

PKS Networking with NSX-T

“you focus on your app, we'll take care of the rest !”

Gaetano Borgione, @TanoBorgione

Angela Chin, @AngelaSChin

Agenda

- What is PKS
- Networking in Kubernetes
- What is NSX
- NSX + PKS
- Cluster Creation
- Cluster Upgrade

Disclaimer

Safe Harbor Statement

The following is intended to outline the general direction of Pivotal's offerings. It is intended for information purposes only and may not be incorporated into any contract. Any information regarding pre-release of Pivotal offerings, future updates or other planned modifications is subject to ongoing evaluation by Pivotal and is subject to change. This information is provided without warranty of any kind, express or implied, and is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions regarding Pivotal's offerings. These purchasing decisions should only be based on features currently available. The development, release, and timing of any features or functionality described for Pivotal's offerings in this presentation remain at the sole discretion of Pivotal. Pivotal has no obligation to update forward looking information in this presentation.

What is PKS

PKS

- Pivotal Container Service

PKS

- Pivotal Container Service
- Managed Kubernetes for multiple clusters

PKS

- Pivotal Container Service
- **Managed** Kubernetes for multiple clusters

PKS

- Pivotal Container Service
- **Managed** Kubernetes for **multiple clusters**

PKS

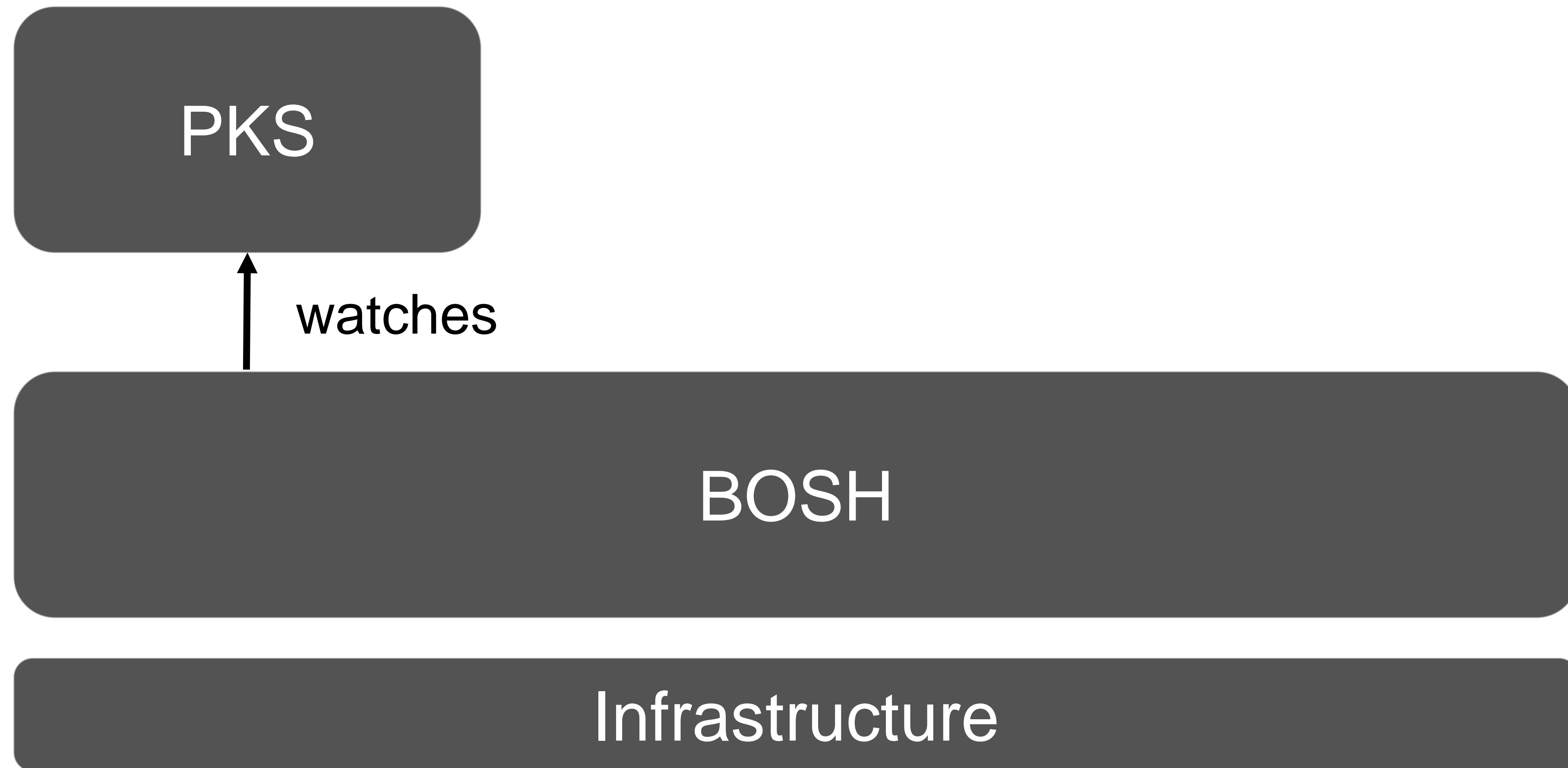
- Pivotal Container Service
- **Managed** Kubernetes for **multiple clusters**
- Aimed at **Day 2** Operations

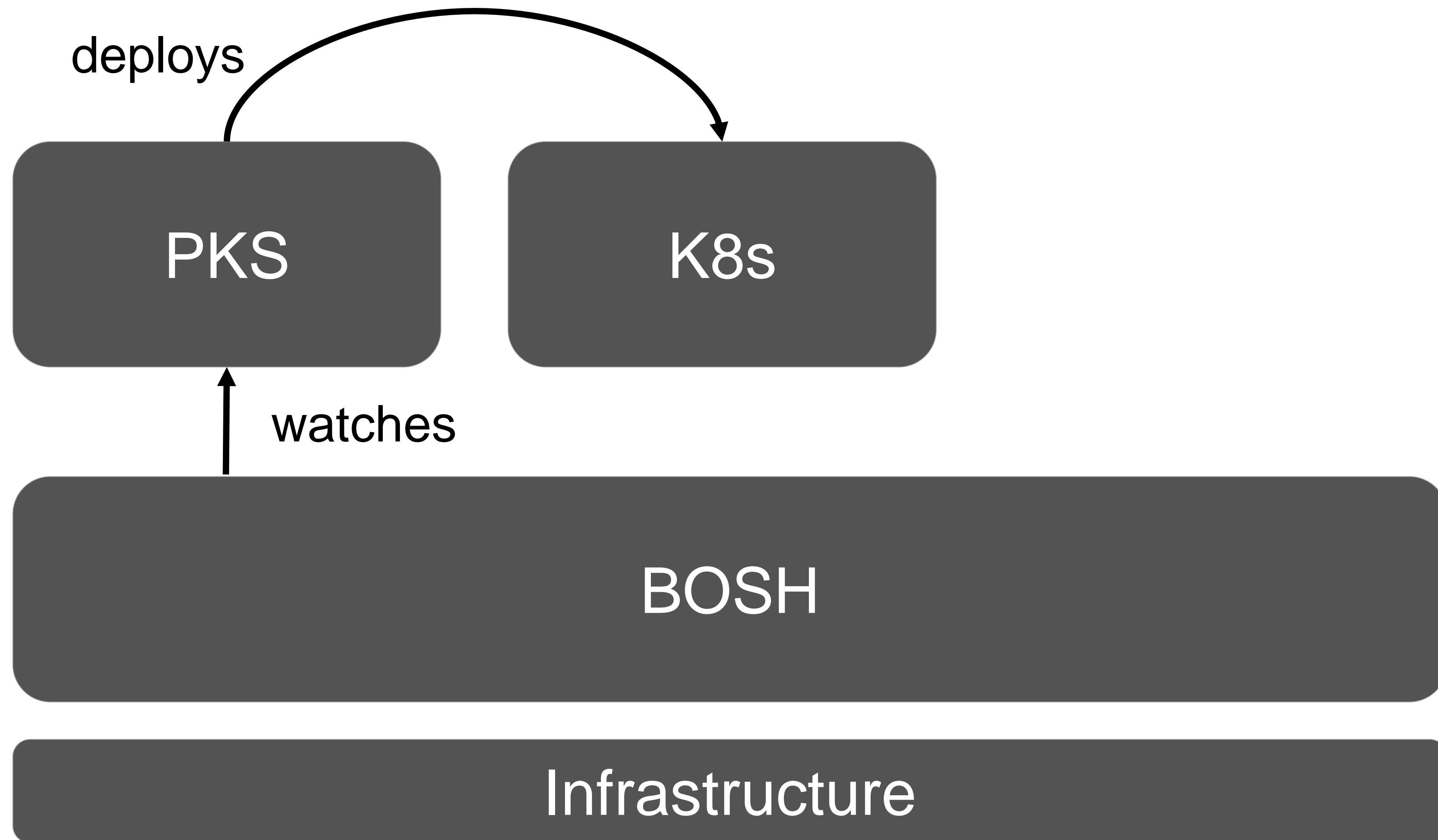
PKS

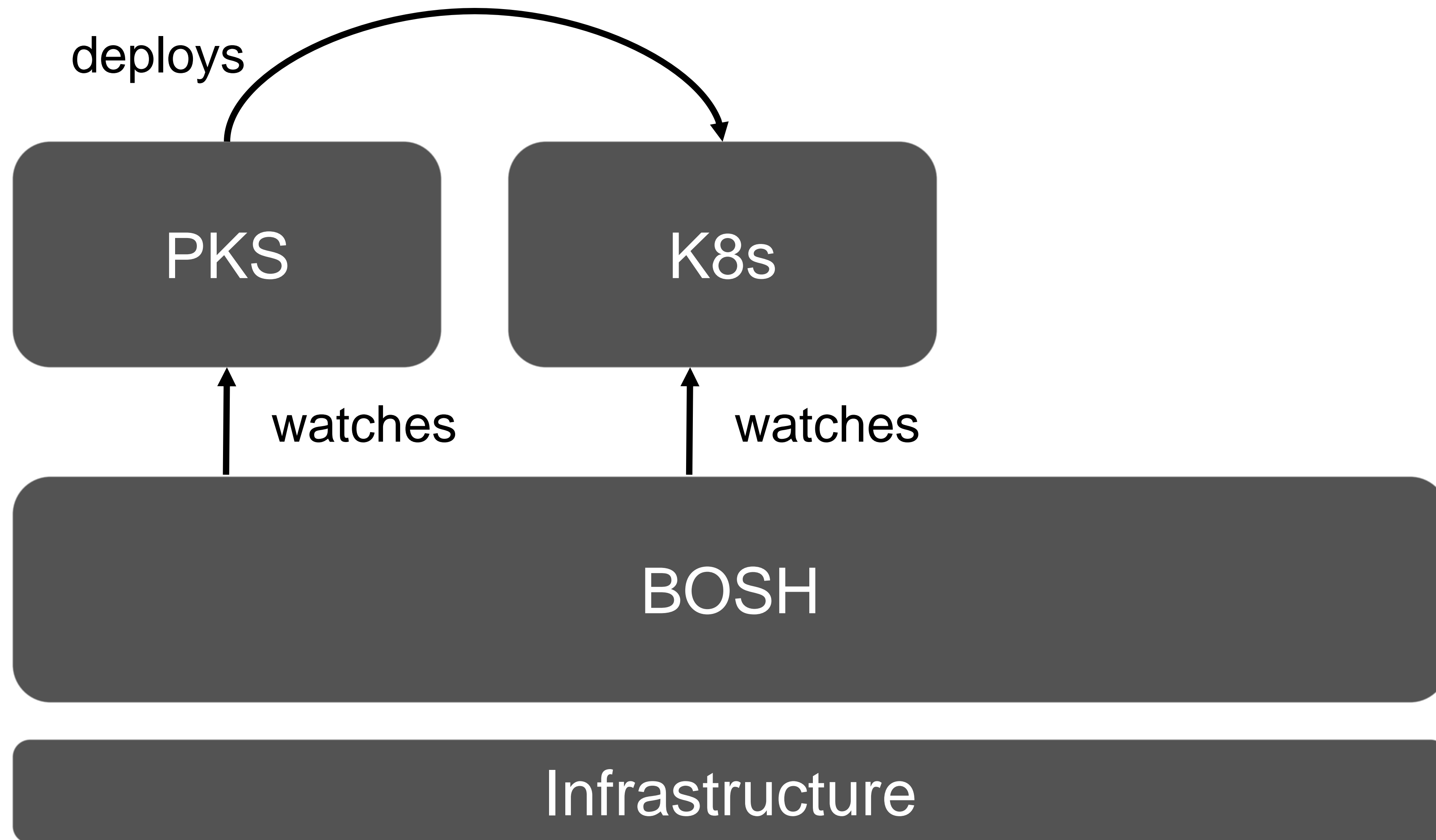
- Pivotal Container Service
- **Managed** Kubernetes for **multiple clusters**
- Aimed at **Day 2** Operations
- Uses **BOSH!**

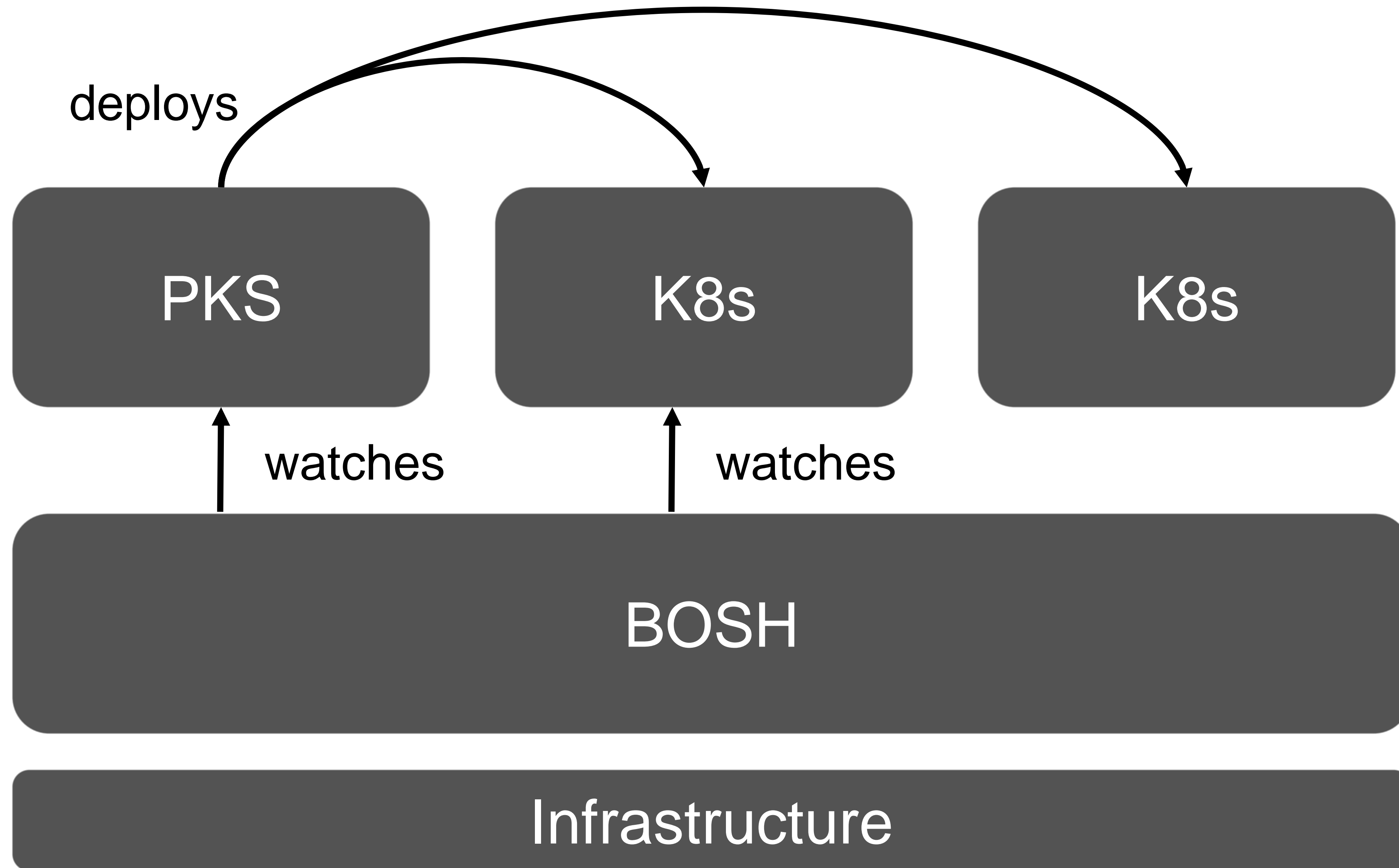
BOSH

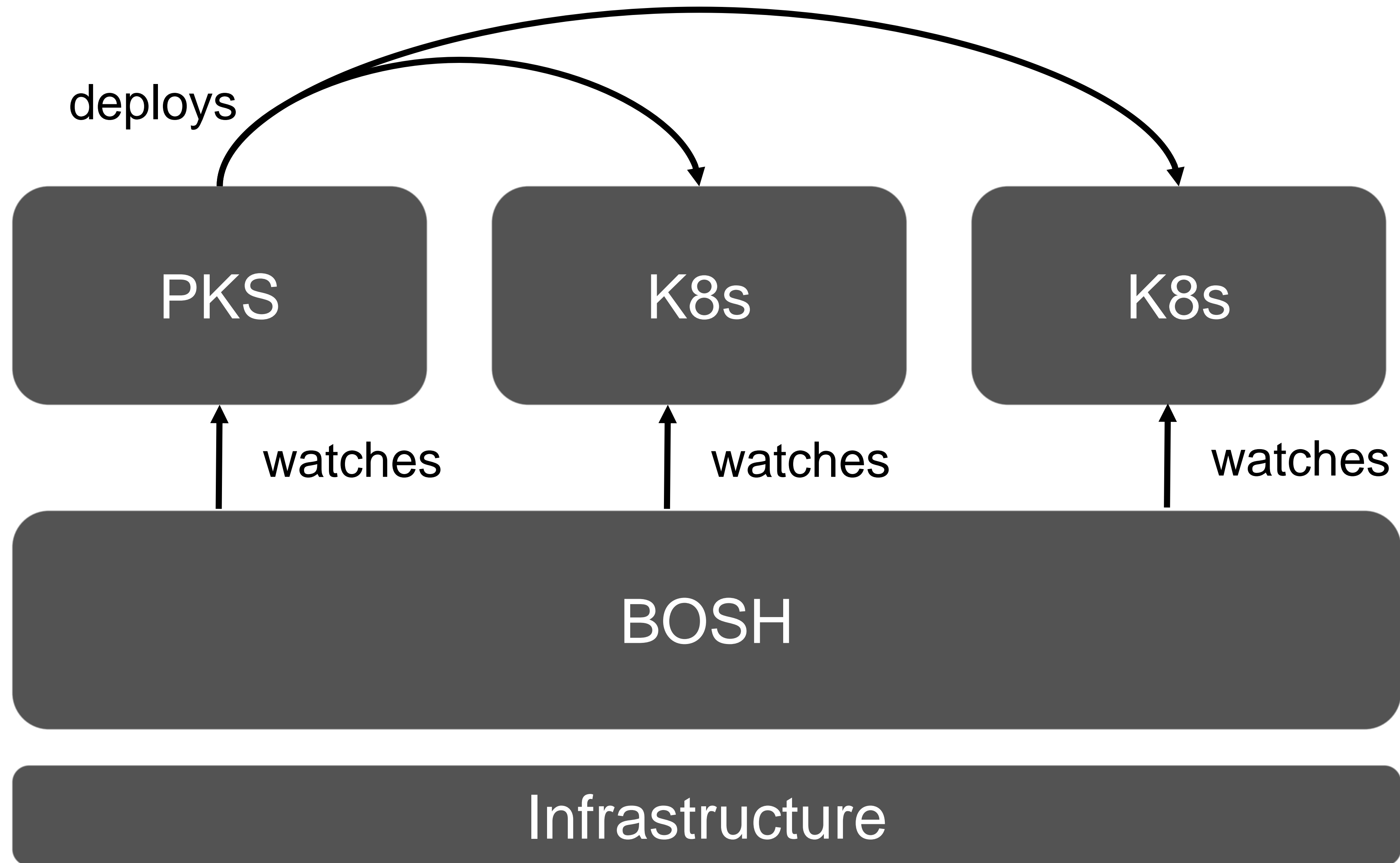
Infrastructure











PKS

- Deploys as a tile in Ops Manager



Import a Product

Installation Dashboard

STEMCELL LIBRARY



Google Cloud Platform

v2.1-build.304



Pivotal Container Service

v1.42.0-1536767118



Pending Changes

Revert

UPDATE Pivotal Container Service

Apply changes

Changelog

Download PCF compatible products at
[Pivotal Network](#)

Delete All Unused Products

PKS

- Deploys as a tile in Ops Manager



PKS

PKS

- Deploys as a tile in Ops Manager
- Interact via the `pks cli`

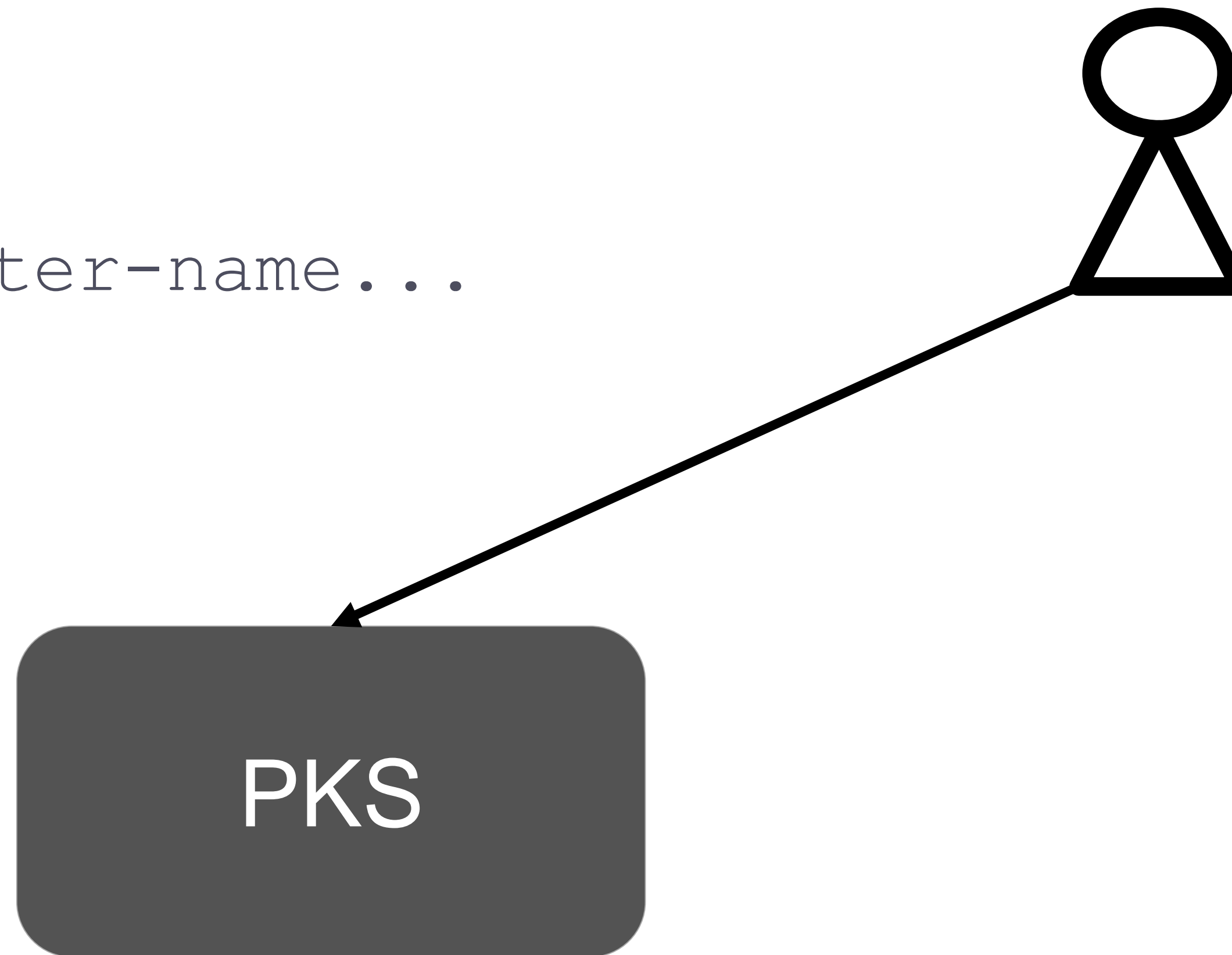


PKS

PKS

- Deploys as a tile in Ops Manager
- Interact via the `pks cli`

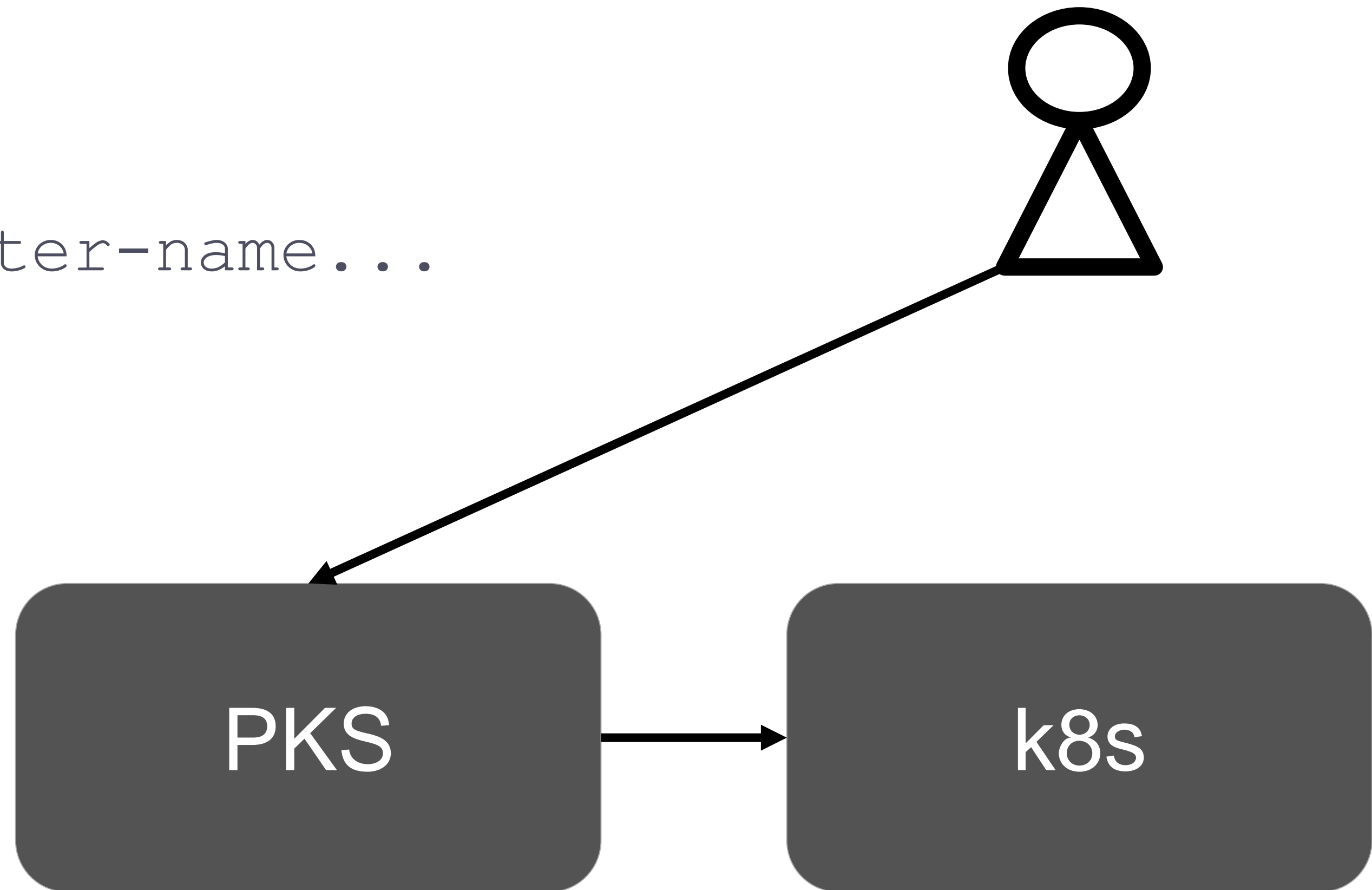
```
pks create-cluster cluster-name...
```



PKS

- Deploys as a tile in Ops Manager
- Interact via the `pks cli`

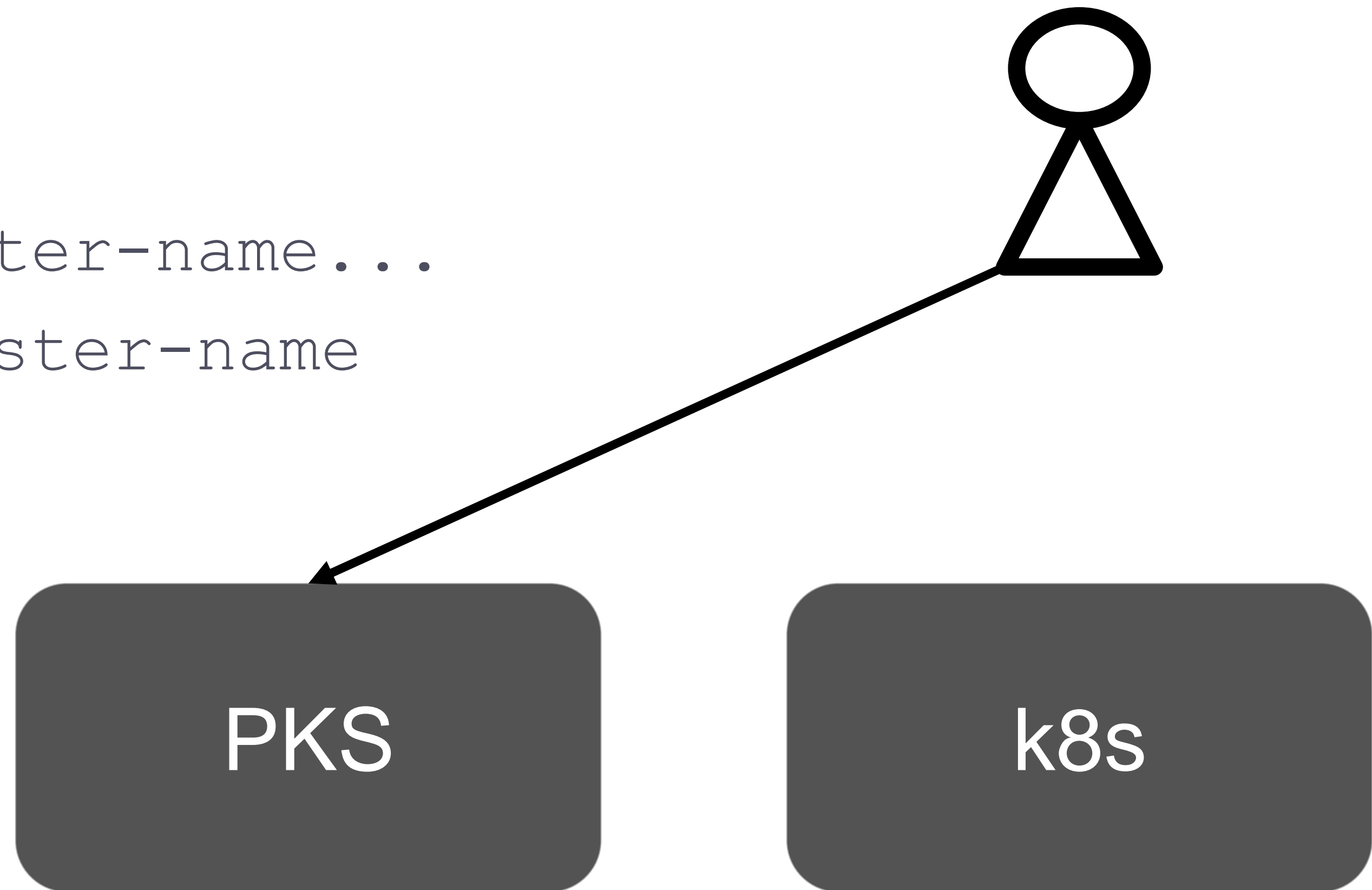
```
pks create-cluster cluster-name...
```



PKS

- Deploys as a tile in Ops Manager
- Interact via the `pks cli`

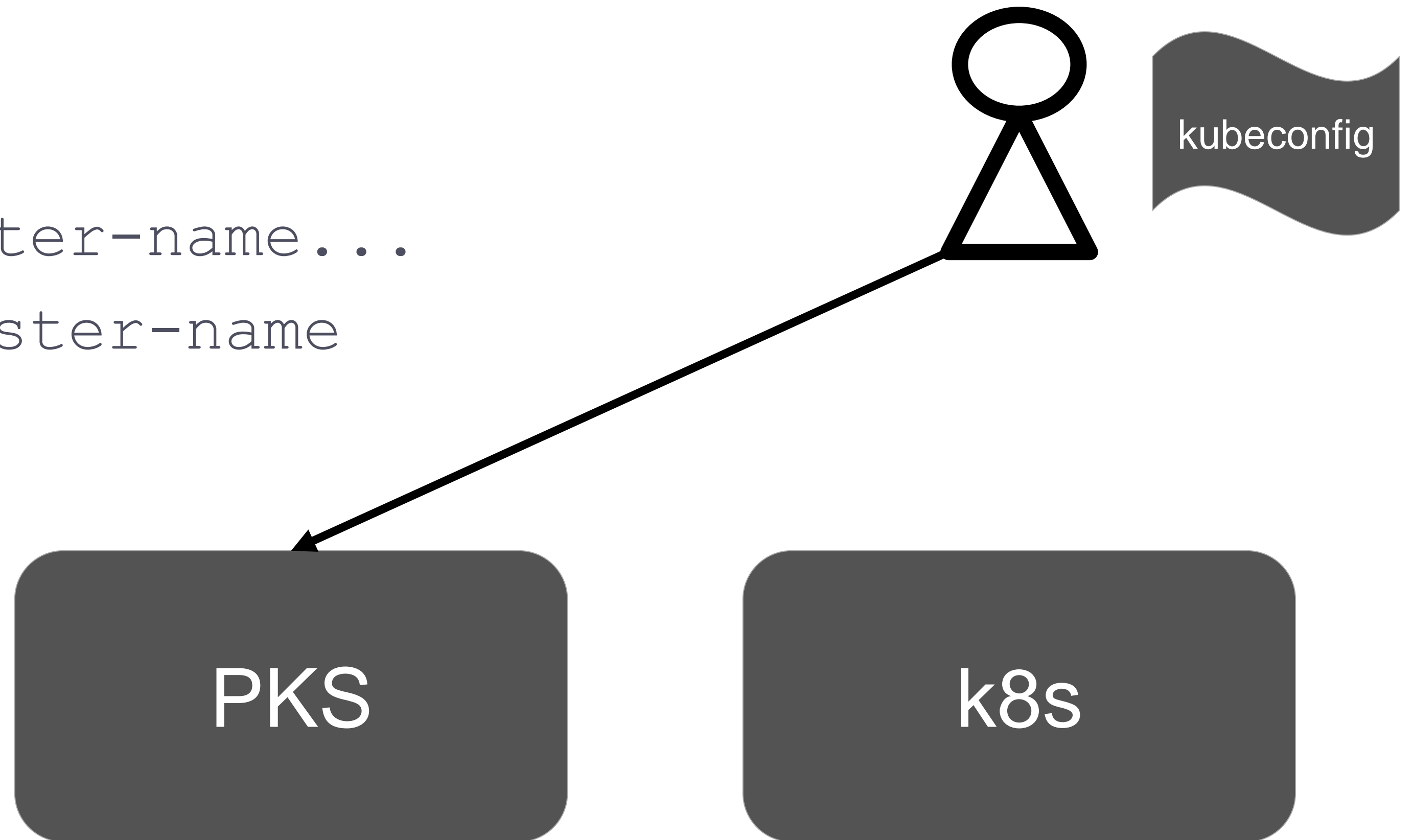
```
pks create-cluster cluster-name...  
pks get-credentials cluster-name
```



PKS

- Deploys as a tile in Ops Manager
- Interact via the `pks cli`

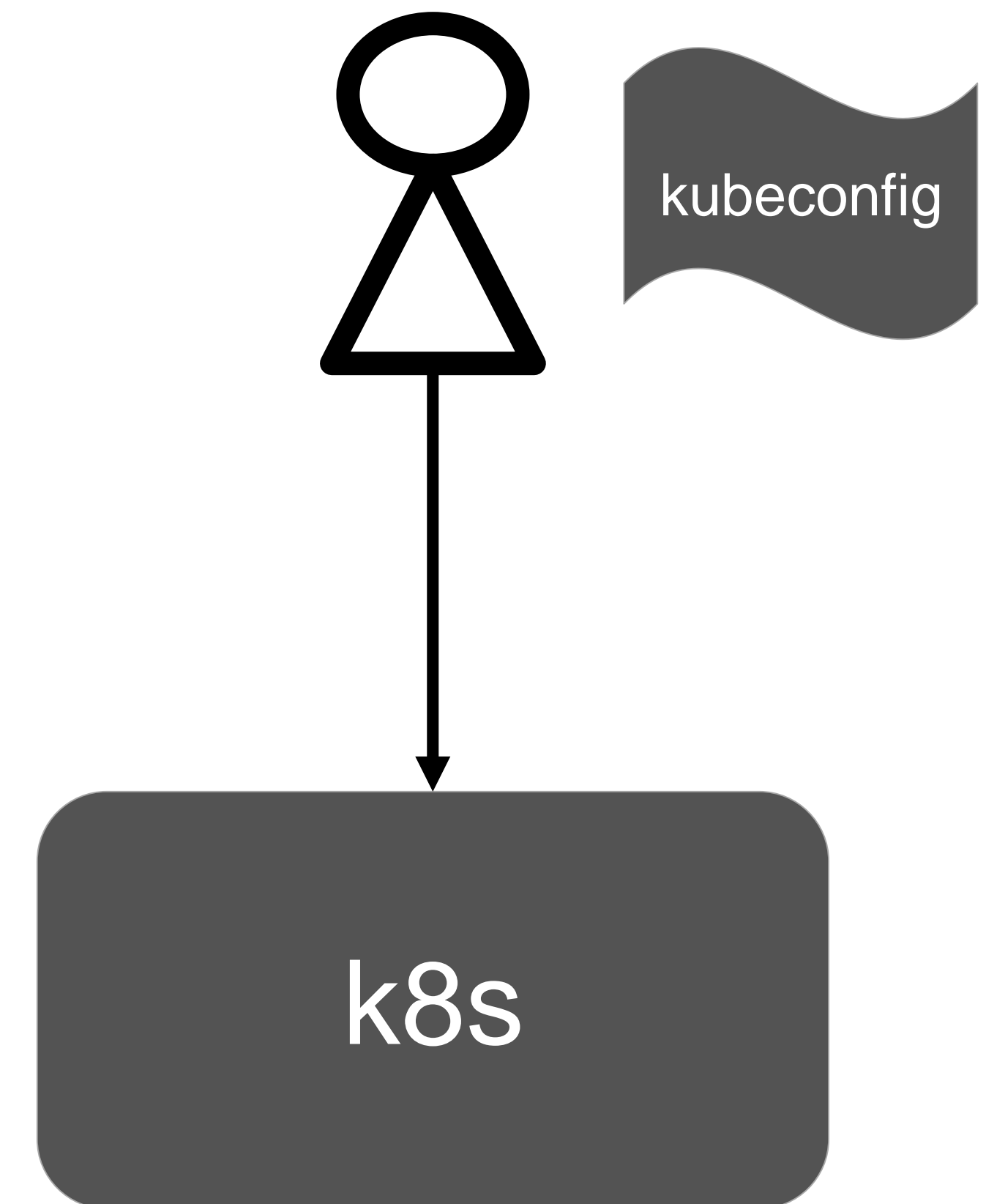
```
pks create-cluster cluster-name...  
pks get-credentials cluster-name
```



PKS

- Deploys as a tile in Ops Manager
- Interact via the `pks cli`

```
pks create-cluster cluster-name...  
pks get-credentials cluster-name  
kubectl get pods
```



Networking in Kubernetes

Networking

- CNI (Container Network Interface): choose your own adventure

Networking

- CNI (Container Network Interface): choose your own adventure
- Baseline Flannel for pod communication

Networking

- CNI (Container Network Interface): choose your own adventure
- Baseline Flannel for pod communication
- Ingress and LoadBalancers for ingress traffic

Networking

- CNI (Container Network Interface): choose your own adventure
- Baseline Flannel for pod communication
- Ingress and LoadBalancers for ingress traffic
- NetworkPolicy for enforcing network rules

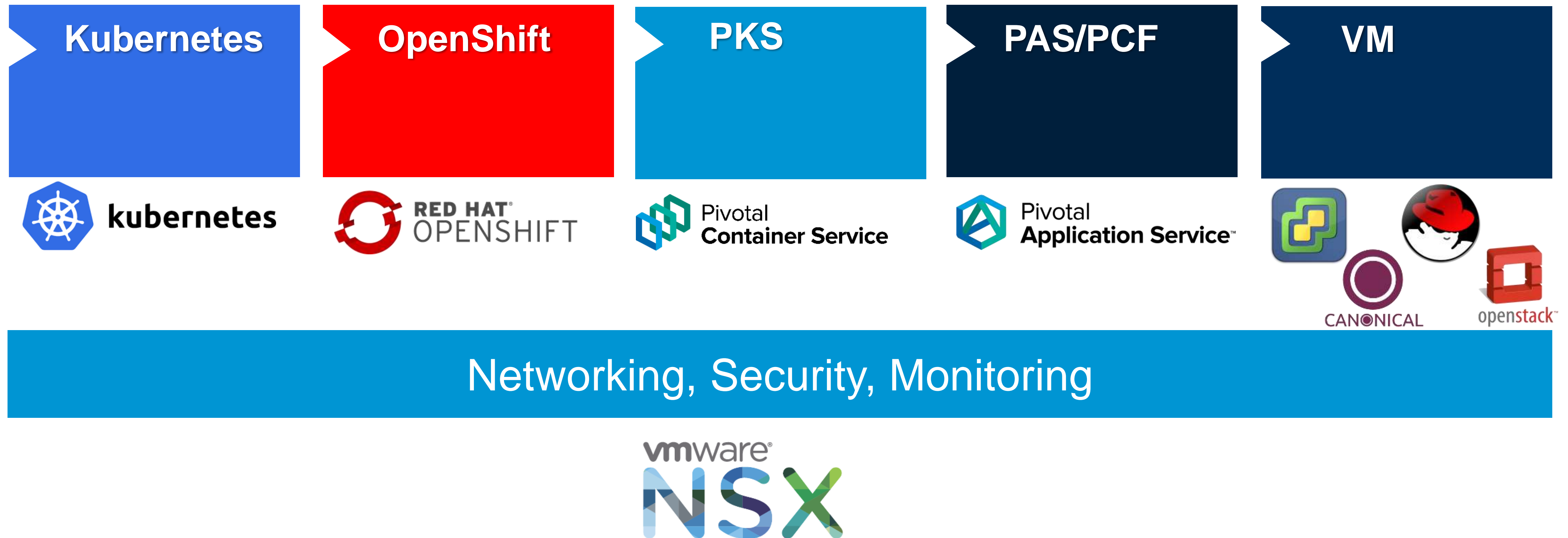
Networking

- CNI (Container Network Interface): choose your own adventure
- Baseline Flannel for pod communication
- Ingress and LoadBalancers for ingress traffic
- NetworkPolicy for enforcing network rules
 - ...but not provided with Flannel alone

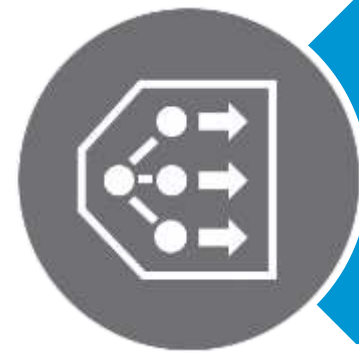
What is NSX-T ?

The NSX-T Platform

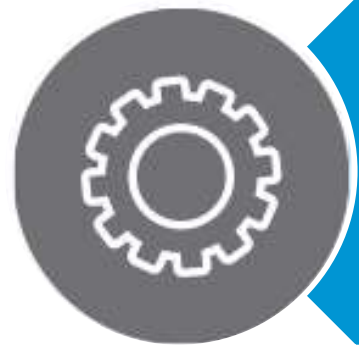
Single platform for Networking, Security & Monitoring for Containers & VMs



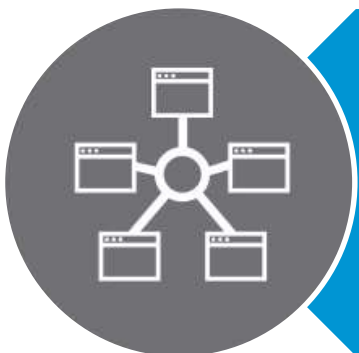
How does PKS with NSX-T add value over opensource solutions



Address all K8s Networking Functions
Load Balancing, IPAM, Routing, Firewalling



Complete automation and dynamic provisioning of
Network Objects required for K8s and Workloads



Support for different topology choices for
Pod and Node Networks (NAT/No-NAT)



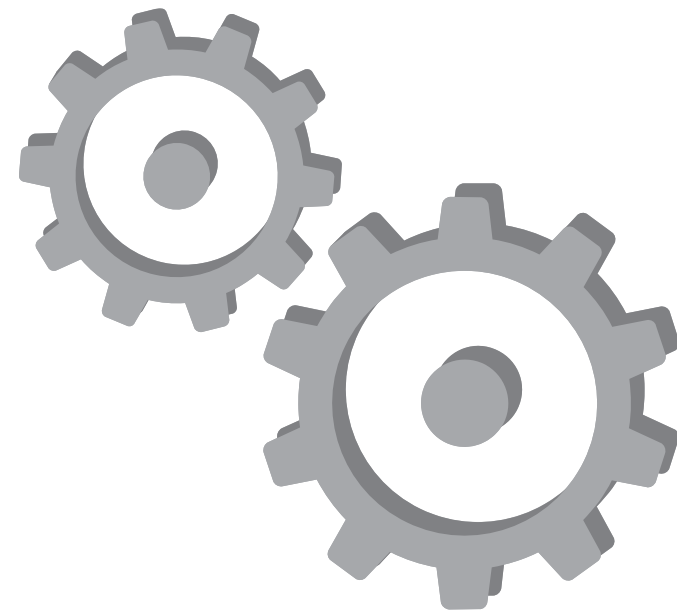
Network Security Policies for Kubernetes Clusters,
Namespaces and Individual Services



Full Network traceability/visibility using NSX-T in-built
operational tools for Kubernetes

NSX-T: operational tools and automation

Automation



REST API support

Dynamic libraries available in multiple languages (Java, Python, Terraform)

Upstream OpenStack Support and Partner Ecosystem

Operations



Getting started wizards

Dashboards

IPFIX, Port Mirroring, Port Connection

Granular RBAC

Upgrade Coordinator

vRealize Log Insight / Splunk plugins

Troubleshooting



Traceflow, Port-Connection, Topology View

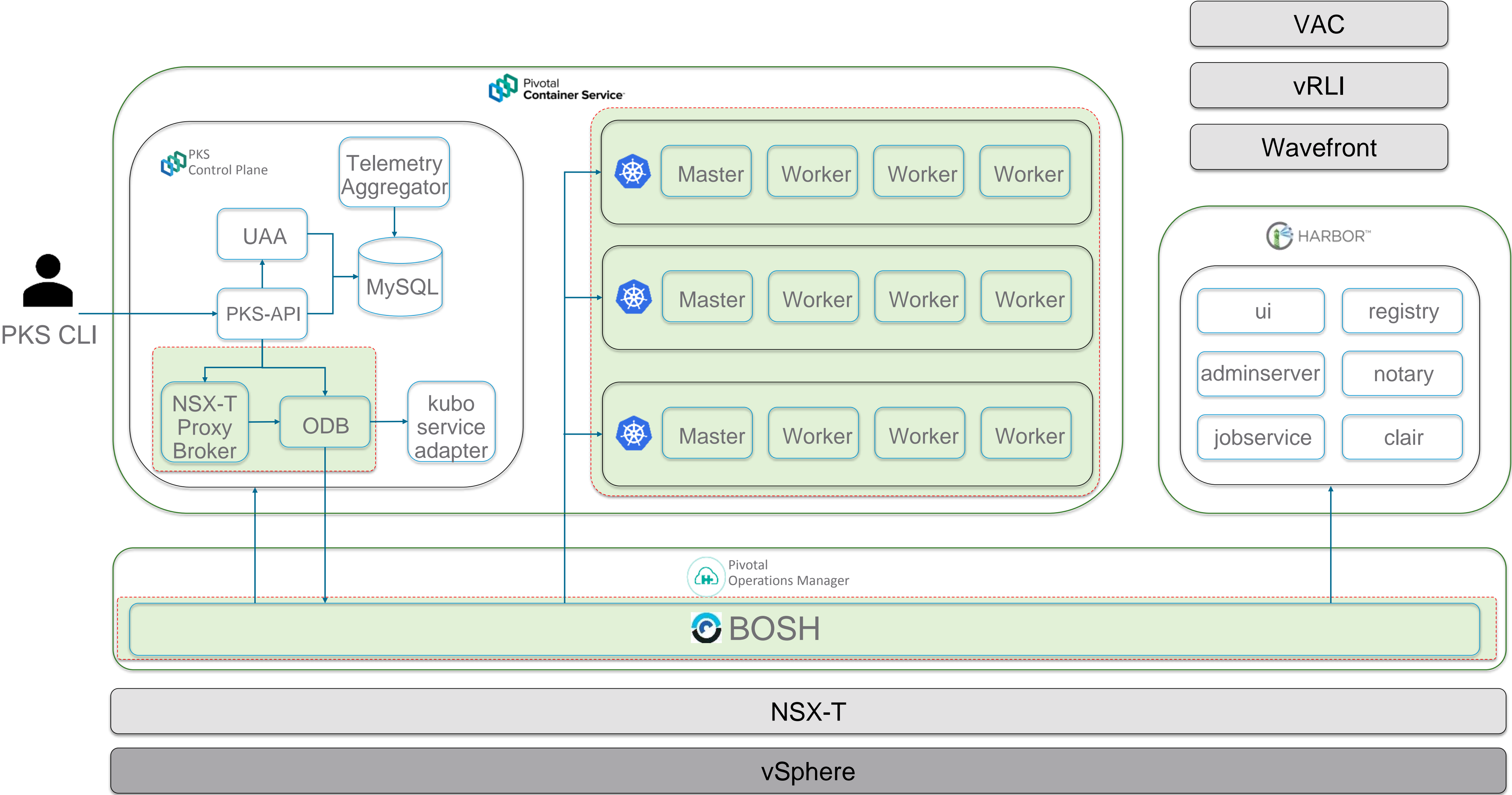
Back-Up & Restore

Selective Tech Support Logs

Monitoring & Stats

NSX-T + PKS

PKS: High Level Architecture

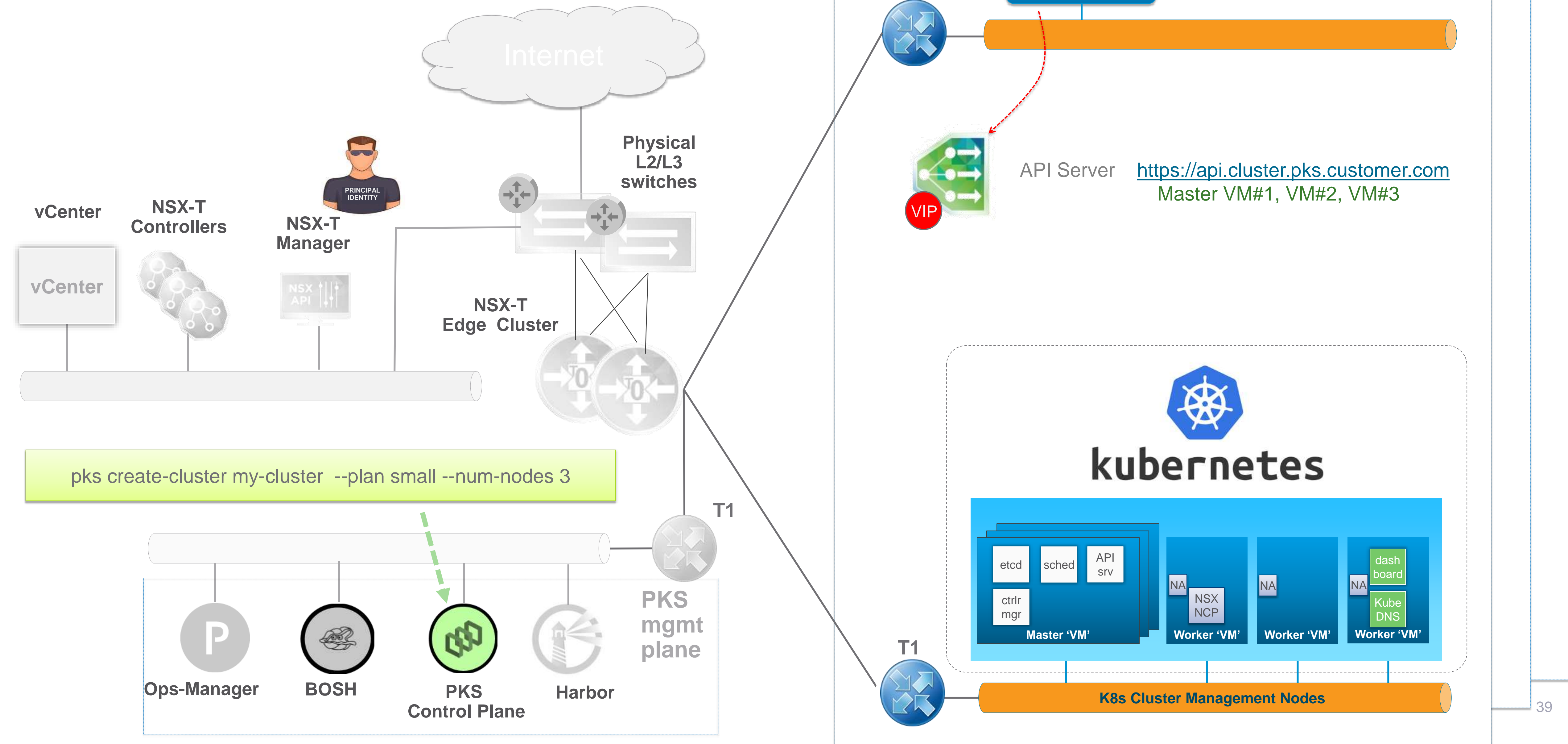


K8s Cluster and NSX-T Provisioning

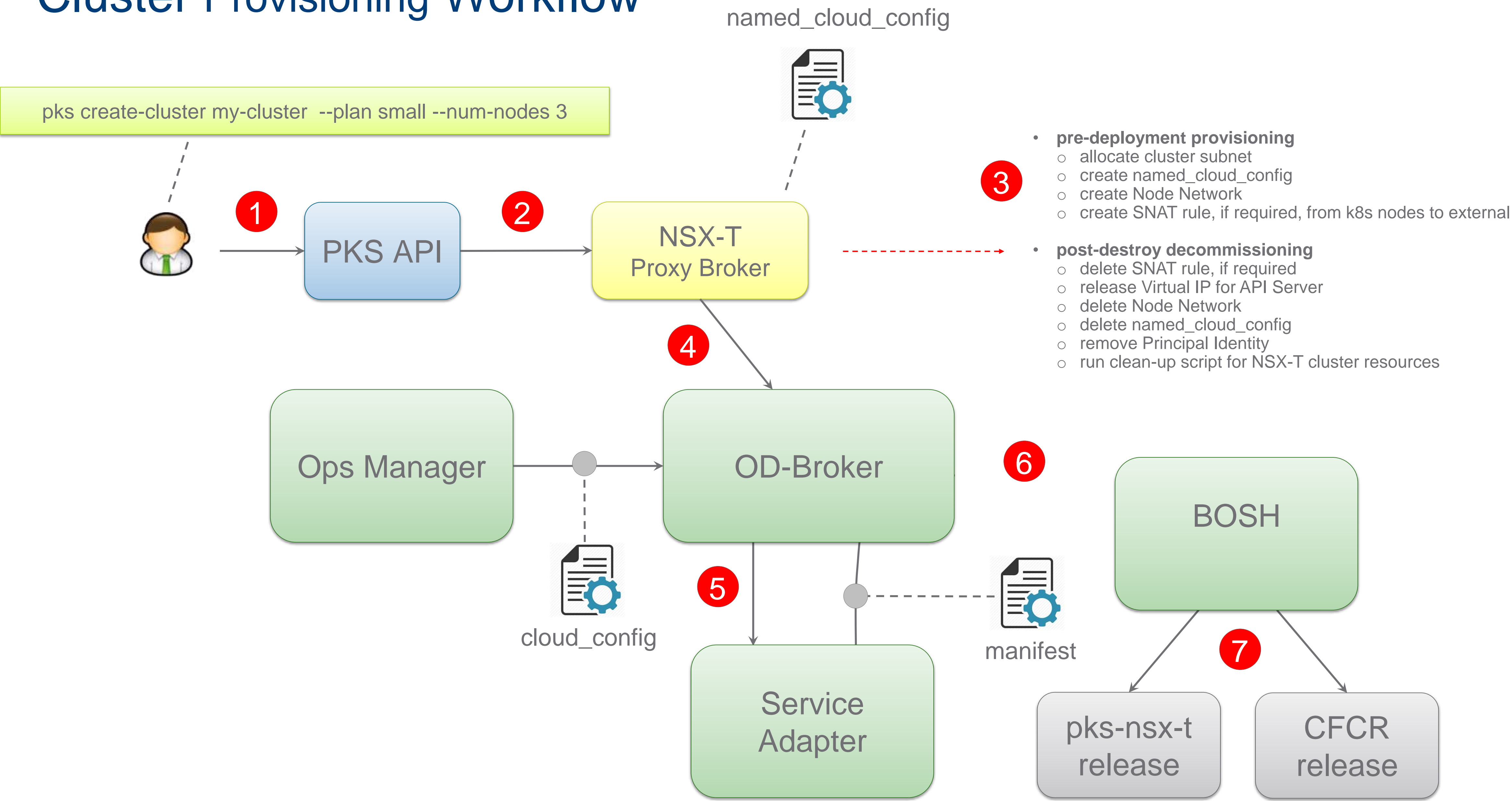
K8s cluster n

K8s cluster 2

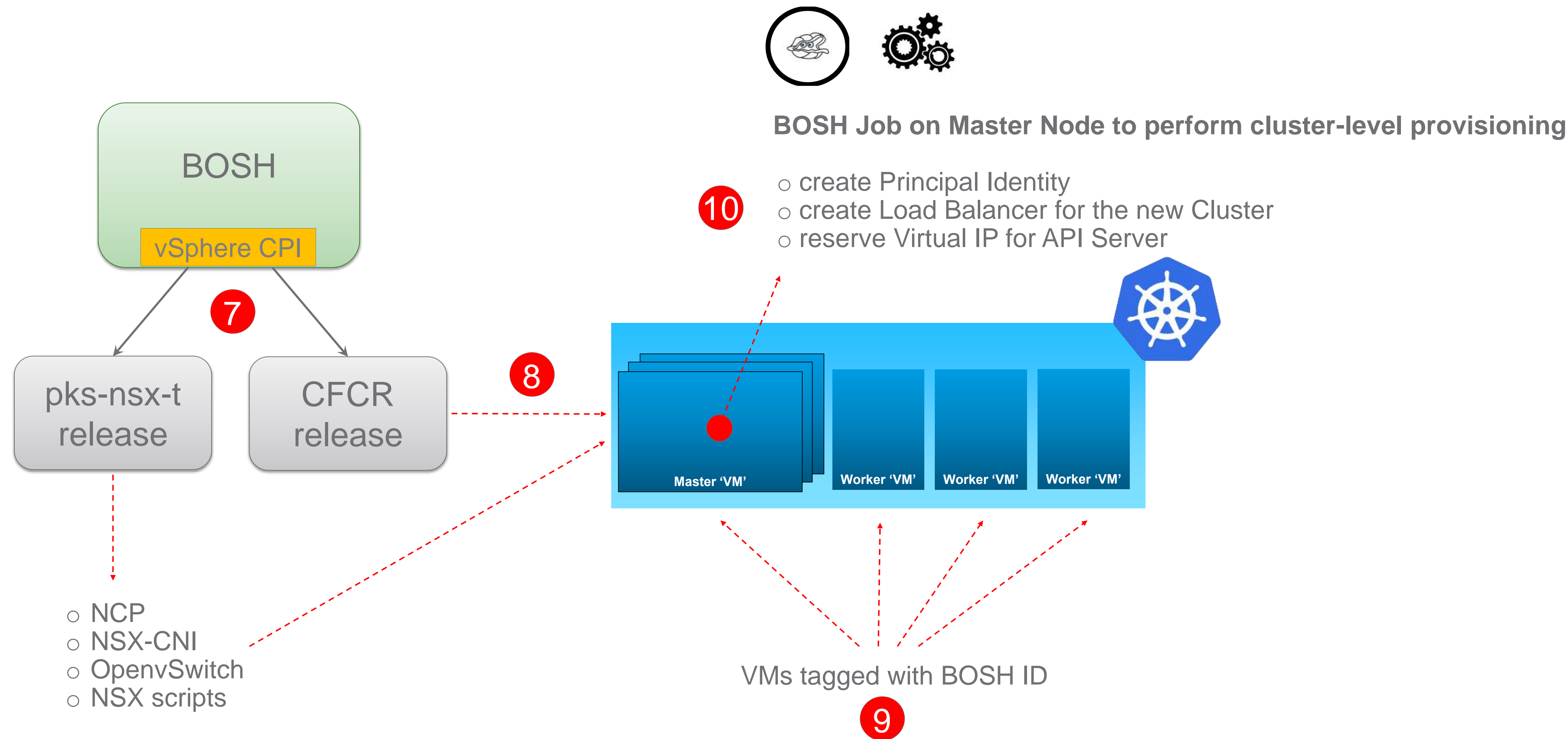
K8s cluster 1



Cluster Provisioning Workflow

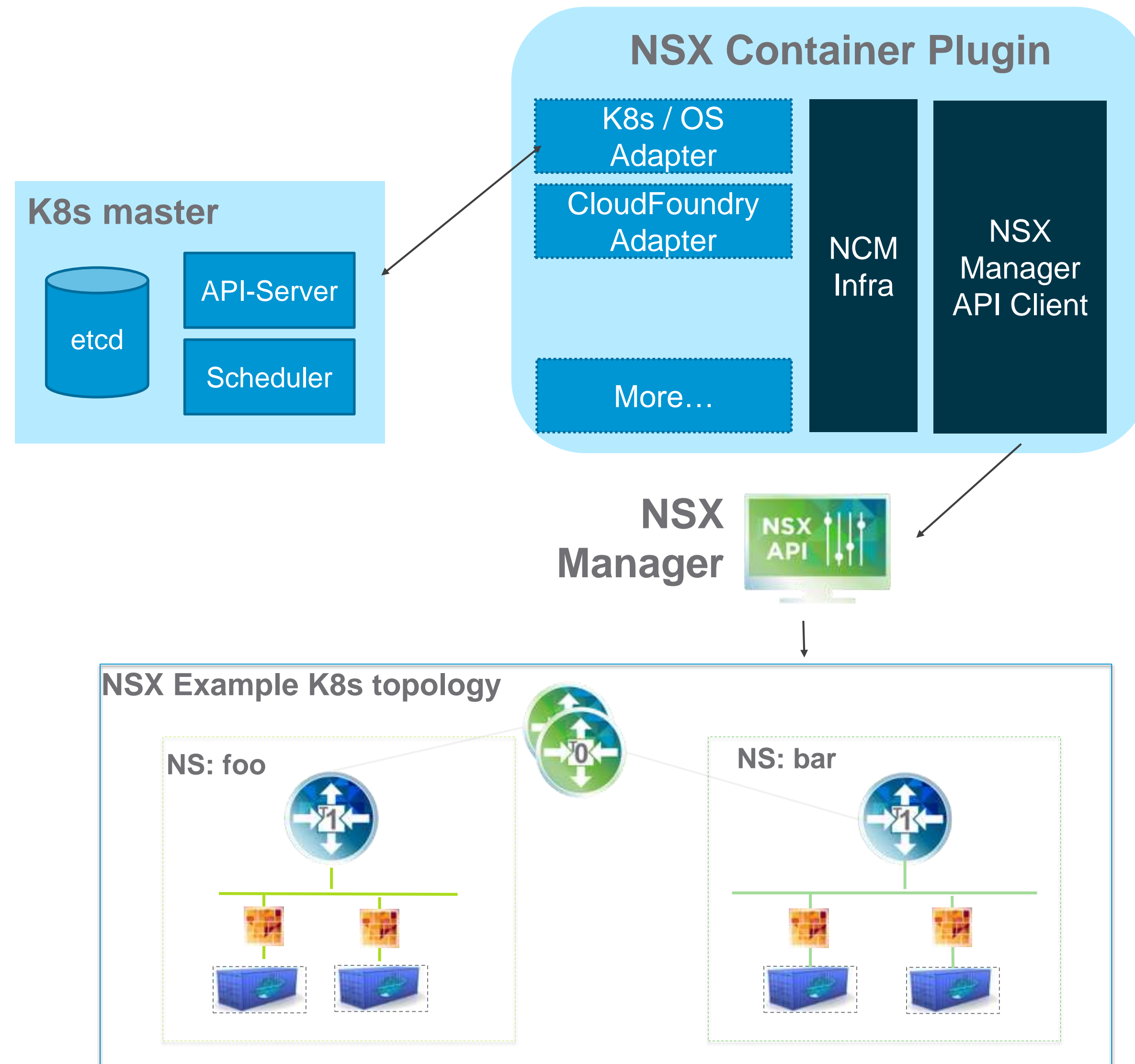


Cluster Provisioning Workflow (cont.)



PKS and NSX-T integration: Key Components

NSX Container Plugin (NCP)

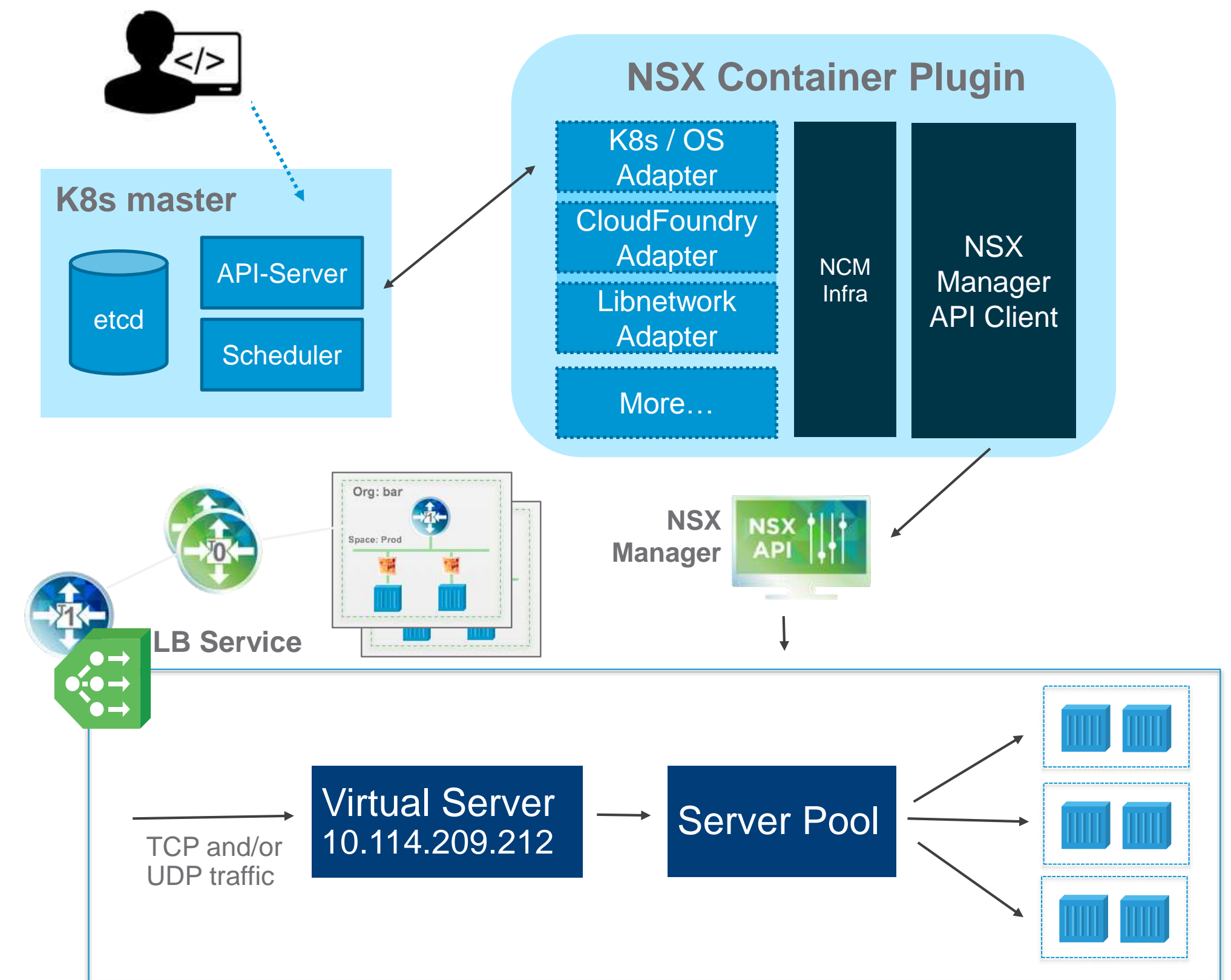
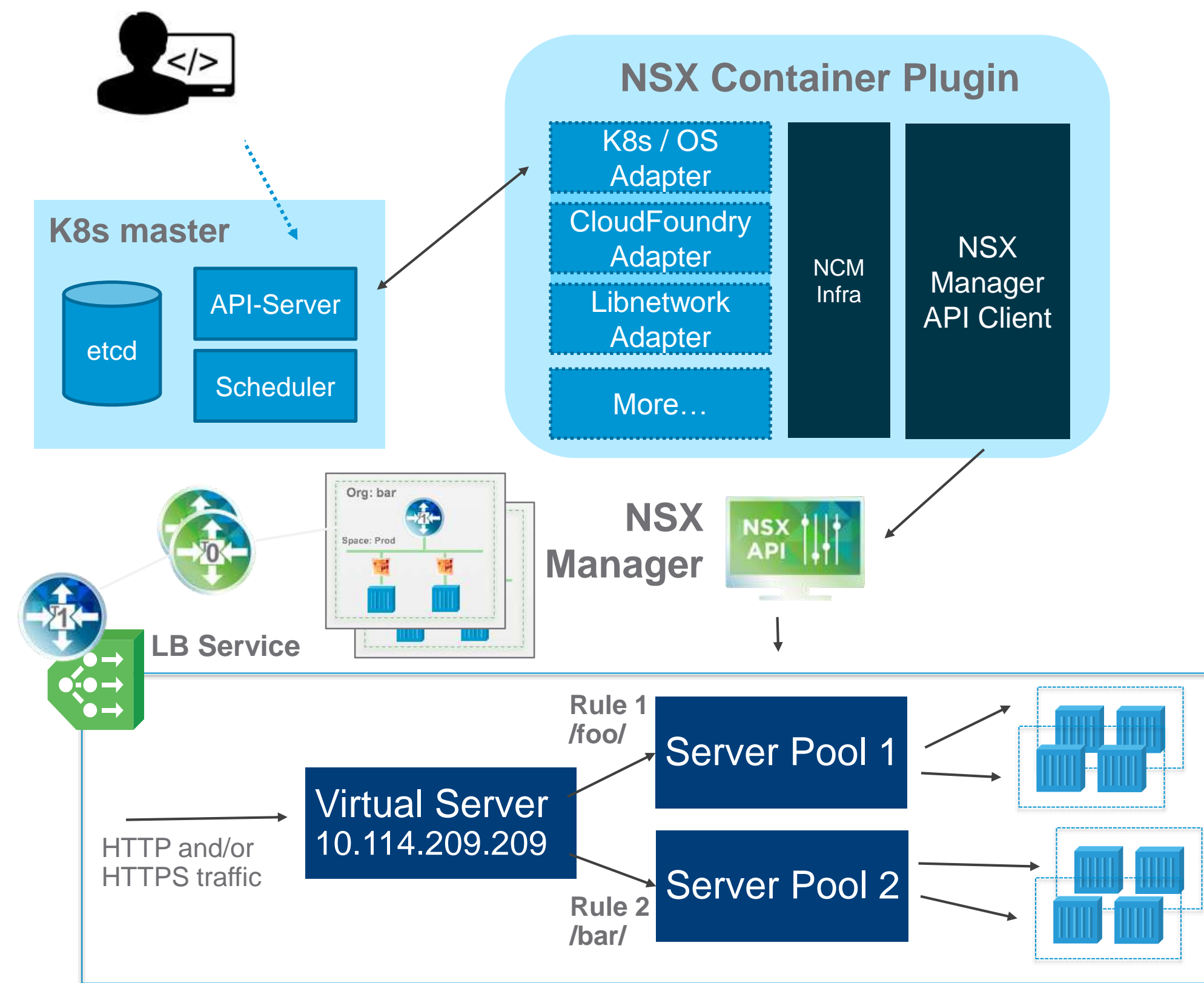


- NCP is a software component provided by VMware, running on a K8s Node..
- NCP is built in a modular way, so that individual adapters can be added for different CaaS and PaaS systems
- NCP 'listens' to Kubernetes/CF to create the required NSX-T constructs.
- [CNI Integration](#) is used.

PKS + NSX-T: built-in Load Balancing

Kubernetes Ingress and Svc Type LB support

Built-in support for Ingress (L7) and Svc Type LB (L4) w/t NSX-T integration. Most other K8s networking choice don't support Svc Type LB (L4), and you need an additional technology like NGINX from Ingress (L7).



PKS + NSX-T: Namespaces

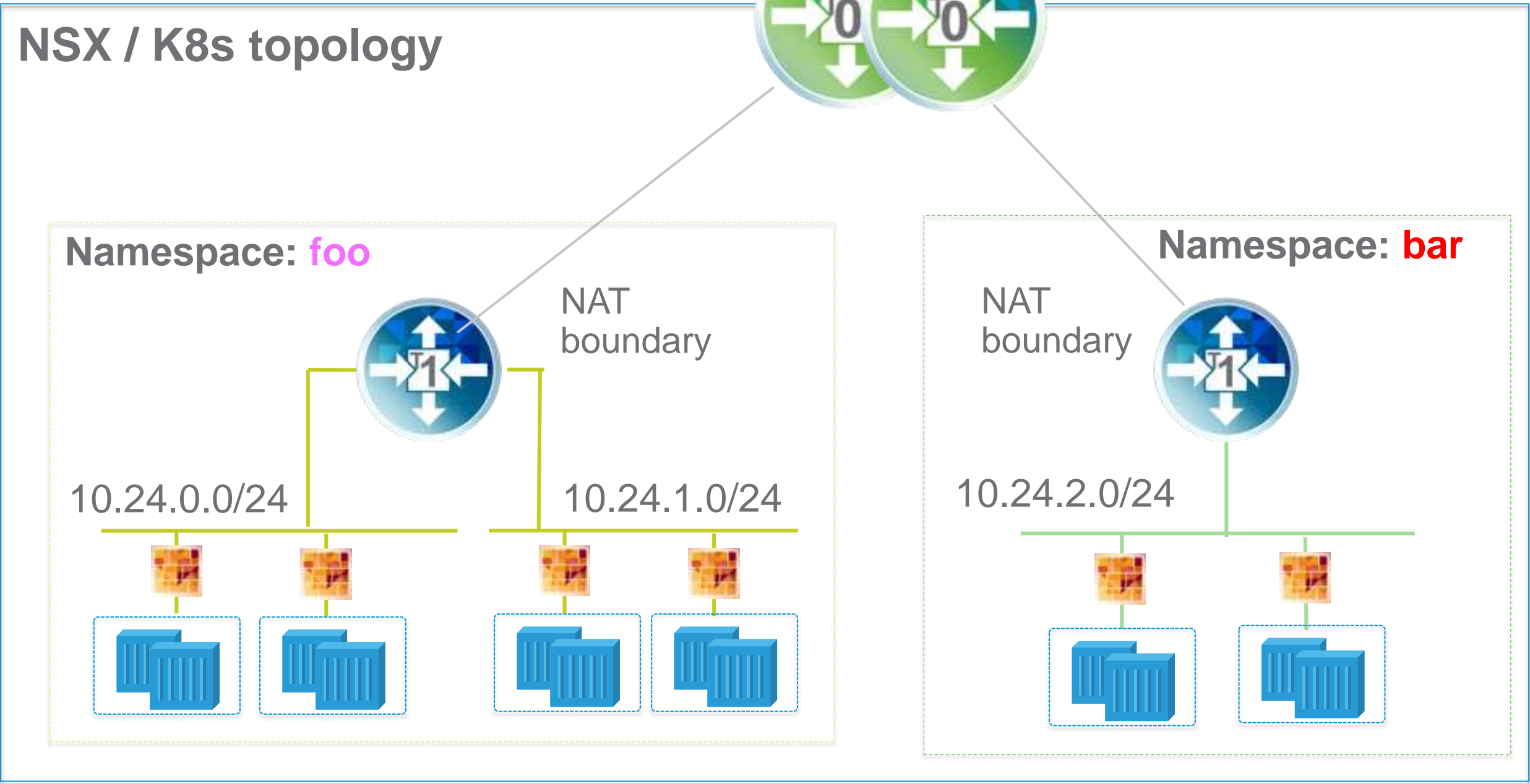
```
1. localadmin@k8s-master: ~ (ssh)

admin@k8s-master:~$ kubectl create namespace foo
namespace "foo" created

admin@k8s-master:~$ kubectl create namespace bar
namespace "bar" created

admin@k8s-master:~$ kubectl run nginx-foo --image=nginx -n foo
deployment "nginx-foo" created

admin@k8s-master:~$ kubectl run nginx-bar --image=nginx -n bar
deployment "nginx-bar" created
```

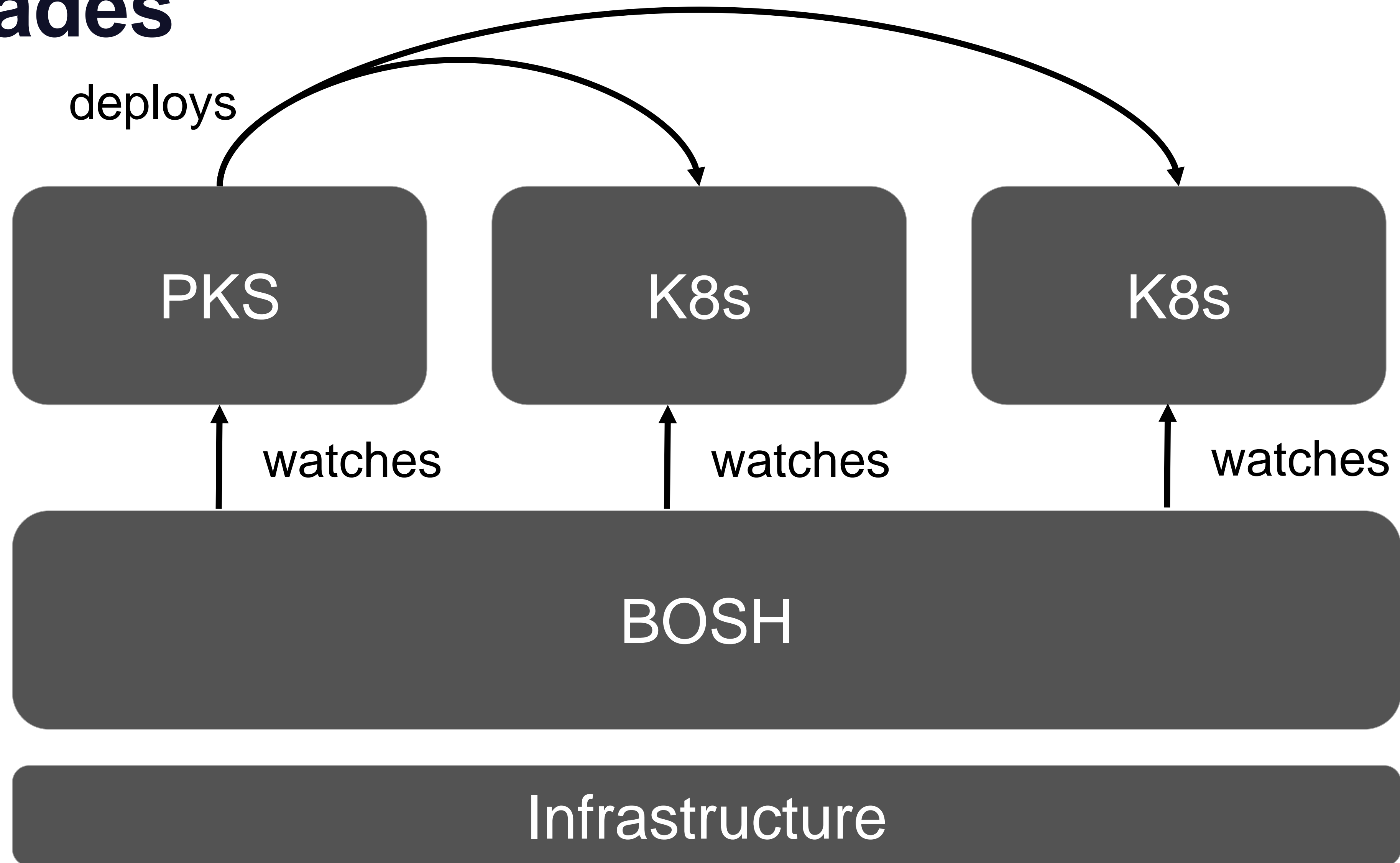


Cluster Upgrades

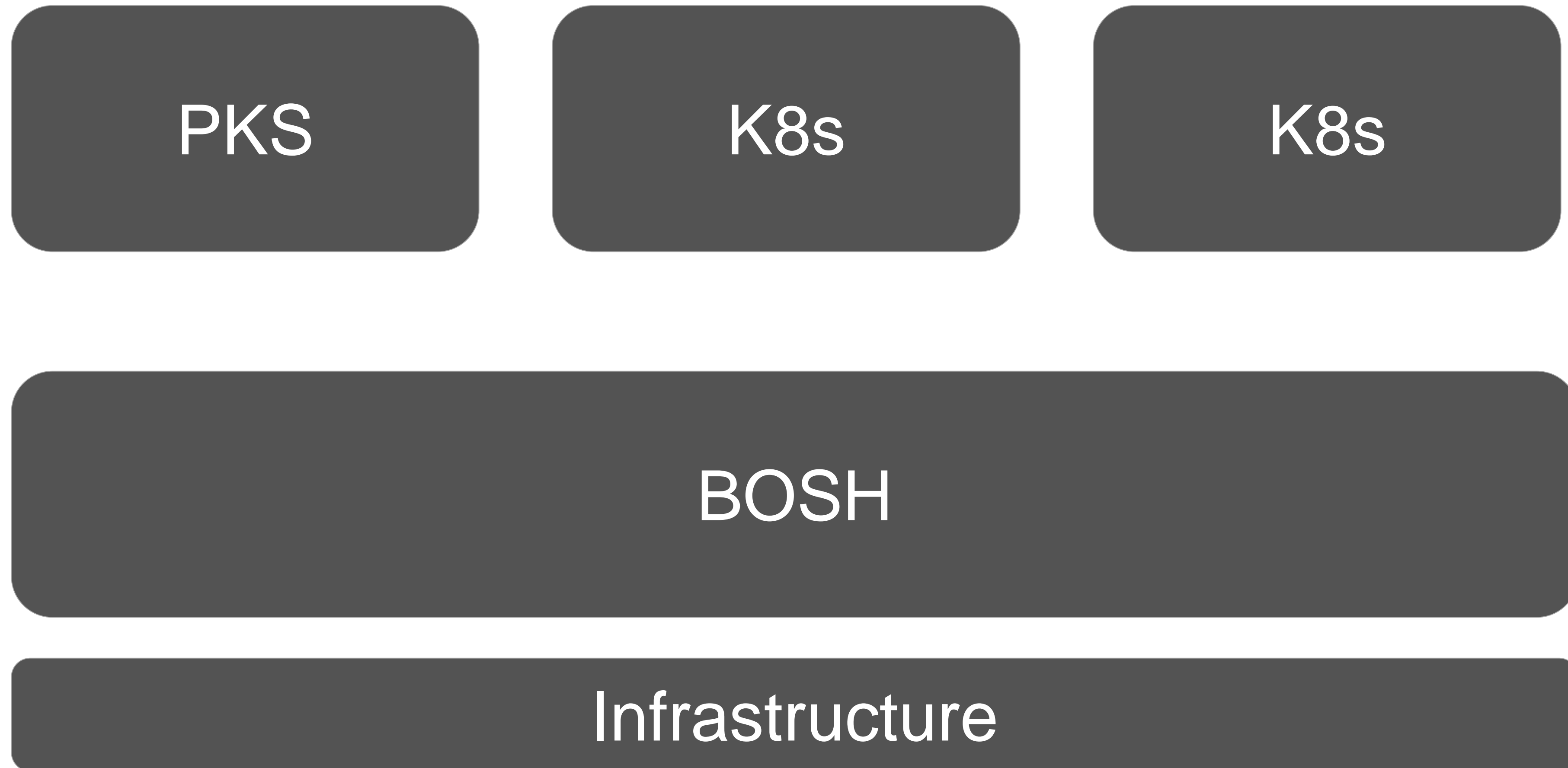
Upgrades

- Install the new tile
- Update any new fields
- Deploy!

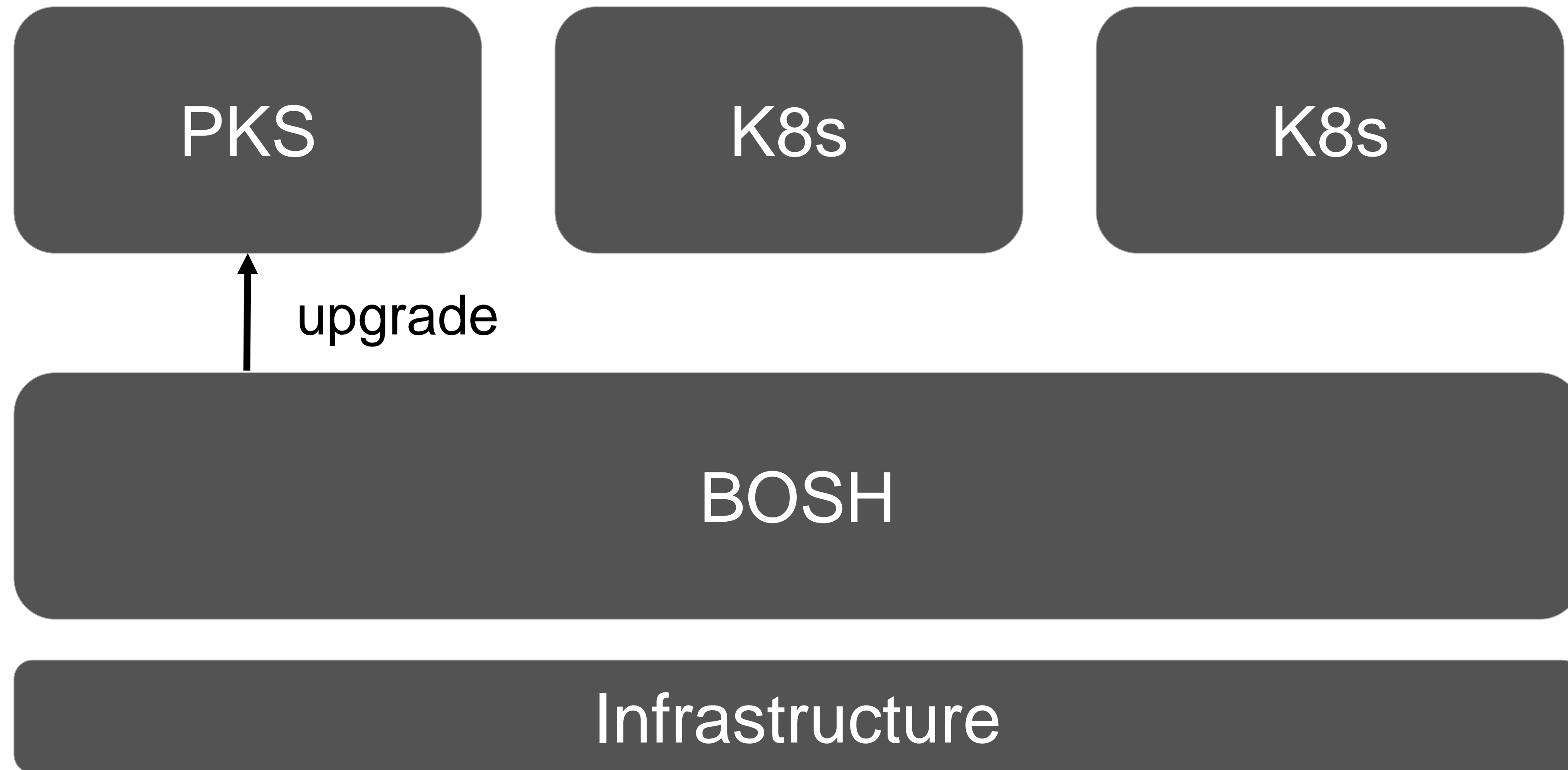
Upgrades



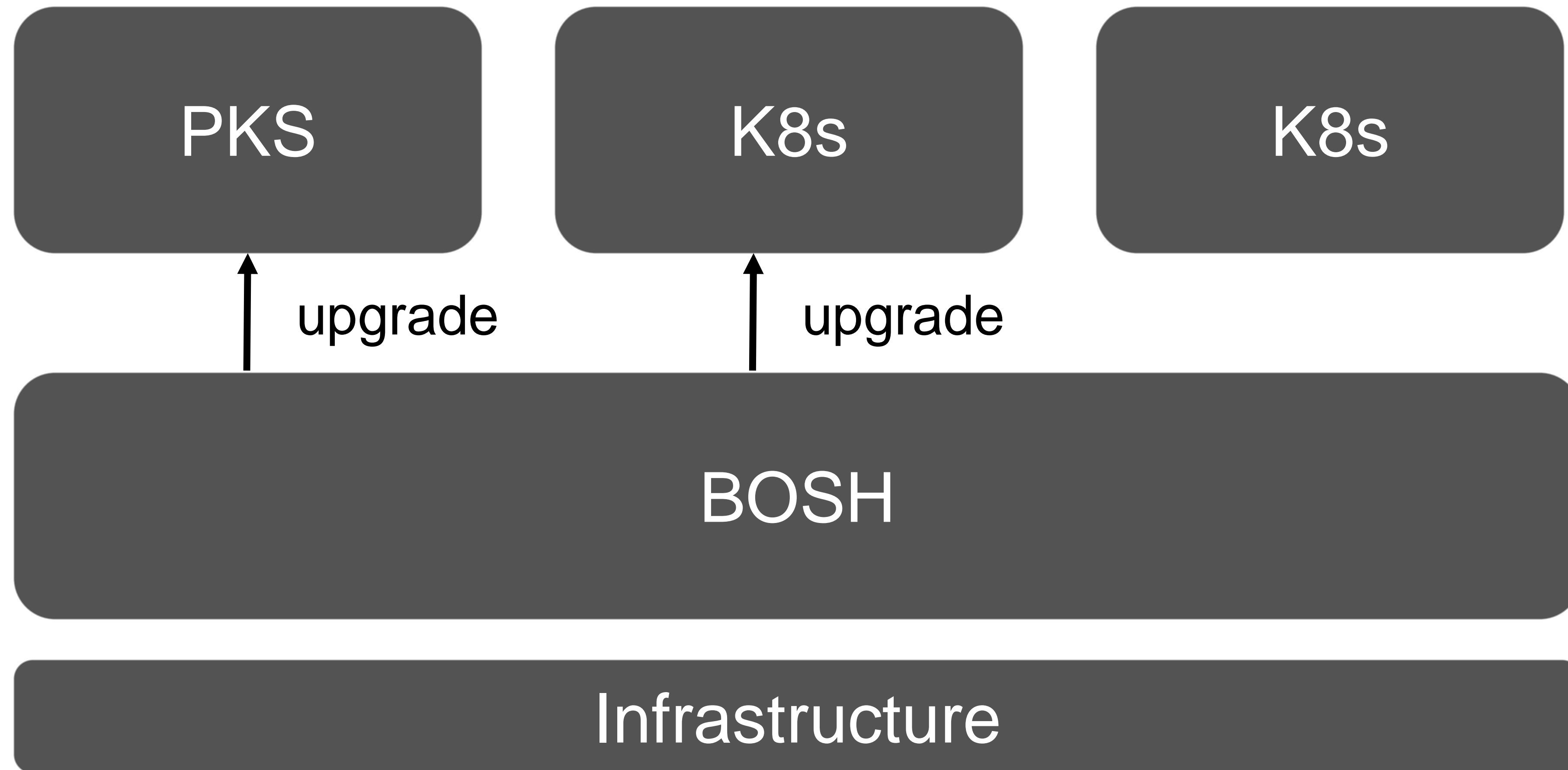
Upgrades



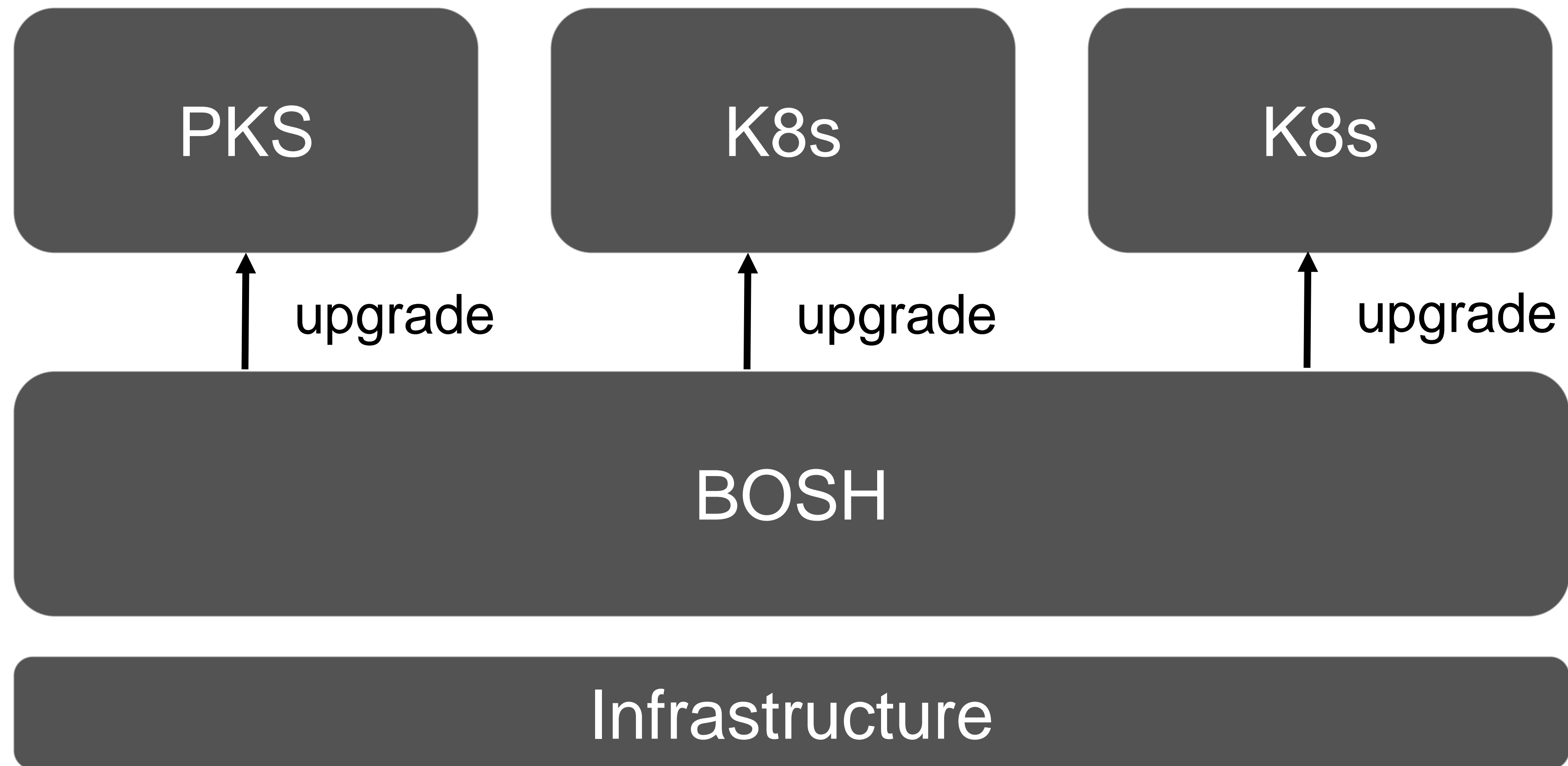
Upgrades



Upgrades



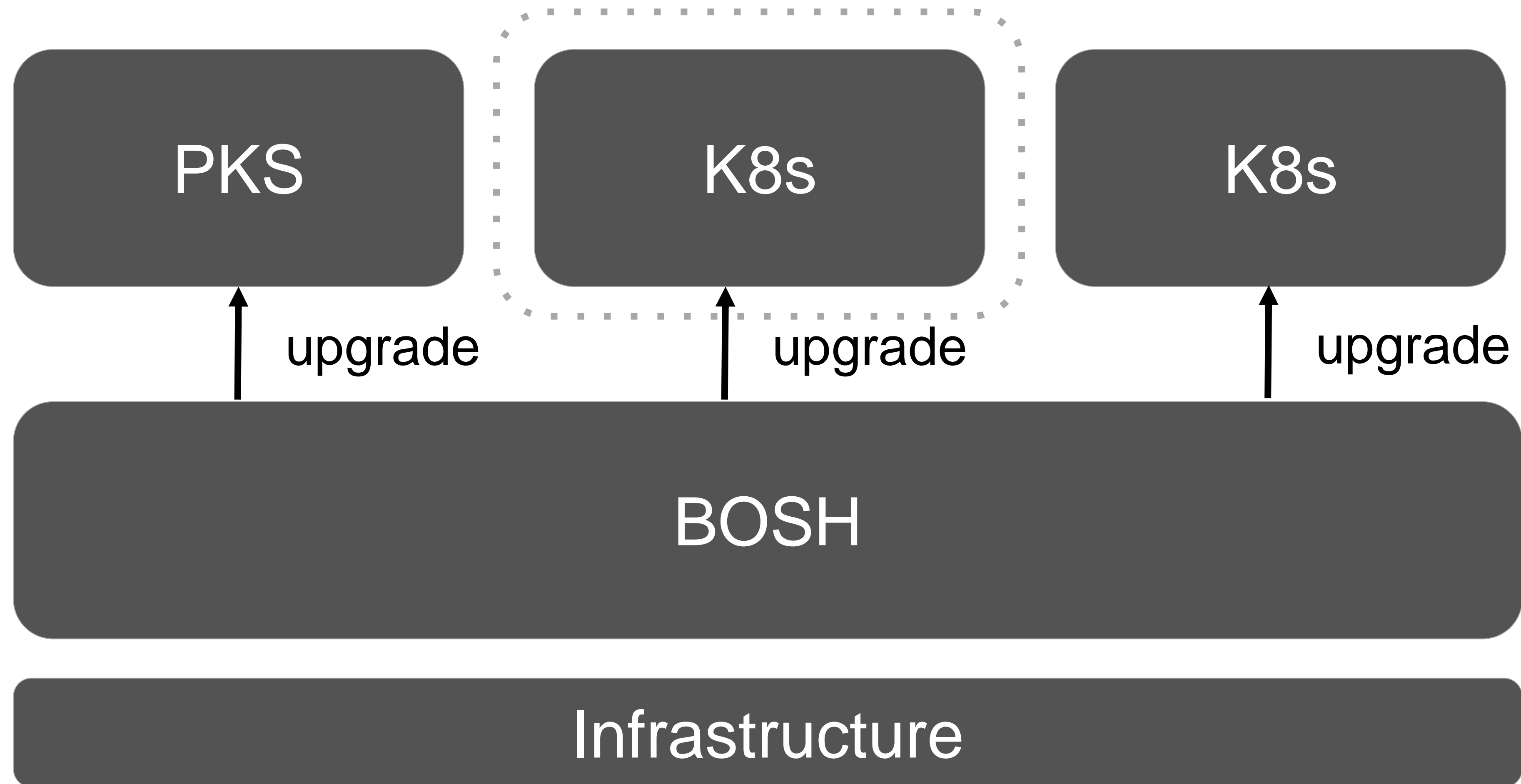
Upgrades



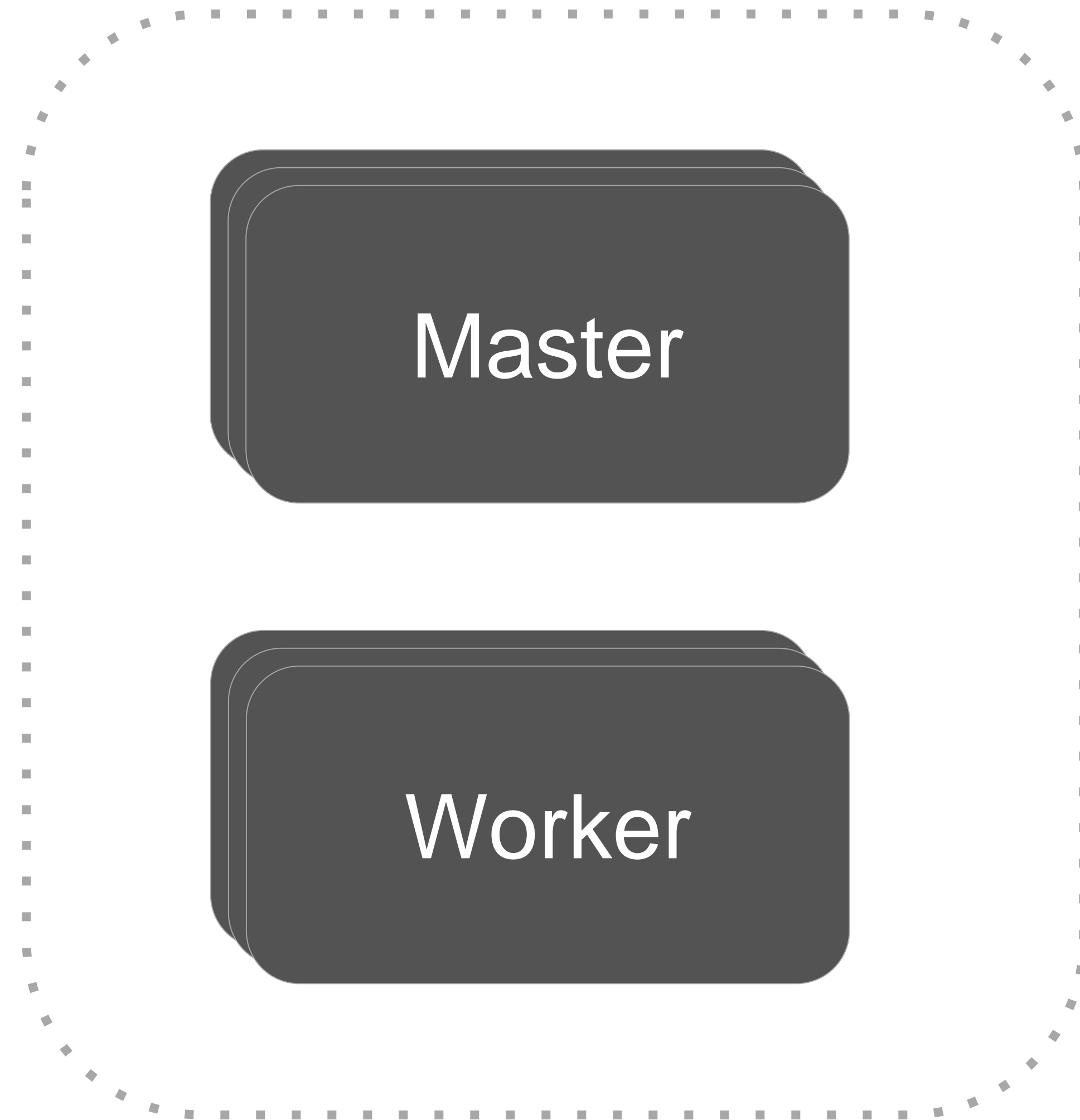
Upgrades

- NCP components upgrade within PKS

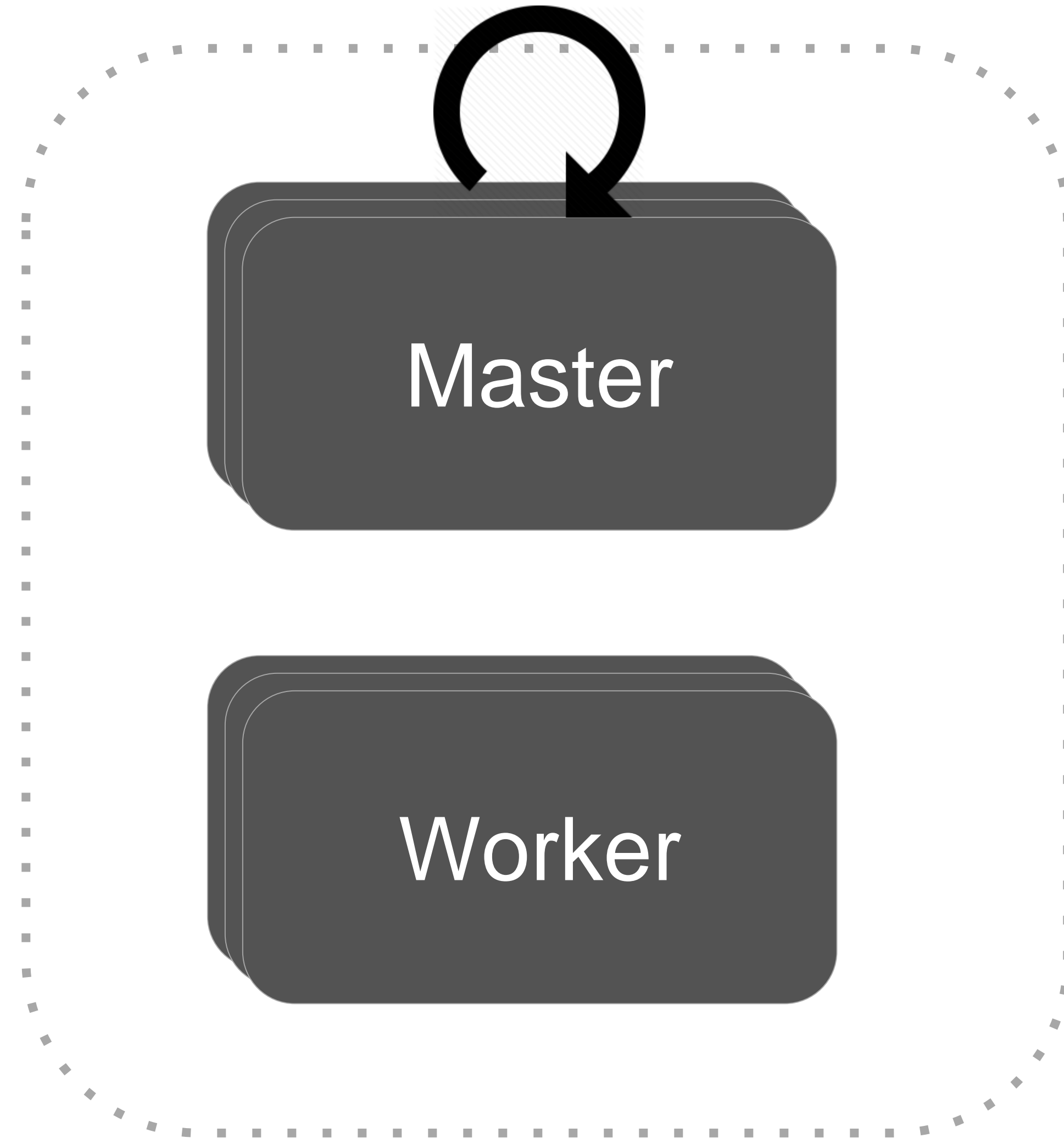
Upgrades



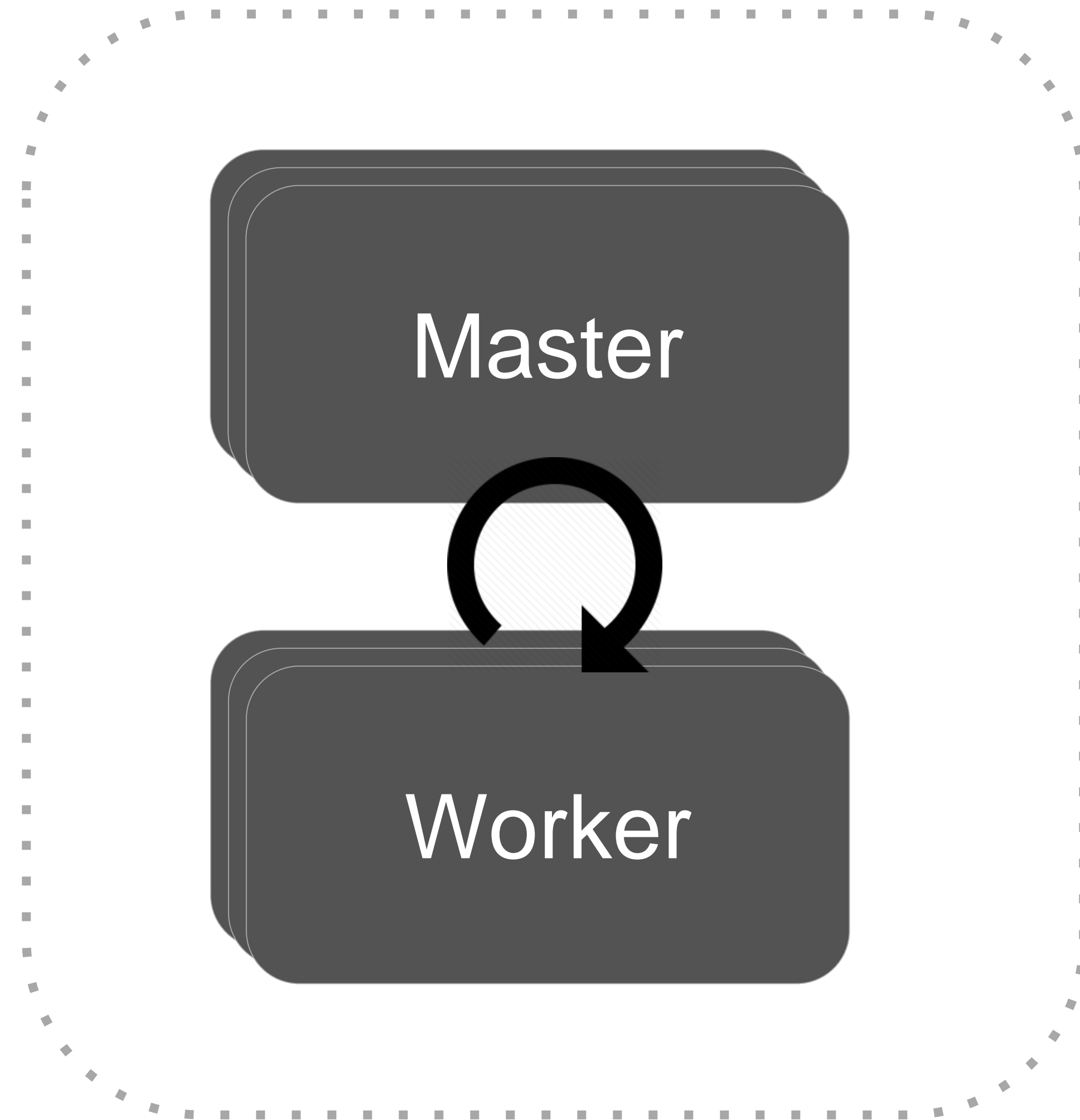
Upgrades



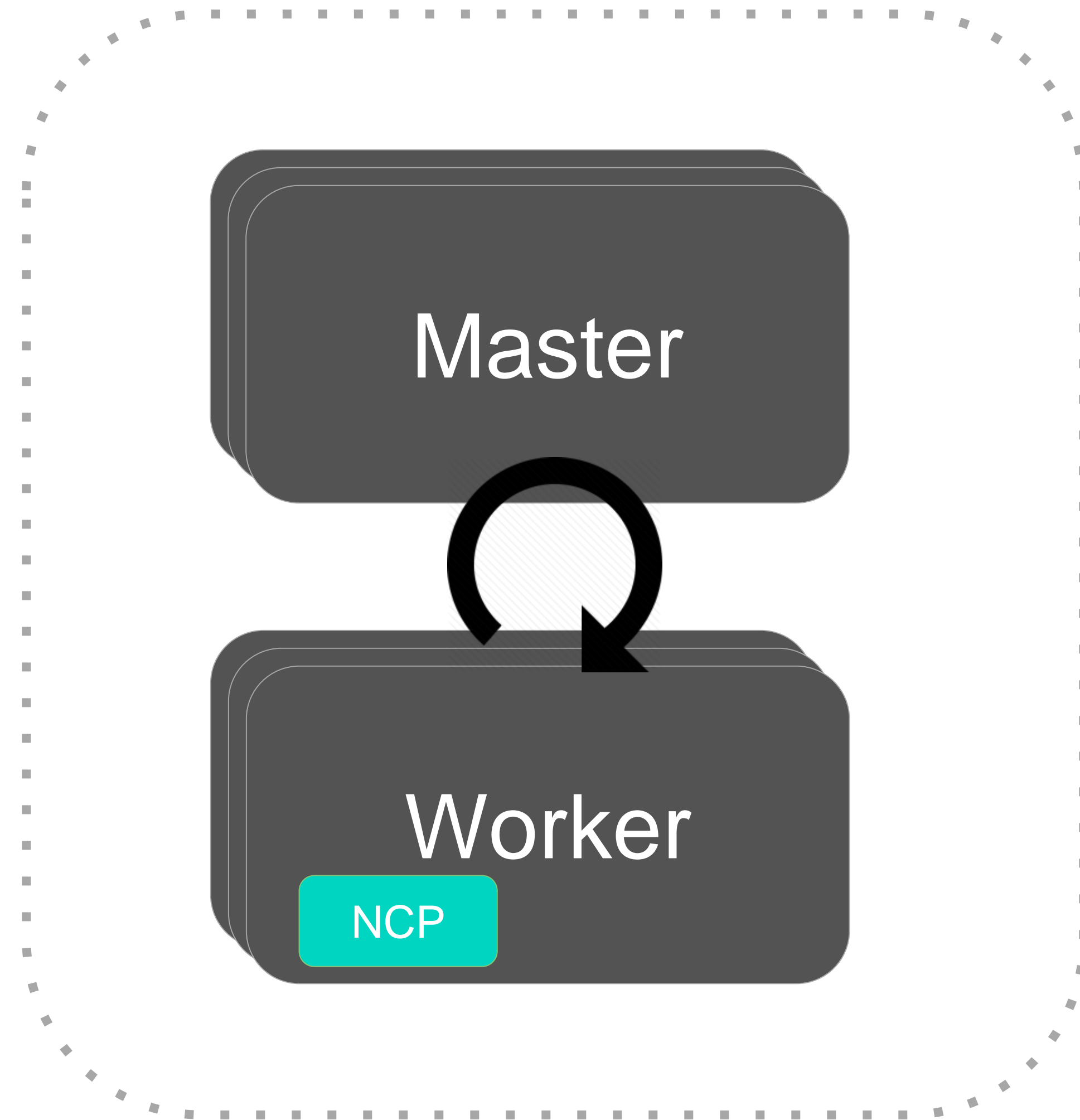
Upgrades



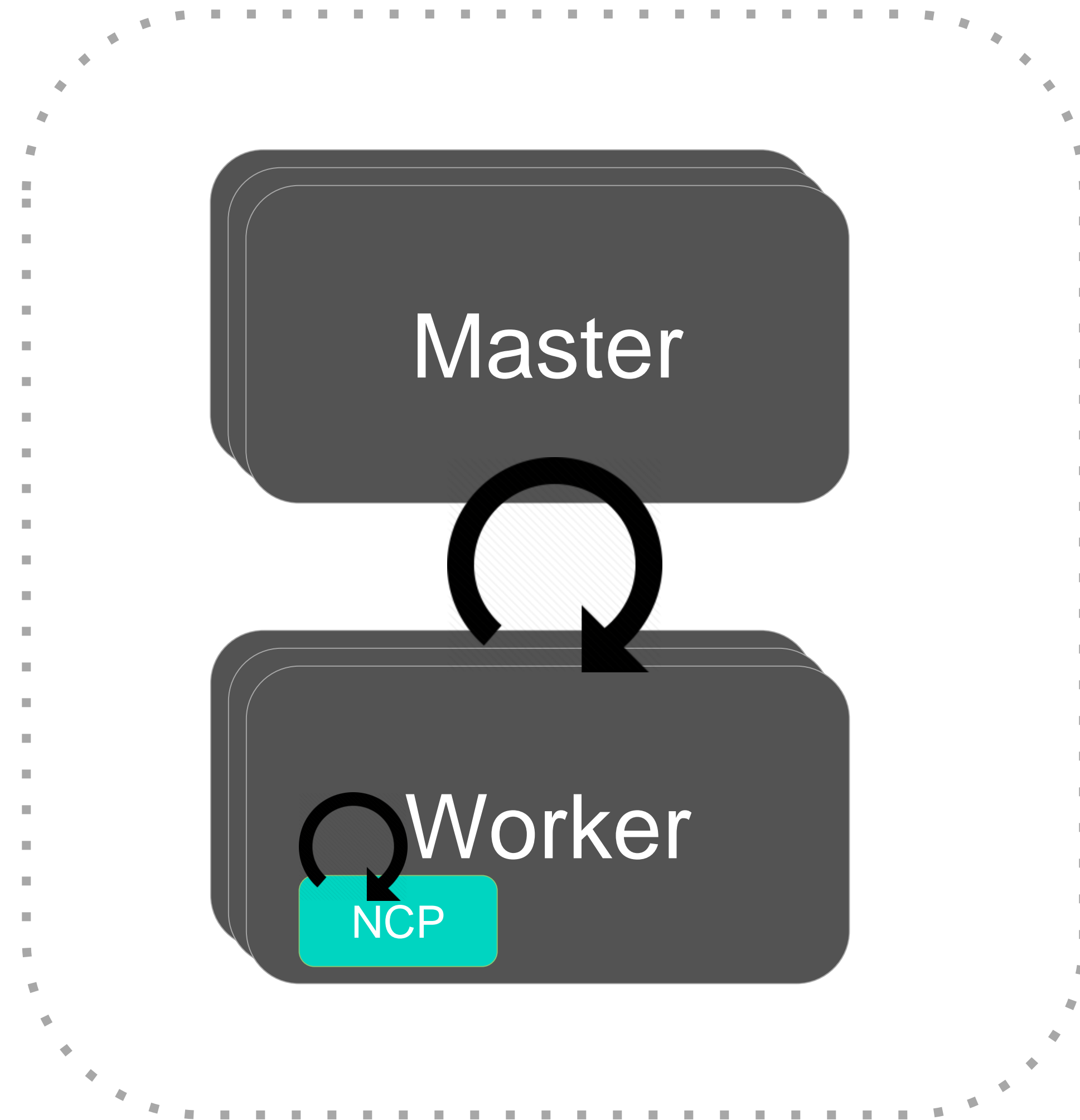
Upgrades



Upgrades



Upgrades



Upgrades

- NCP components upgrade within PKS
- NSX-T can be upgraded independent of PKS