CORE

Managing k8s

moving from the Openstack Magnum to the Cluster API

PCI DSS COMPLIANT

MADE IN LUXEMBOURG

GUINNESS WORLD RECORDS

Powered by intel

About me:

Andrei Novoselov

Edge Cloud system Engineers
Team Lead at Gcore

andrei.novoselov@gcore.com

t.me/andrei_novoselov

GCore Edge Cloud

20+ locations

US

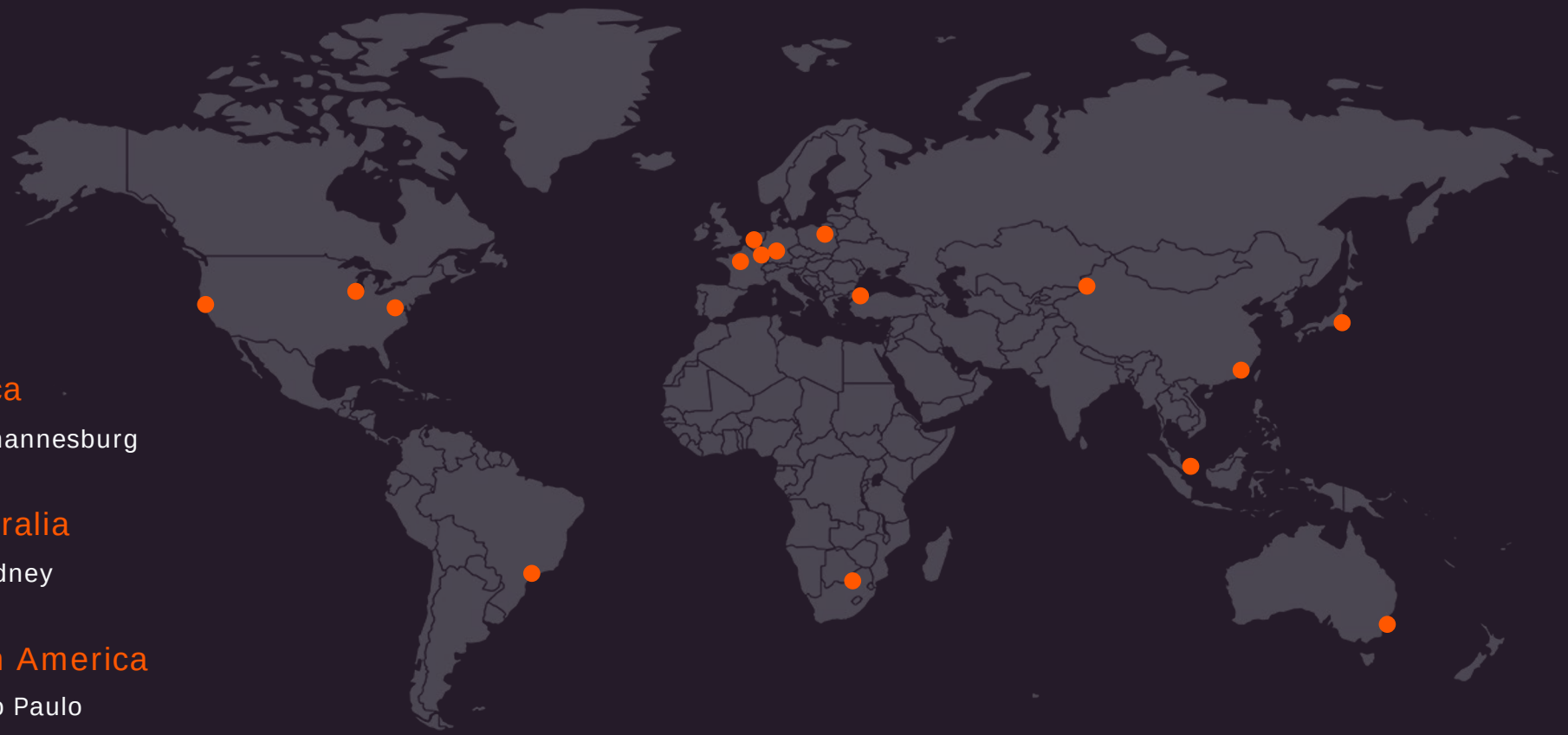
Santa Clara
Chicago
Ashburn

Europe

- Amsterdam
- Frankfurt
- Luxembourg
- Paris
- Istanbul

Asia

- Hong Kong
- Singapore
- Tokyo



Africa

- Johannesburg

Australia

- Sydney

Latin America

- São Paulo

CORE

What services is my team responsible for:

- MKaaS

What services is my team responsible for:

- MKaaS
- FaaS

What services is my team responsible for:

- MKaaS
- FaaS
- LaaS

What services is my team responsible for:

- MKaaS
- FaaS
- LaaS
- *aaS

MKaaS

- Openstack Magnum
- Cluster API

Magnum

- Orchestrates container clusters
- Supports docker swarm and k8s

Magnum

- Orchestrates container clusters
- Supports docker swarm and k8s

- Creates cloud-init config for k8s nodes
- Updates applications verions / configuration on the k8s node

Magnum

- Magnum API

Magnum

- Magnum API
- RabbitMQ

Magnum

- Magnum API
- RabbitMQ
- Magnum conductor

Heat

- Heat API

Heat

- Heat API
- RabbitMQ

Heat

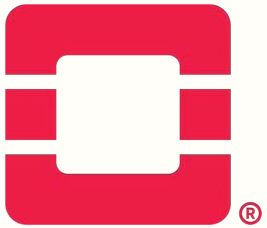
- Heat API
- RabbitMQ
- Heat engine

Heat

- Heat API
- RabbitMQ
- Heat engine
- Heat agent

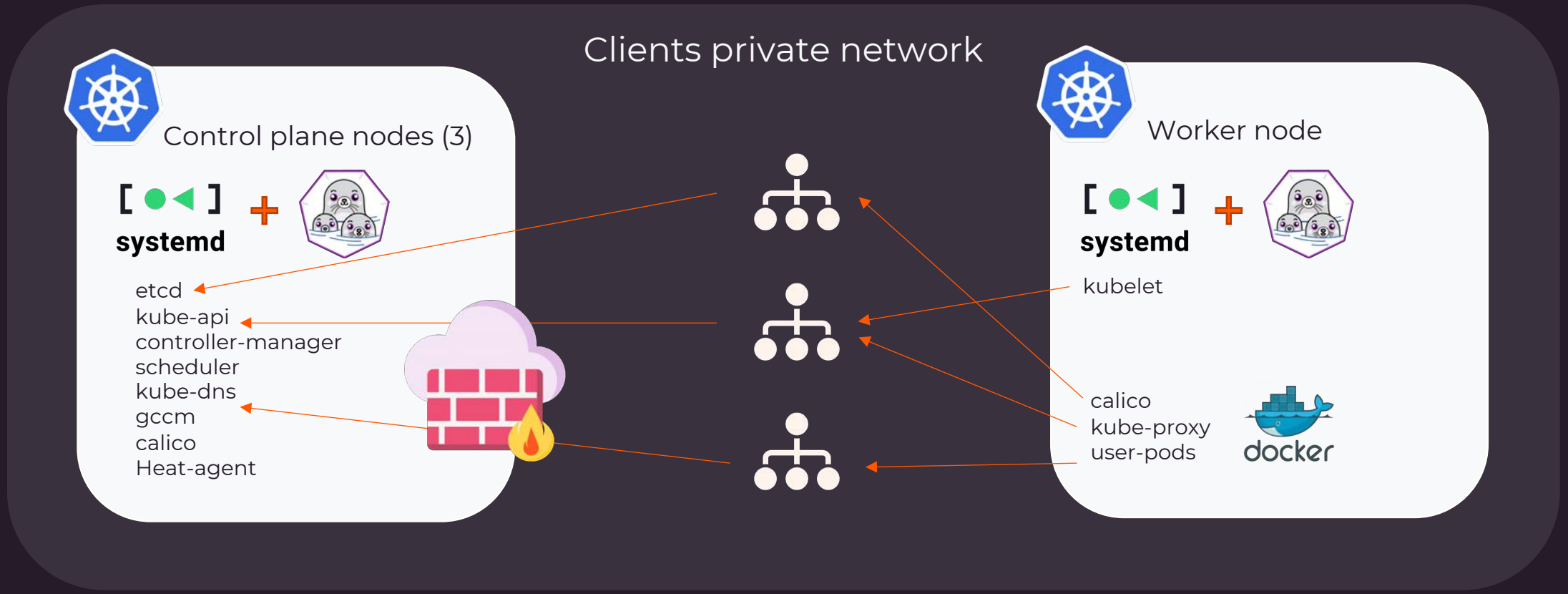
Limitations

No control-plane isolation
from user



It’s an Openstack API
(it needs to be behind GCore Cloud API)

GCore Magnum based managed k8s service architecture:



Magnum + Heat: pros and cons

+

- Extra rps for Cloud API
- Fragile
- Observability
- No baremetal nodes

Magnum + Heat: pros and cons

Magnum

Documentation

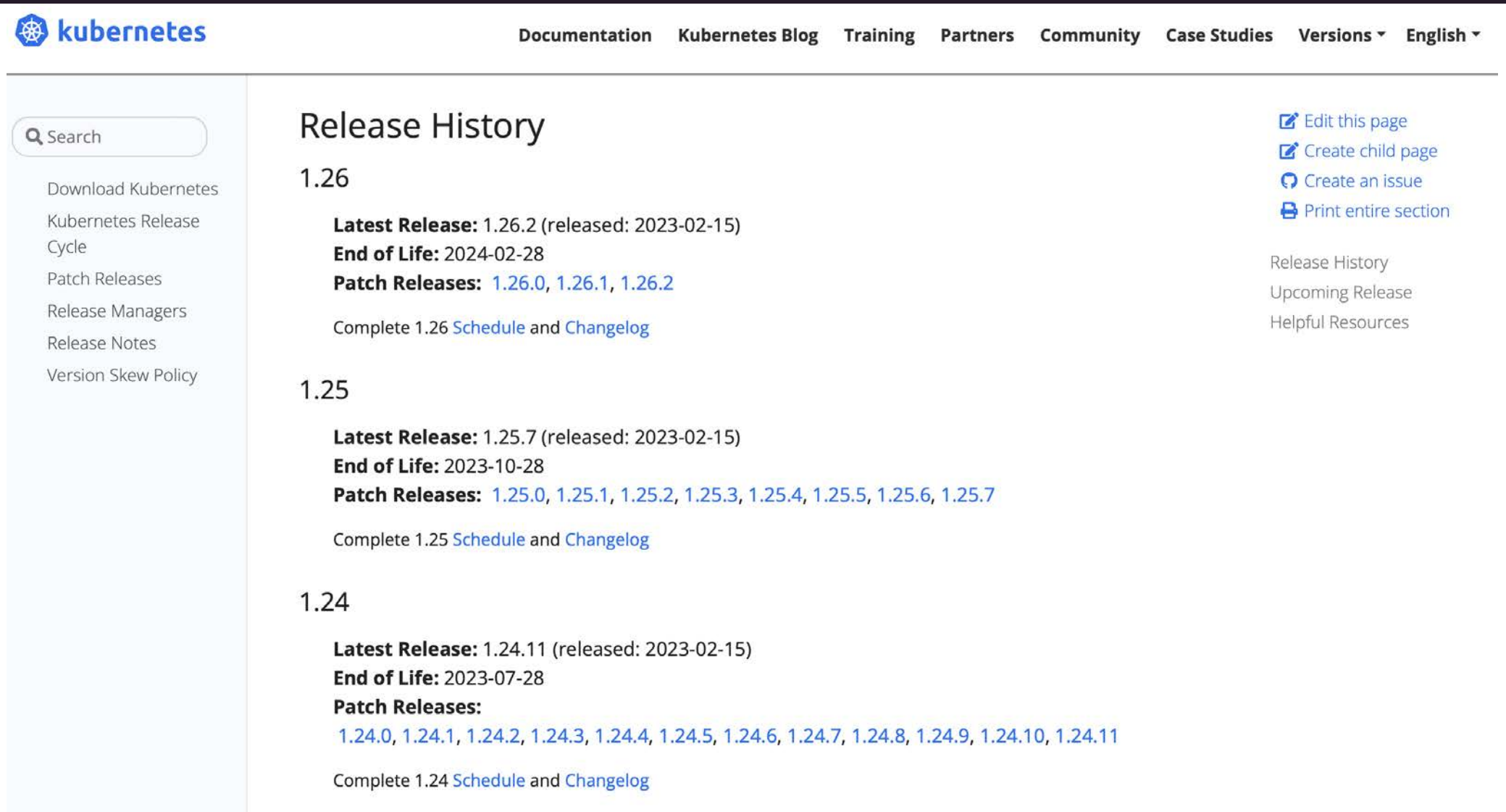
Please see Magnum Documentation

Compatibility Matrix

The following table captures what we know about releases of Kubernetes (kube_tag) that are compatible with different releases of OpenStack Magnum.

Release	kube_tag		os_distro	os version	required labels
	max	default			
14.0.0 (Yoga)	v1.23.3	v1.23.3	fedora-coreos	fcos 35	container_infra_prefix
13.0.0 (Xena)	v1.21.x	v1.18.16	fedora-coreos	fcos 31	container_infra_prefix
12.0.0 (Wallaby)	v1.21.x	v1.18.16	fedora-coreos		container_infra_prefix
11.1.1 (Victoria)	v1.21.x	v1.18.16	fedora-coreos		container_infra_prefix
10.1.0 (Ussuri)	v1.21.x	v1.18.2	fedora-coreos		container_infra_prefix
9.4.1 (Train)	v1.15.x	v1.15.7	fedora-atomic		
9.4.1 (Train)	v1.18.x	v1.15.7	fedora-atomic		use_podman=true

Magnum + Heat: pros and cons



kubernetes

Documentation | Kubernetes Blog | Training | Partners | Community | Case Studies | Versions | English

Search

- Download Kubernetes
- Kubernetes Release Cycle
- Patch Releases
- Release Managers
- Release Notes
- Version Skew Policy

Release History

1.26

Latest Release: 1.26.2 (released: 2023-02-15)
End of Life: 2024-02-28
Patch Releases: 1.26.0, 1.26.1, 1.26.2

Complete 1.26 Schedule and Changelog

1.25

Latest Release: 1.25.7 (released: 2023-02-15)
End of Life: 2023-10-28
Patch Releases: 1.25.0, 1.25.1, 1.25.2, 1.25.3, 1.25.4, 1.25.5, 1.25.6, 1.25.7

Complete 1.25 Schedule and Changelog

1.24

Latest Release: 1.24.11 (released: 2023-02-15)
End of Life: 2023-07-28
Patch Releases:
1.24.0, 1.24.1, 1.24.2, 1.24.3, 1.24.4, 1.24.5, 1.24.6, 1.24.7, 1.24.8, 1.24.9, 1.24.10, 1.24.11

Complete 1.24 Schedule and Changelog

Edit this page
Create child page
Create an issue
Print entire section

Release History
Upcoming Release
Helpful Resources

Cluster api

cluster-api.sigs.k8s.io

- Manages the lifecycle (create, scale, upgrade, destroy) of k8s clusters using a declarative API.
- Work in different environments, both on-premises and in the cloud
- Defines common operations, provide a default implementation, and provide the ability to swap out implementations for alternative ones
- Cluster API can be extended to support any infrastructure (AWS, Azure, vSphere, etc.),
bootstrap or control plane (kubeadm is built-in) provider.

What is cluster api made of?

- capi-controller-manager
- capi-bootstrap-controller-manager
- capi-control-plane-controller-manager
- infrastructure-provider

What do we have out of the box?

Bootstrap

- Kubeadm
- MicroK8s
- Talos
- EKS

Control Plane

- Kubeadm
- MicroK8s
- Talos
- Nested

What do we have out of the box?

Infrastructure:

AWS	**Azure**	**Azure Stack HCI**	**BYOH**	**CloudStack**	**CoxEdge**	**DigitalOcean**
vSphere	**GCP**	**Hetzner**	**IBM Cloud**	**KubeKey**	**KubeVirt**	**MAAS**
Metal3	**Microvm**	**Nested**	**Nutanix**	**OCI**	**OpenStack**	**Outscale**
Sidero	**Tinkerbell**	**vcluster**	**Virtink**	**VMware Cloud Director**		**Equinix Metal (formerly Packet)**

CORE

Some yaml

API: x-k8s.io/v1beta1

Cluster

Control-plane

Machine-deployment

```
spec:
  clusterNetwork:
    pods:
      cidrBlocks:
        - 192.168.0.0/21
    serviceDomain: cluster.local
    services:
      cidrBlocks:
        - 192.168.12.0/22
  controlPlaneEndpoint:
    host: 10.0.0.38
    port: 6443
  controlPlaneRef:
    apiVersion: controlplane.cluster.x-k8s.io/v1beta1
    kind: KubeadmControlPlane
    name: my-cluster-control-plane
    namespace: my-cluster
  infrastructureRef:
    apiVersion: infrastructure.cluster.x-k8s.io/v1beta1
    kind: GcoreCluster
    name: my-cluster
```

CORE

Some yaml

API: x-k8s.io/v1beta1

Cluster

Control-plane

Machine-deployment

```
spec:
  clusterNetwork:
    pods:
      cidrBlocks:
        - 192.168.0.0/21
    serviceDomain: cluster.local
    services:
      cidrBlocks:
        - 192.168.12.0/22
  controlPlaneEndpoint:
    host: 10.0.0.38
    port: 6443
  controlPlaneRef:
    apiVersion: controlplane.cluster.x-k8s.io/v1beta1
    kind: KubeadmControlPlane
    name: my-cluster-control-plane
    namespace: my-cluster
  infrastructureRef:
    apiVersion: infrastructure.cluster.x-k8s.io/v1beta1
    kind: GcoreCluster
    name: my-cluster
```

CORE

Some yaml

API: x-k8s.io/v1beta1

Cluster

Control-plane

Machine-deployment

```
spec:
 kubeadmConfigSpec:
   clusterConfiguration:
    apiServer:
     extraArgs:
       cloud-provider: external
    controllerManager:
     extraArgs:
       cloud-provider: external
  format: cloud-config
   initConfiguration:
    localAPIEndpoint: {}
    nodeRegistration:
     criSocket: /var/run/crio/crio.sock
     kubeletExtraArgs:
       cloud-provider: external
       cluster-dns: 169.254.25.10
```

```
machineTemplate:
   infrastructureRef:
     apiVersion: infrastructure.cluster.x-k8s.io/v1beta1
     kind: GcoreMachineTemplate
     name: my-cluster-control-plane-9f71e857-feb7-45c0-9f76-
3c7b8249e301
     namespace: my-cluster
replicas: 3
  rolloutStrategy:
   rollingUpdate:
     maxSurge: 1
   type: RollingUpdate
  version: v1.24.10
```

CORE

Some yaml

API: x-k8s.io/v1beta1

Cluster

Control-plane

Machine-deployment

```
spec:
 kubeadmConfigSpec:
   clusterConfiguration:
    apiServer:
     extraArgs:
       cloud-provider: external
    controllerManager:
     extraArgs:
       cloud-provider: external
  format: cloud-config
   initConfiguration:
    localAPIEndpoint: {}
    nodeRegistration:
     criSocket: /var/run/crio/crio.sock
     kubeletExtraArgs:
       cloud-provider: external
       cluster-dns: 169.254.25.10
```

```
machineTemplate:
   infrastructureRef:
     apiVersion: infrastructure.cluster.x-k8s.io/v1beta1
     kind: GcoreMachineTemplate
     name: my-cluster-control-plane-9f71e857-feb7-45c0-9f76-3c7b8249e301
     namespace: my-cluster
replicas: 3
  rolloutStrategy:
   rollingUpdate:
     maxSurge: 1
   type: RollingUpdate
  version: v1.24.10
```

CORE

Some yaml

API: x-k8s.io/v1beta1

Cluster

Control-plane

Machine-deployment

```
spec:
  clusterName: my-cluster
  replicas: 6
  strategy:
    rollingUpdate:
      maxSurge: 1
      maxUnavailable: 0
    type: RollingUpdate
  template:
    metadata:
      labels:
        cluster.x-k8s.io/cluster-name:my-cluster
        cluster.x-k8s.io/deployment-name: my-cluster-md-0
```

```
spec:
    bootstrap:
      configRef:
        apiVersion: bootstrap.cluster.x-k8s.io/v1beta1
        kind: KubeadmConfigTemplate
        name: my-cluster-md-0
    clusterName: my-cluster
    infrastructureRef:
      apiVersion: infrastructure.cluster.x-k8s.io/v1beta1
      kind: GcoreMachineTemplate
      name:my-cluster-md-0-68fc39a2-3239-4dbf-bf50-94652f37c260
    nodeDrainTimeout: 1m0s
    version: v1.24.10
```

CORE

Some yaml

API: x-k8s.io/v1beta1

Cluster

Control-plane

Machine-deployment

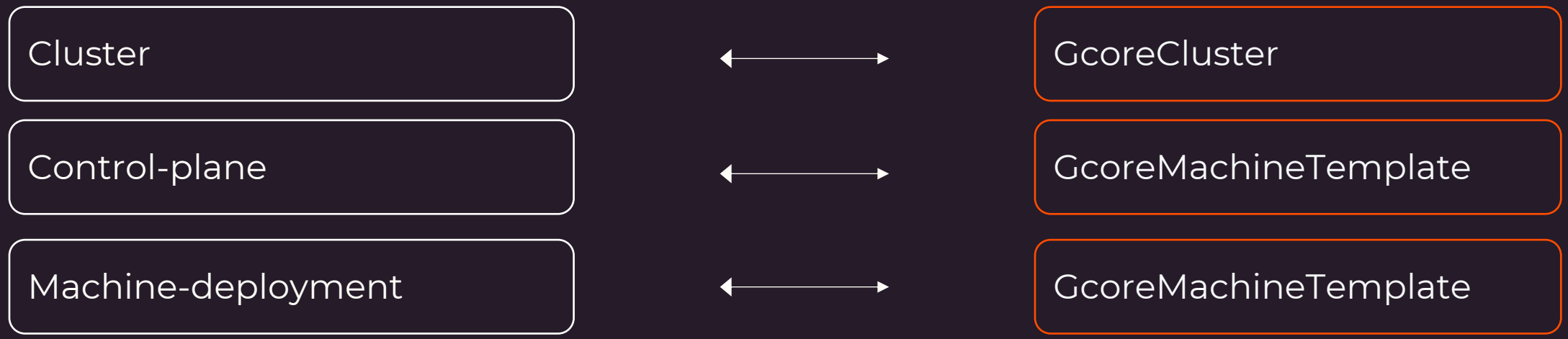
```
spec:
  clusterName: my-cluster
  replicas: 6
  strategy:
    rollingUpdate:
      maxSurge: 1
      maxUnavailable: 0
    type: RollingUpdate
  template:
    metadata:
      labels:
        cluster.x-k8s.io/cluster-name:my-cluster
        cluster.x-k8s.io/deployment-name: my-cluster-md-0
```

```
spec:
  bootstrap:
    configRef:
      apiVersion: bootstrap.cluster.x-k8s.io/v1beta1
      kind: KubeadmConfigTemplate
      name: my-cluster-md-0
  clusterName: my-cluster
  infrastructureRef:
    apiVersion: infrastructure.cluster.x-k8s.io/v1beta1
    kind: GcoreMachineTemplate
    name:my-cluster-md-0-68fc39a2-3239-4dbf-bf50-94652f37c260
  nodeDrainTimeout: 1m0s
  version: v1.24.10
```

CORE

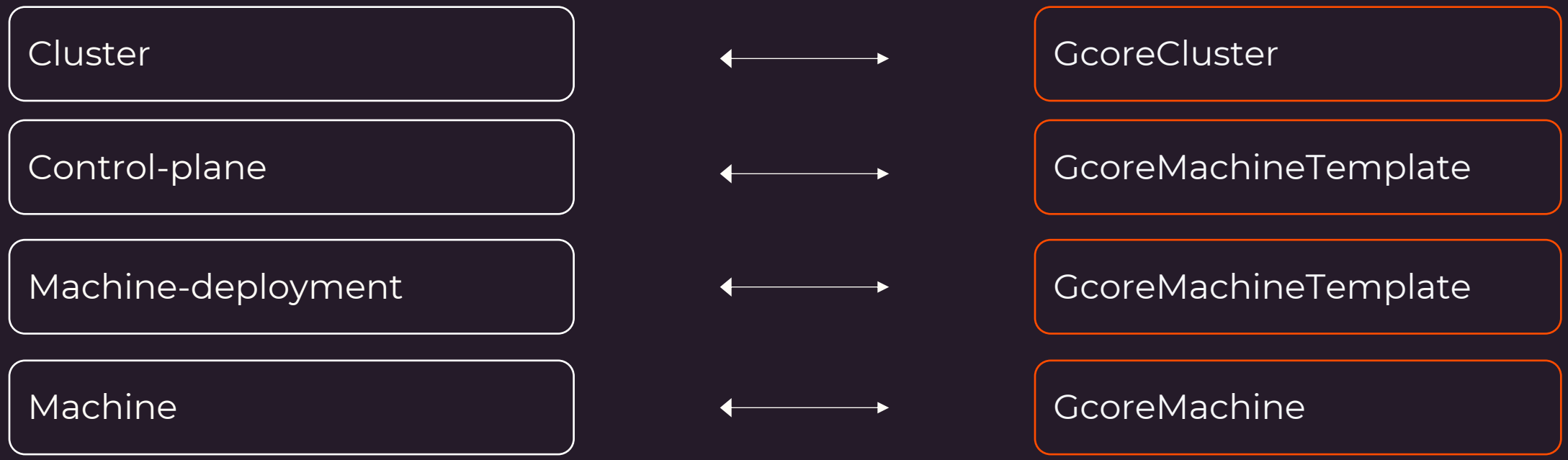
Some yaml

API: x-k8s.io/v1beta1



Some yaml

API: x-k8s.io/v1beta1



Limitations

Bootstrap: **kubeadm** Control plane: **kubeadm** Infrastructure: **none**

- No control-plane isolation from user
- You need a k8s cluster to create a k8s cluster
- No GCore provider
- 3 VMs for control-plane

Our implementation

Bootstrap:

gcore-bootstrap-controller

Control plane:

gcore-control-plane-controller

Infrastructure:

capgc (thanks for our colleges from the Wargaming for help with that ;)

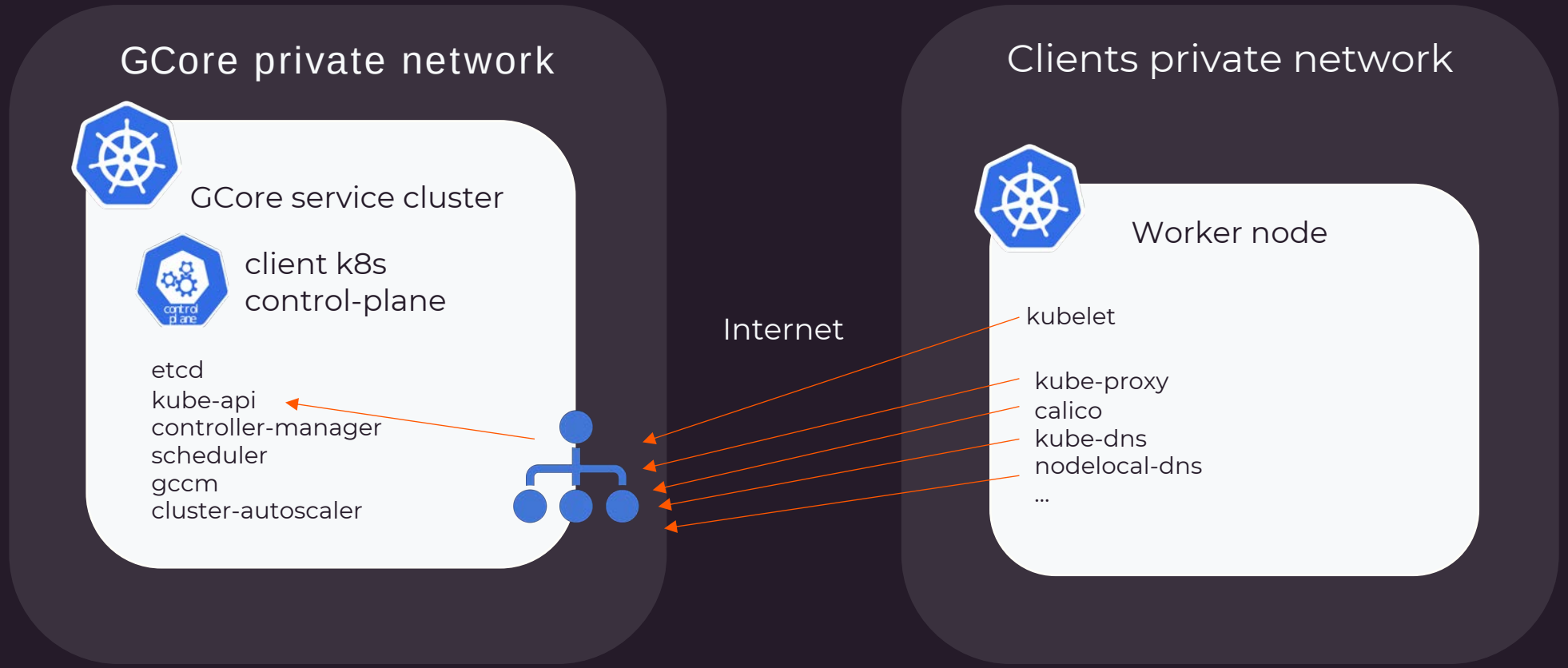
- Control Plane components are pods inside the service k8s cluster
- All CAPI objects are at the same namespace as control plane pods
- no VMs for control-plane

Two more things

- OpenVPN controller

- AgroCD

GCore ClasterAPI based managed k8s service architecture:



Reverse network connectivity

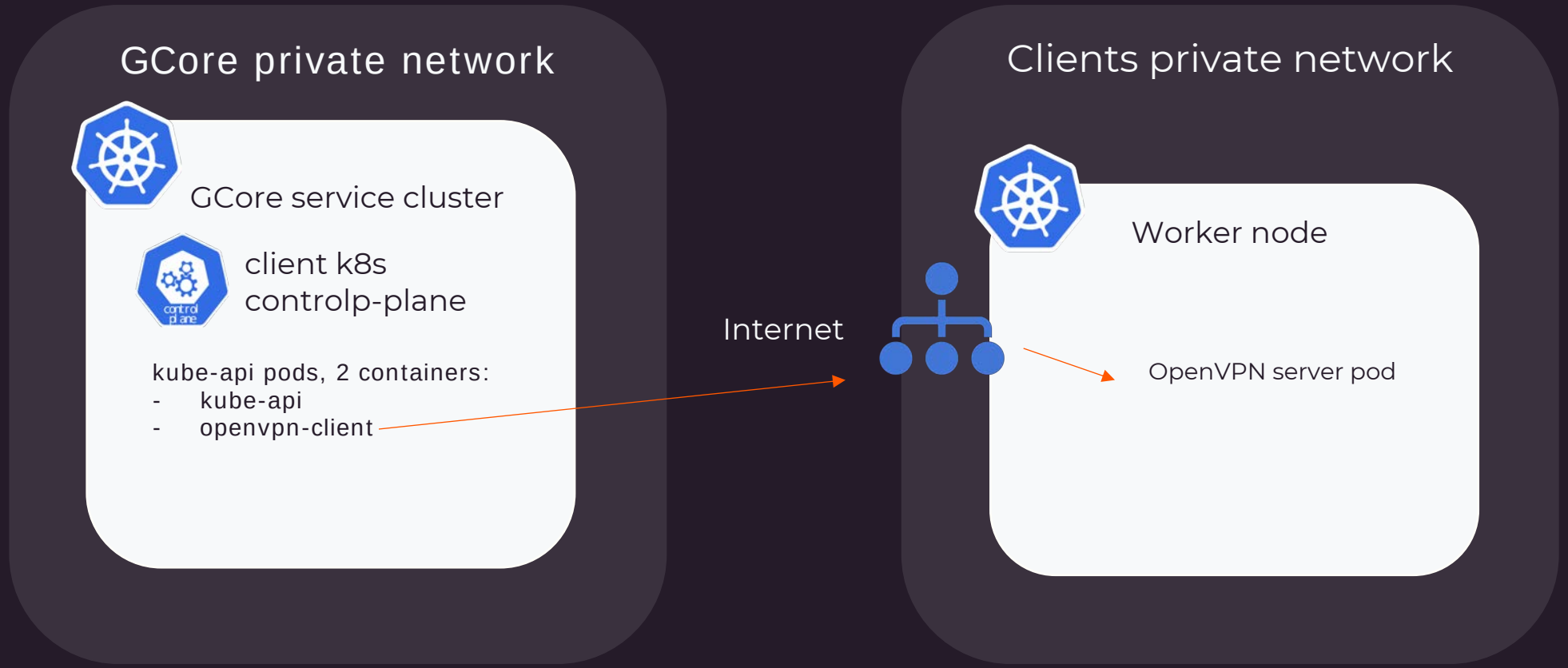
- kubectl logs?
- kubectl port-forward?
- Admission web hooks?

Reverse network connectivity

- kubectl logs?
- kubectl port-forward?
- Admission web hooks?

VPN!

GCore ClasterAPI based managed k8s service architecture:



What if client deletes something?

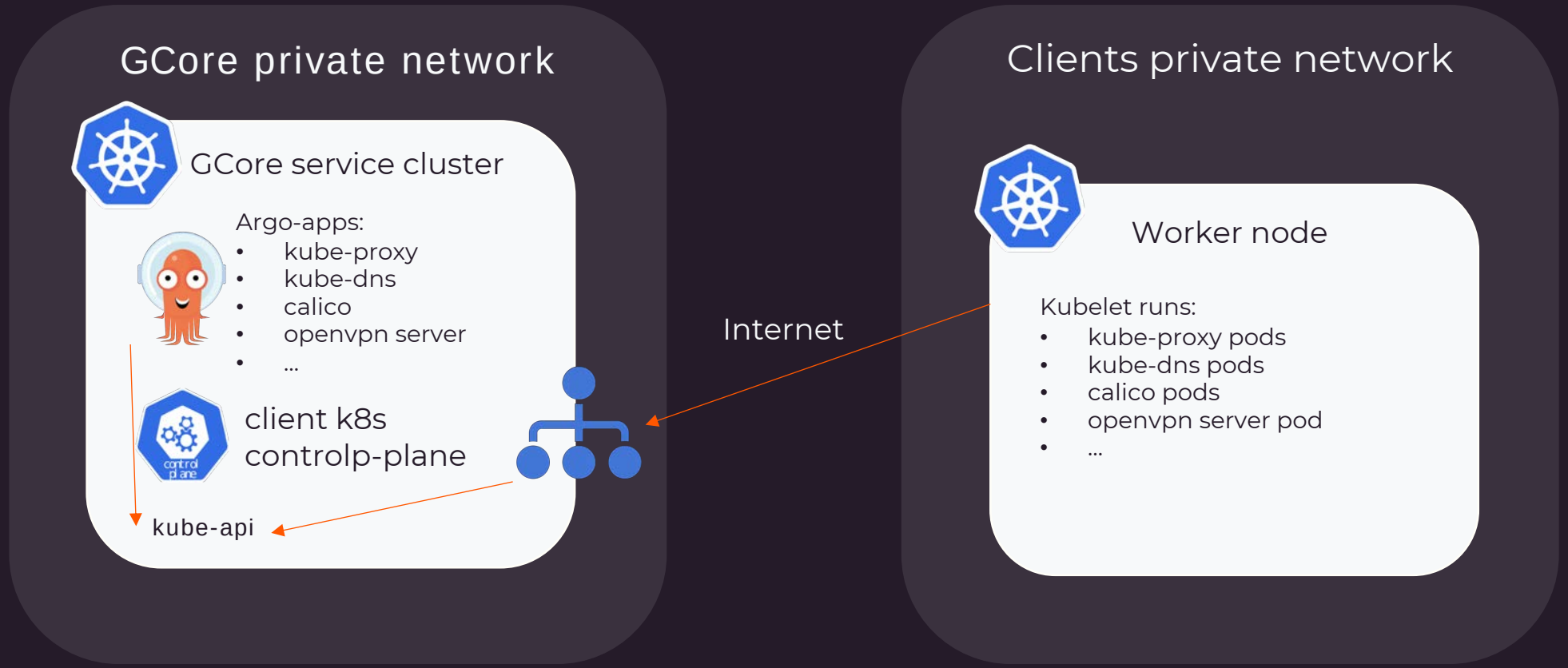
- openvpn server?
- calico?
- kube-proxy?

What if client deletes something?

- openvpn server?
- calico?
- kube-proxy?

ArgoCD!

GCore ClasterAPI based managed k8s service architecture:



Observability

kubectl get cluster –A

NAMESPACE	NAME	CLUSTER	READY	VERSION
76-12706	cluster-1-control-plane	cluster-1	true	v1.24.10
76-15	some-cluster-1-control-plane	some-cluster-1	true	v1.24.10
76-250683	cluster-1-control-plane	cluster-1	true	v1.24.10
76-309102	cluster-tst-2-control-plane	cluster-tst-2	true	v1.24.10

Observability

kubectl get gcorecontrolplane –A

NAMESPACE	NAME	CLUSTER	READY	VERSION
76-12706	cluster-1-control-plane	cluster-1	true	v1.24.10
76-15	some-cluster-1-control-plane	some-cluster-1	true	v1.24.10
76-250683	cluster-1-control-plane	cluster-1	true	v1.24.10
76-309102	cluster-tst-2-control-plane	cluster-tst-2	true	v1.24.10

Observability

kubectl get machinedeployment –A

NAMESPACE	NAME	CLUSTER	REPLICAS	READY	UPDATED	UNAVAILABLE	PHASE	AGE	VERSION
76-12706	cluster-1-pool-1-machine-deployment	cluster-1	3	3	3	0	Running	2d9h	v1.24.10
76-15	some-cluster-1-pool-1-machine-deployment	some-cluster-1	1	1	1	0	Running	8d	v1.24.10
76-250683	cluster-1-pool-1-machine-deployment	cluster-1	1	1	1	0	Running	6d2h	v1.24.10
76-309102	cluster-tst-2-pool-1-machine-deployment	cluster-tst-2	2	2	2	0	Running	8d	v1.24.10

Observability

kubectl get machine –A

NAMESPACE	NAME	CLUSTER	NODENAME	PROVIDERID	PHASE	AGE	VERSION
76-12706	**cluster-1-pool-1-machine-deployment-556c46fb6b-cth78**	**cluster-1**	**cluster-1-pool-1-machine-deployment-556c46fb6b-cth78**	**gcore:///3e198962-8694-4de1-a174-8138811bc1c3**	**Running**	**2d9h**	**v1.24.10**
76-12706	cluster-1-pool-1-machine-deployment-556c46fb6b-g8s95	cluster-1	cluster-1-pool-1-machine-deployment-556c46fb6b-g8s95	gcore:///41ae73a4-9e01-4a87-a8d0-82d1daa6b72d	Running	2d9h	v1.24.10
76-12706	cluster-1-pool-1-machine-deployment-556c46fb6b-xvf8c	cluster-1	cluster-1-pool-1-machine-deployment-556c46fb6b-xvf8c	gcore:///2c36b142-123d-4ec5-9d12-a54ad5a33a55	Running	2d9h	v1.24.10
76-15	some-cluster-1-pool-1-machine-deployment-bbb5fd448-m2sww	some-cluster-1	some-cluster-1-pool-1-machine-deployment-bbb5fd448-m2sww	gcore:///fea0a8ec-a903-433c-993b-5eb33f73ae7c	Running	8d	v1.24.10
76-250683	cluster-1-pool-1-machine-deployment-556c46fb6b-fs2lg	cluster-1	cluster-1-pool-1-machine-deployment-556c46fb6b-fs2lg	gcore:///80eb4694-a62e-434d-8cb5-a146942c137e	Running	6d2h	v1.24.10
76-309102	cluster-tst-2-pool-1-machine-deployment-58d854f865-j9s75	cluster-tst-2	cluster-tst-2-pool-1-machine-deployment-58d854f865-j9s75	gcore:///94174303-b888-4a99-8eec-12df4876e9ed	Running	8d	v1.24.10
76-309102	cluster-tst-2-pool-1-machine-deployment-58d854f865-pjrpx	cluster-tst-2	cluster-tst-2-pool-1-machine-deployment-58d854f865-pjrpx	gcore:///d6ac1244-ae20-44cb-8f48-d52b40b4a268	Running	8d	v1.24.10

CORE

ClusterAPI vs Magnum

- Speed up
- Reconcilation loop
- Transparency
- Easy updates and 1.24, 1.25, 1.26 + (and easy k8s infra updates)
- Baremetal worker nodes
- No control-plane nodes – no extra cost

Powered by intel

Thanks for your attention!

For any questions feel free to contact me:

Andrei Novoselov Edge Cloud system Engineers Team Lead at Gcore

andrei.novoselov@gcore.com
t.me/andrei_novoselov