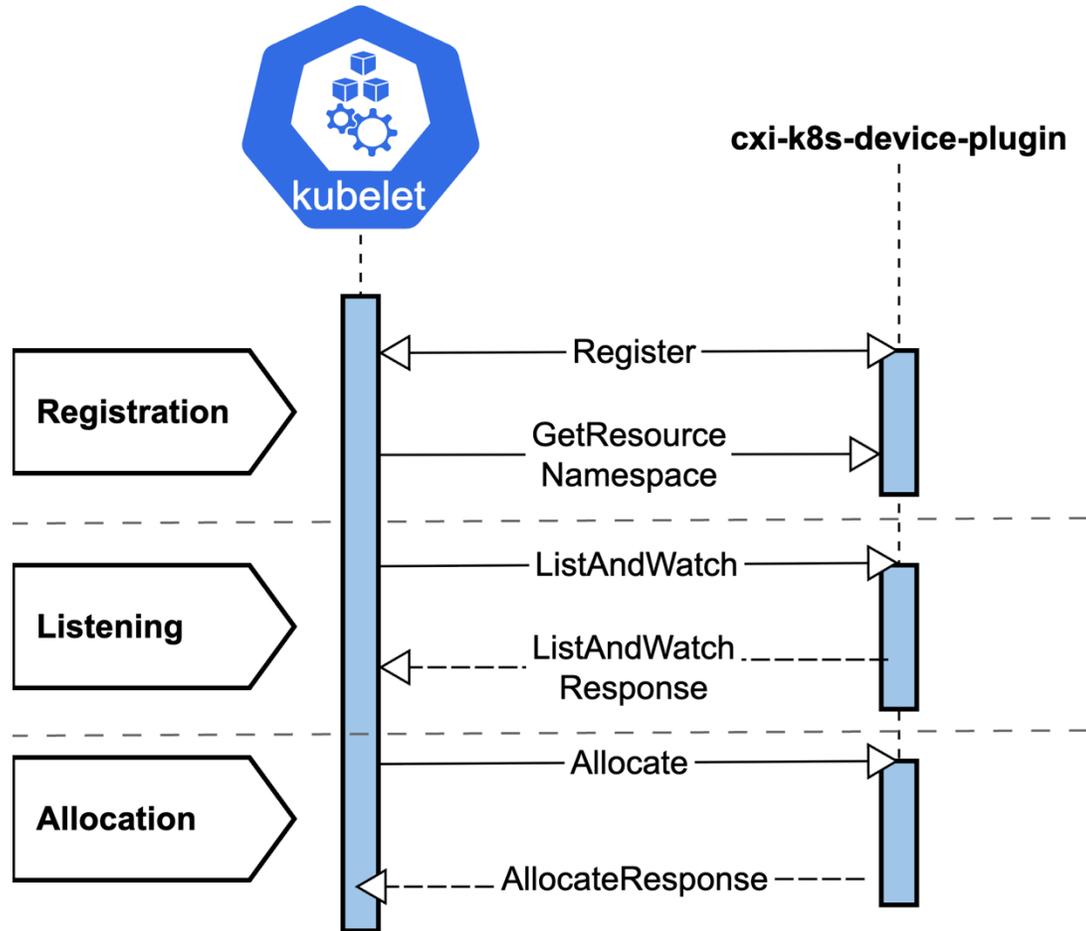
# Device plugin



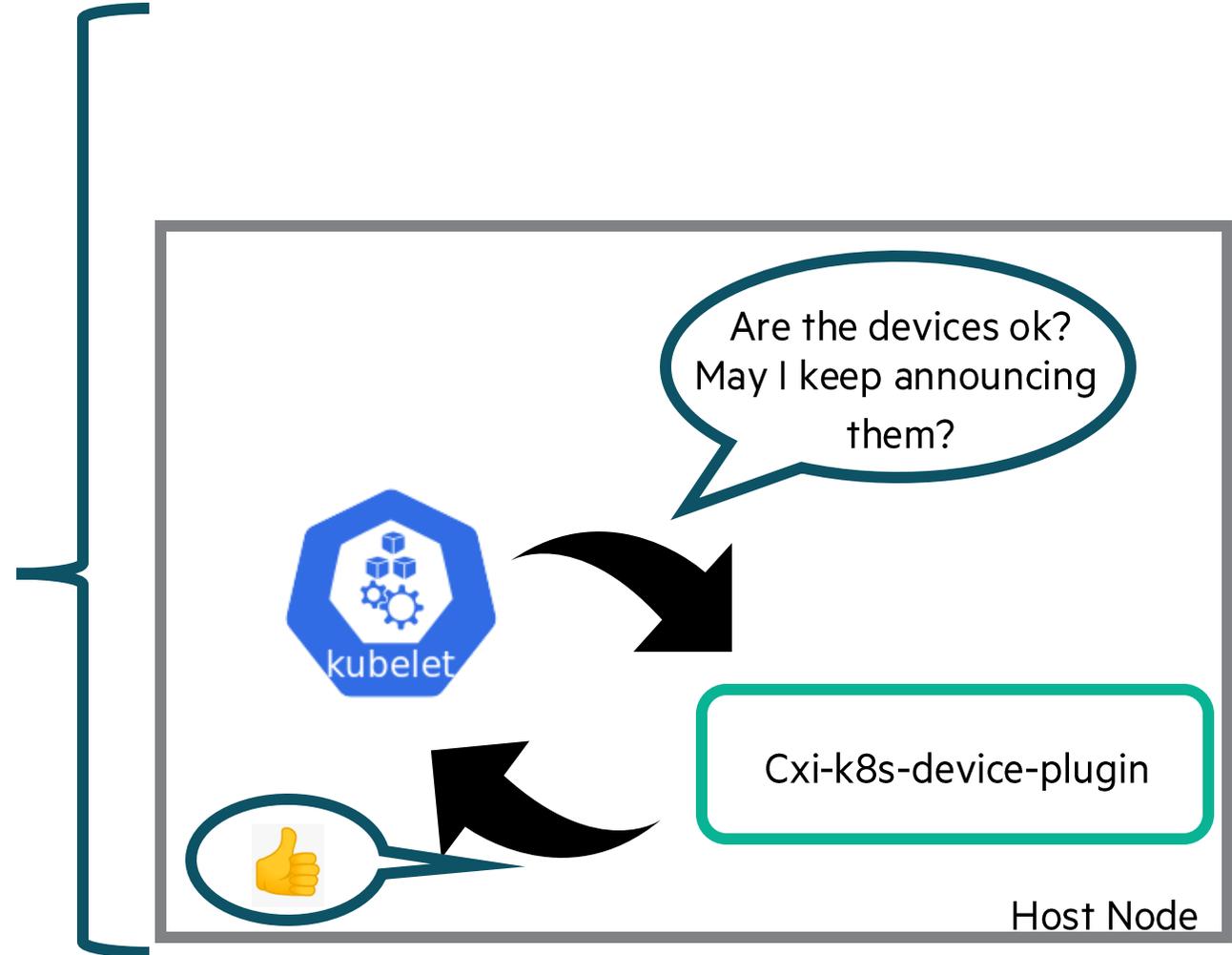
## Registration



# Device plugin



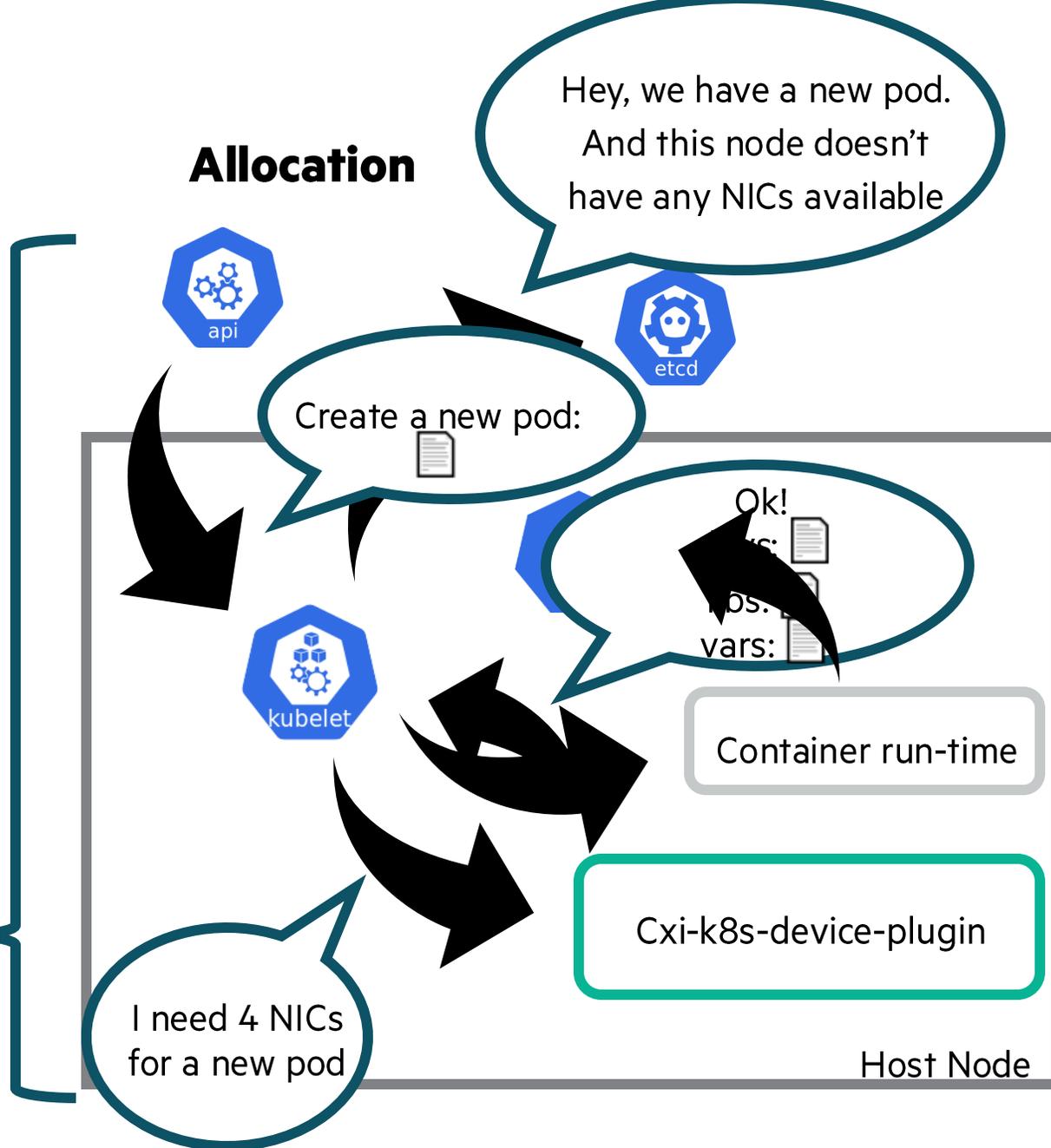
## Listening



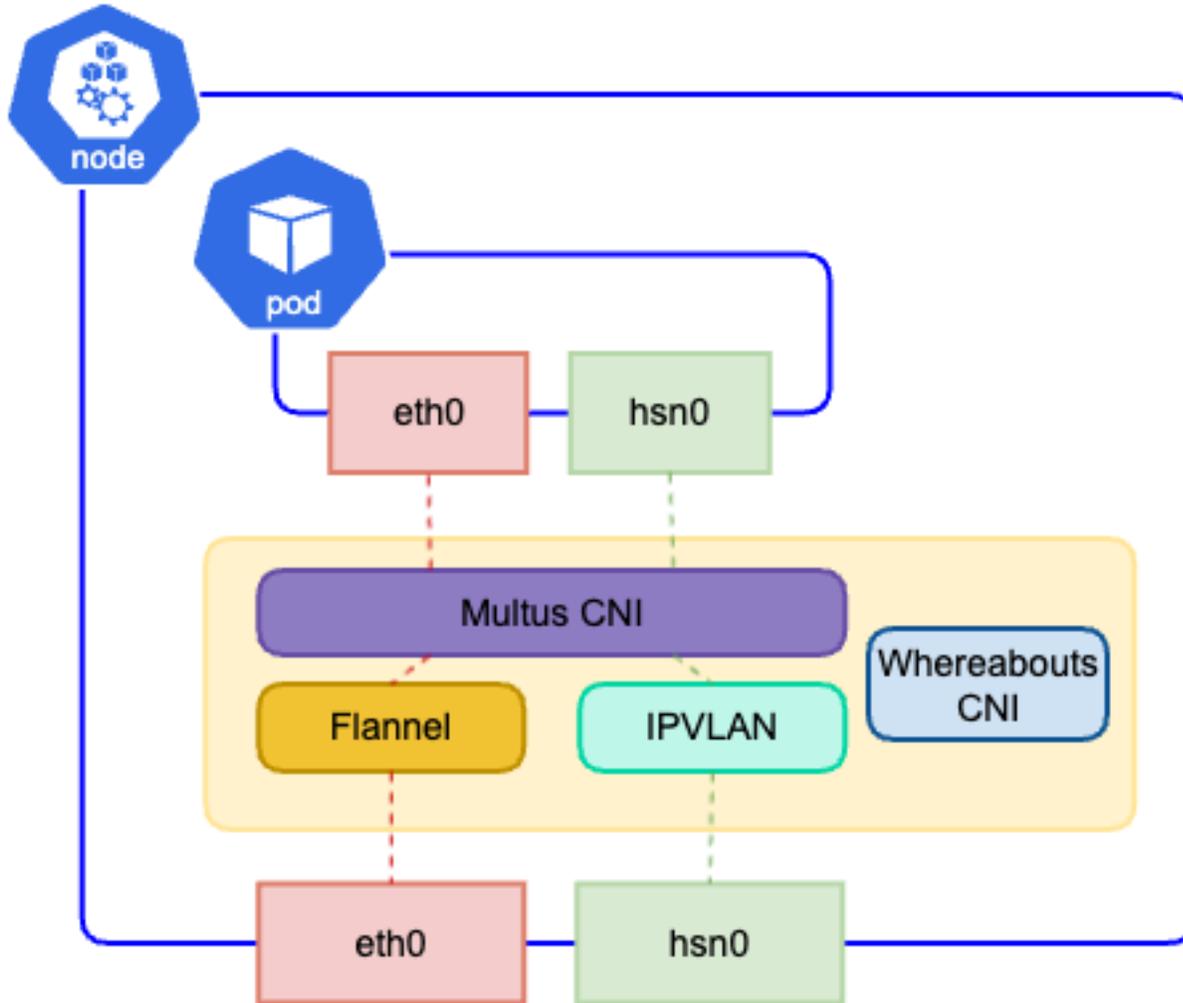
# Device plugin

```
apiVersion: v1
kind: Pod
metadata:
  name: cxi-test
spec:
  containers:
  - name: cxi-test-container
    image: ubuntu
    command:
    - sleep
    - "3600"
  resources:
    requests:
      beta.hpe.com/cxi: 4      # requesting four NICs
```

## Allocation



# CNI plugins



- CNI plugins implement functional network requirements for the K8s cluster.

**Multus** enables multiple interfaces in the pod.

**IPVLAN** makes the network plumbing between host and pod.

**Whereabouts** publish the new interfaces' IPs throughout the cluster.

# Deployment

```
ssh 192.168.1.101
# cxi_service enable -d cxi0 -s 1
Successfully enabled service: 1
# cxi_service enable -d cxi1 -s 1
Successfully enabled service: 1
# cxi_service enable -d cxi2 -s 1
Successfully enabled service: 1
# cxi_service enable -d cxi3 -s 1
Successfully enabled service: 1
#
```

- Requirements
  - **CXI Services Enabled**
  - Mutating Admission Policies
  - Multus CNI
  - Whereabouts CNI
- Install device plugin



# Deployment

```
ssh ㄿ#1
# sed -i '/^      - kube-apiserver$/a\      - --feature-gates=MutatingAdmissionPolicy=true \n      - --runtime-config=admissionregistration.k8s.io/v1alpha1=true' kube-apiserver.yaml
#
# cat kube-apiserver.yaml | grep -B 4 -A 1 "feature-gates"
spec:
  containers:
  - command:
    - kube-apiserver
    - --feature-gates=MutatingAdmissionPolicy=true
    - --runtime-config=admissionregistration.k8s.io/v1alpha1=true
# █
```

- Requirements
  - CXI Services Enabled
  - **Mutating Admission Policies**
  - Multus CNI
  - Whereabouts CNI
- Install device plugin



# Deployment

```
ssh ㄿ#1
# kubectl apply -f https://raw.githubusercontent.com/k8snetworkplum
bingwg/multus-cni/master/deployments/multus-daemonset-thick.yml
customresourcedefinition.apiextensions.k8s.io/network-attachment-de
finitions.k8s.cni.cncf.io unchanged
clusterrole.rbac.authorization.k8s.io/multus unchanged
clusterrolebinding.rbac.authorization.k8s.io/multus unchanged
serviceaccount/multus unchanged
configmap/multus-daemon-config unchanged
daemonset.apps/kube-multus-ds unchanged
# █
```

- Requirements
  - CXI Services Enabled
  - Mutating Admission Policies
  - **Multus CNI**
  - Whereabouts CNI
- Install device plugin



# Deployment

```
ssh 1
# cd whereabouts/
# kubectl apply \
>   -f doc/crds/daemonset-install.yaml \
>   -f doc/crds/whereabouts.cni.cncf.io_ippools.yaml \
>   -f doc/crds/whereabouts.cni.cncf.io_overlappingrangeipreservations.yaml
serviceaccount/whereabouts unchanged
clusterrolebinding.rbac.authorization.k8s.io/whereabouts unchanged
clusterrole.rbac.authorization.k8s.io/whereabouts-cni unchanged
configmap/whereabouts-config unchanged
daemonset.apps/whereabouts unchanged
customresourcedefinition.apiextensions.k8s.io/ippools.whereabouts.cni.cncf.io unchanged
customresourcedefinition.apiextensions.k8s.io/overlappingrangeipreservations.whereabouts.cni.cncf.io unchanged
# █
```

- Requirements
  - CXI Services Enabled
  - Mutating Admission Policies
  - Multus CNI
  - **Whereabouts CNI**
- Install device plugin



# Deployment

```
ssh ㄿ#1
# cd cxi-k8s-device-plugin/
#
# kubectl apply \
>   -f ./deploy/NetworkAttachmentDefinition \
>   -f ./deploy/MutatingAdmissionPolicy \
>   -f ./deploy/hpecxi-device-plugin-ds.yaml
networkattachmentdefinition.k8s.cni.cncf.io/hsn0-ipvlan unchanged
networkattachmentdefinition.k8s.cni.cncf.io/hsn1-ipvlan unchanged
networkattachmentdefinition.k8s.cni.cncf.io/hsn2-ipvlan unchanged
networkattachmentdefinition.k8s.cni.cncf.io/hsn3-ipvlan unchanged
mutatingadmissionpolicybinding.admissionregistration.k8s.io/hpecxi-
mutating-admission-policy-binding unchanged
mutatingadmissionpolicy.admissionregistration.k8s.io/hpecxi-mutatin
g-admission-policy configured
daemonset.apps/hpecxi-device-plugin-daemonset configured
#
```

- Requirements
  - CXI Services Enabled
  - Mutating Admission Policies
  - Multus CNI
  - Whereabouts CNI
- **Install device plugin**

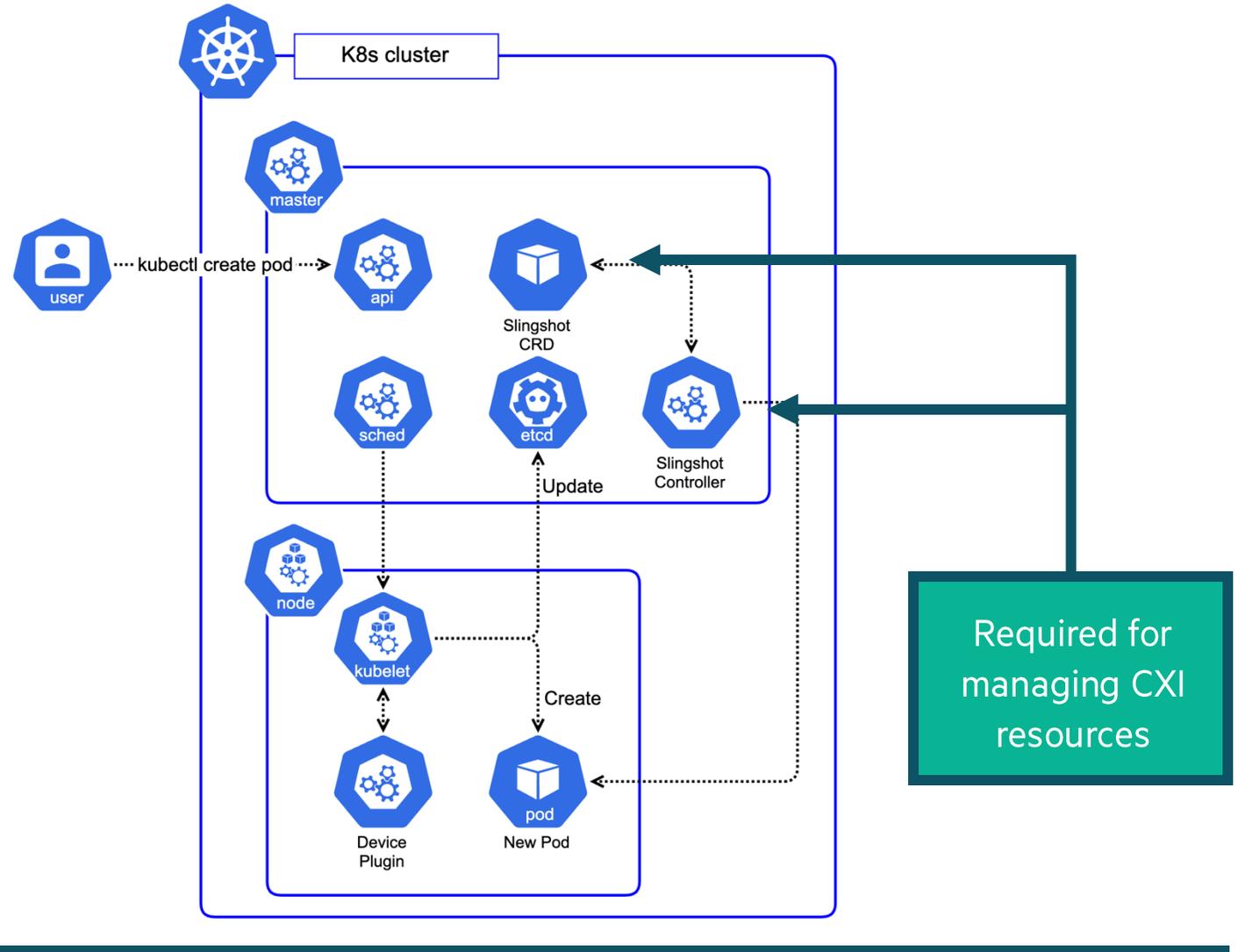


# Limitations

No shared CXI devices across pods

Full load always

Software versioning

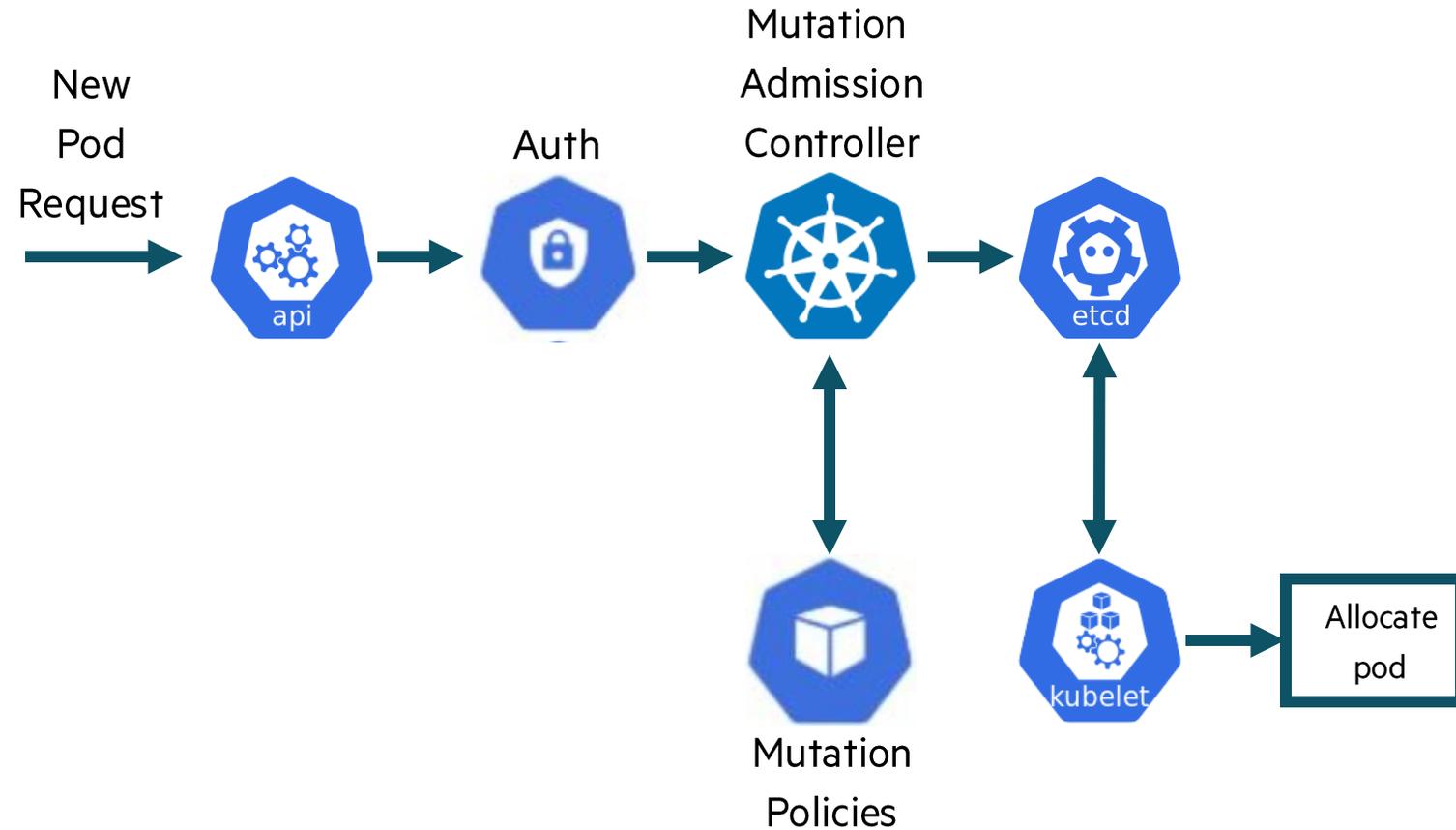


# Limitations

**No shared CXI devices across pods.**

**Full load always**

**Software versioning**



# Limitations

**No shared CXI devices across pods.**

**Full load always**

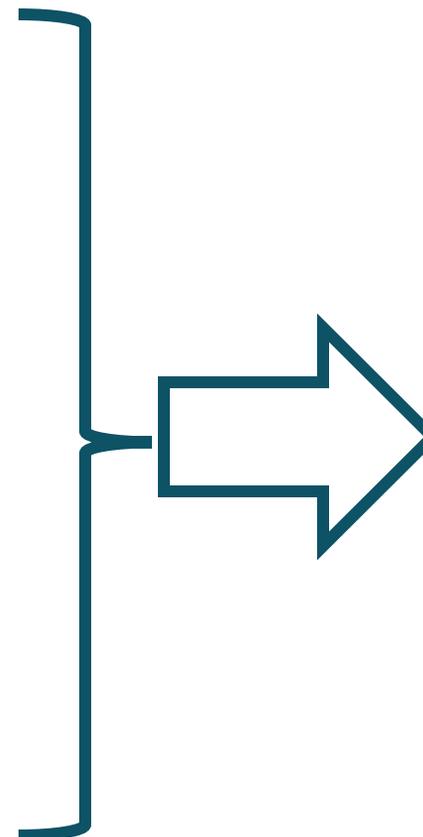
**Software versioning**



Components:  
pytorch v.X  
CUDA v.Y  
NCCL v.Z  
OFI-plugin v.A

Requirements:  
Libfabric v.B

Host:  
Libfabric v.C



**RUNTIME  
ERROR**

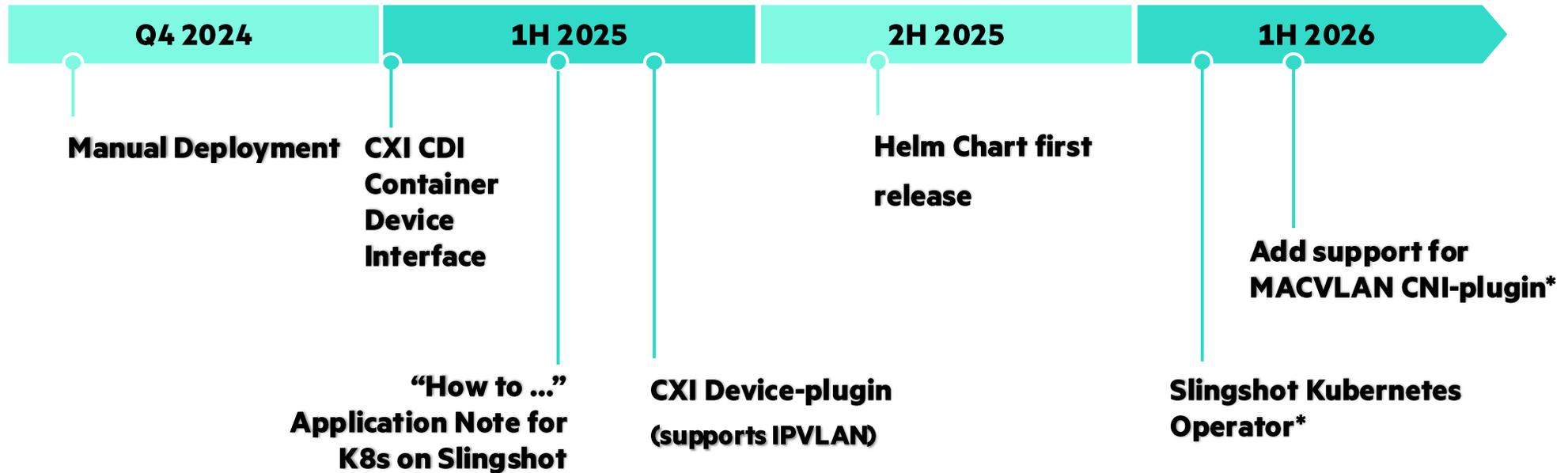


# Distribution Specific Considerations

- Currently tested with Vanilla Kubernetes v1.32
- Not tested, but should work:
  - Rancher
  - Ezmeral
  - OpenShift
- Distribution-specific may impact deployment steps
- Kubernetes version matters. Some required features are only available in v1.32+



# Next Steps for Kubernetes on Slingshot



# Conclusion

---

- Kubernetes runs on HPE Slingshot
- Tooling helps to simplify the installation/configuration
- New tooling is required (and planned) to further enable capabilities and ease of install
  - Helm Chart to facilitate installation and versioning
  - K8 operator to enable NIC resource management – including VNI
- New fabric features will enable additional capabilities
  - MACVLAN



**Thank You!**

**[caio.davi@hpe.com](mailto:caio.davi@hpe.com)**



[HPE Slingshot Configuration Guide](#)

