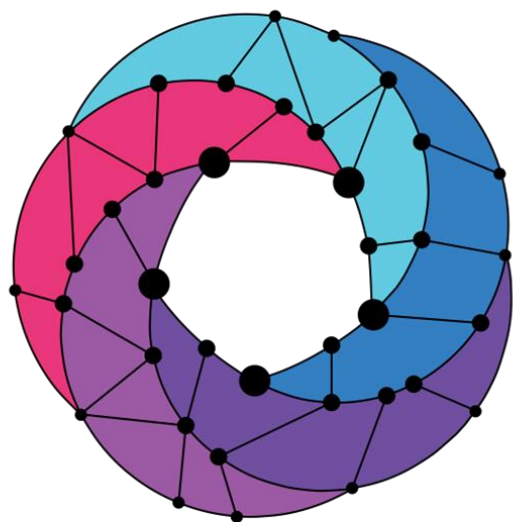




Open Source
MANO

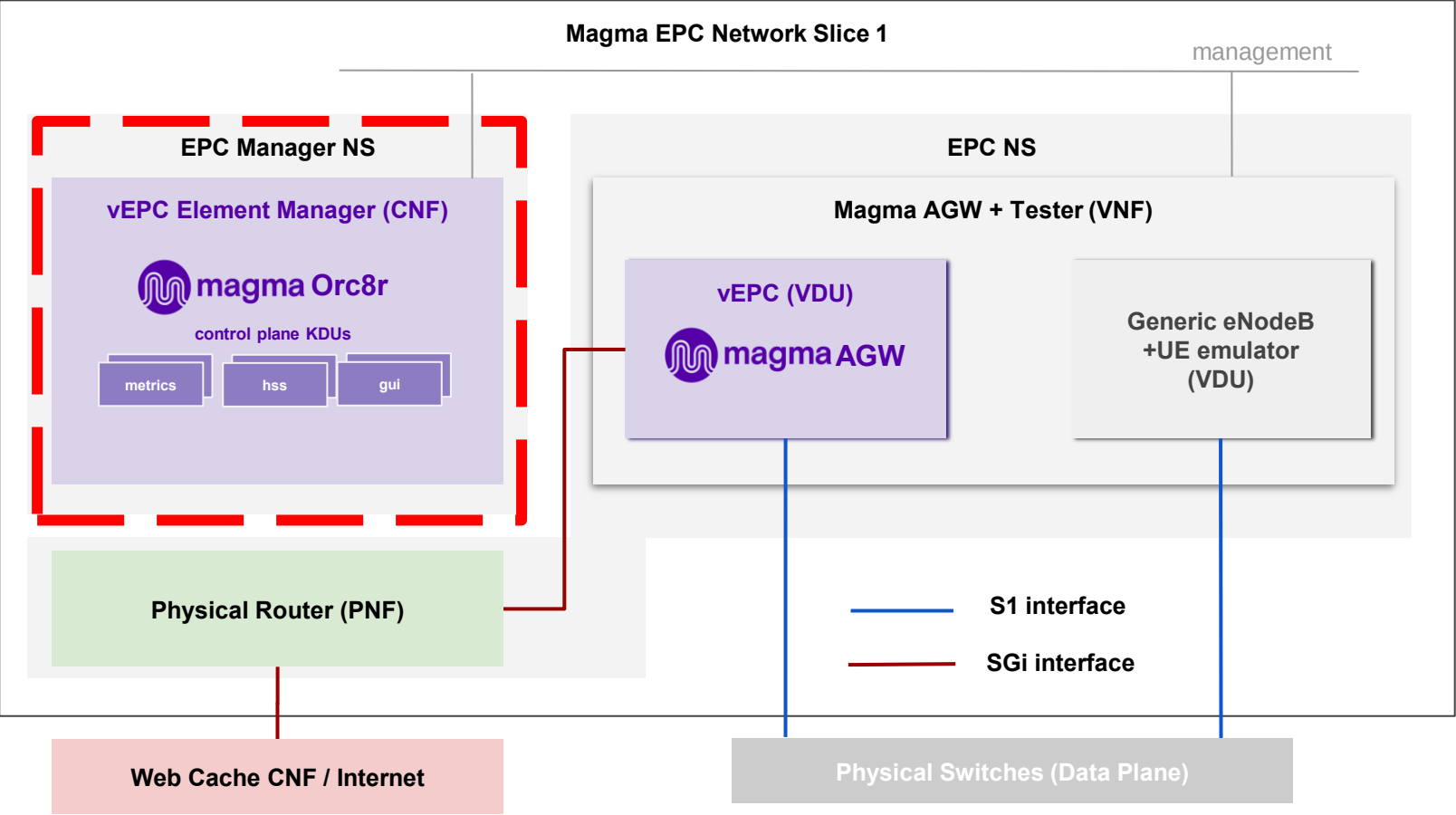
OSM#10 Hackfest – Day 2

Session 6. Orchestrating a CNF with Helm Charts - Magma Orch

Gerardo García (Telefónica)

Context of this session

Magma Orchestrator CNF



- Adding a K8s cluster (association to a VIM)
- Onboarding of NS and NF packages
 - Adding a repo for the VNF Catalog
 - Onboarding NF package
 - Onboarding NS package
- Structure of the packages
 - NS packages(fb_magma_ns)
 - VNF package (fb_magma_knf)

Session topics

The details behind what you did yesterday

- Helm charts
 - Intro
 - Structure of the magma/orc8r helm chart
 - Adding a Helm chart repo
- LCM of the NS
 - Instantiation
 - Operation
 - Termination



Open Source
MANO

Adding a K8s cluster



Adding a K8s cluster

Your Kubernetes cluster needs to meet the following requirements:

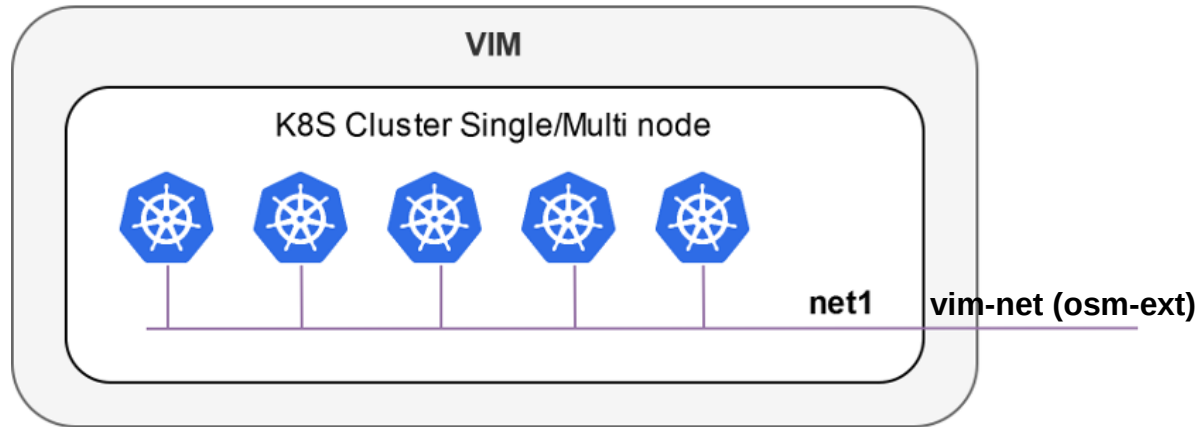
- Kubernetes Load Balancer, to expose your CNFs to the network
- Kubernetes default Storage Class, to support persistent volumes.

A K8s cluster was already created for you!
You just need to add it to OSM.

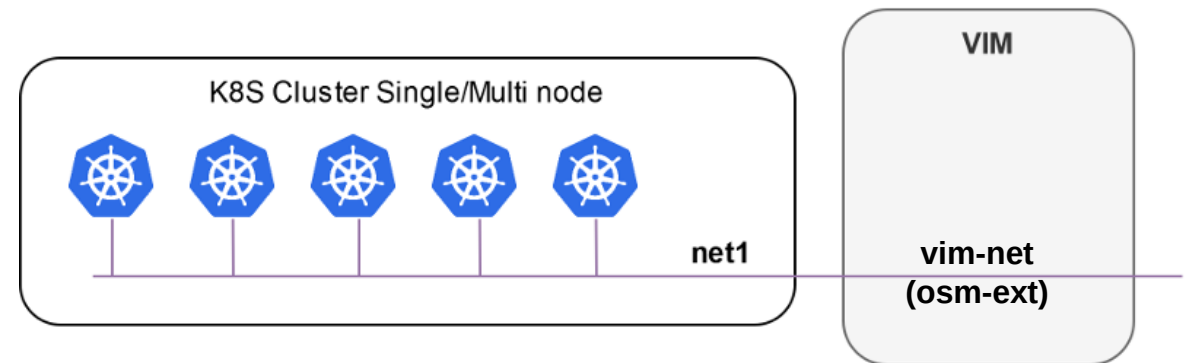
Association of K8s cluster to a VIM

A K8s cluster is expected to be connected

K8s cluster deployed inside a VIM



K8s cluster deployed outside a VIM,
connected to a VIM network



Adding a K8s cluster

- Information to create the cluster:
 - Credentials file: kube.yaml
 - Version: 1.18
 - VIM: osm_hackfest_1
 - K8s nets:
 - net1: osm-ext
 - ClusterName: hackfest

Let's add your K8s cluster

- Add the VIM

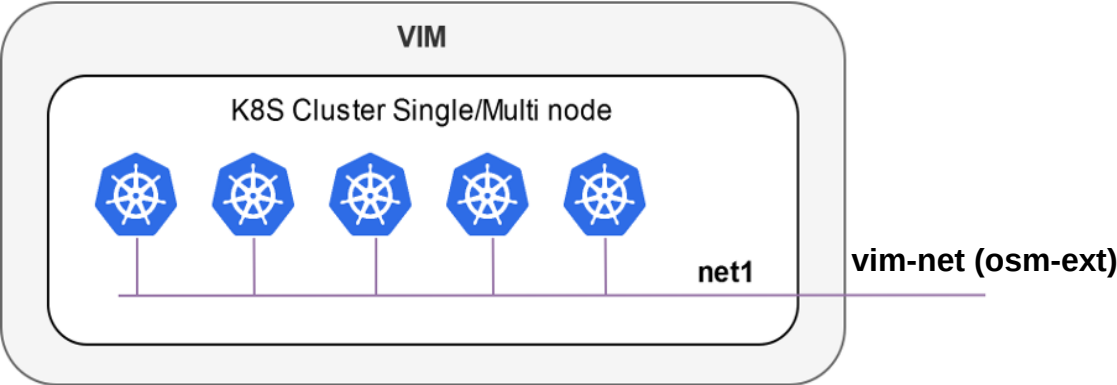
```
osm vim-create --name osm_hackfest_1 --user ${OS_USERNAME} --password ${OS_PASSWORD} \  
               --auth_url ${OS_AUTH_URL} --tenant ${OS_PROJECT_NAME} \  
               --account_type openstack --config '{management_network_name: osm-ext}'  
osm vim-list
```

- Add the cluster

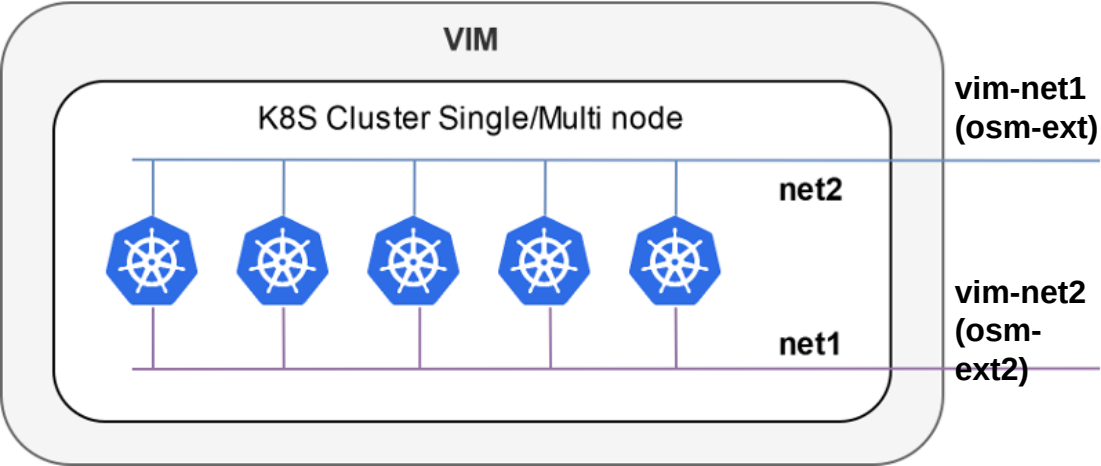
```
osm k8scluster-add --creds ~/kube.yaml \  
                  --version '1.18' \  
                  --vim osm_hackfest_1 \  
                  --description "K8s cluster for user ${HACKFEST_TENANT}" \  
                  --k8s-nets '{"net1": "osm-ext"}' \  
                  hackfest  
osm k8scluster-list
```

Services in the K8s cluster might be exposed in a single network, but maybe in two networks

K8s cluster services exposed in a single VIM network



K8s cluster services exposed in two VIM networks



What if I have an isolated K8s cluster?

- Although these situations are discouraged (an isolated K8s cluster does not make sense in the context of an operator network), it is still possible by creating a dummy VIM target and associating the K8s cluster to that VIM target

```
osm vim-create --name mylocation1 --user u --password p --tenant p --account_type dummy \  
              --auth_url http://localhost/dummy  
  
osm k8scluster-add cluster --creds .kube/config --vim mylocation1 \  
                          --k8s-nets '{k8s_net1: null}' --version "1.18" \  
                          --description="Isolated K8s cluster in mylocation1"
```



Open Source
MANO

Onboarding of NS
and NF packages



Adding a VNF catalog

- Repo name: vnfrepo
- URI: <https://osm.etsi.org/vnf-catalog/Testing/>

```
osm repo-add vnfrepo https://osm.etsi.org/vnf-catalog/Testing/ \  
                --description "Repository for OSM VNF Catalog"  
osm repo-list
```

- More info about OSM repositories here:
<https://osm.etsi.org/docs/user-guide/06-osm-platform-configuration.html#osm-repositories>

Onboarding the VNF and NS packages

- Listing VNF and NS packages in the repo

```
osm nfpkg-repo-list  
osm nspkg-repo-list
```

- Getting details of specific VNF and NS packages in the repo

```
osm nfpkg-repo-show --repo vnfrepo fb_magma_knf  
osm nspkg-repo-show --repo vnfrepo fb_magma_ns
```

- Onboarding the VNF and NS packages (first the VNF, then NS)

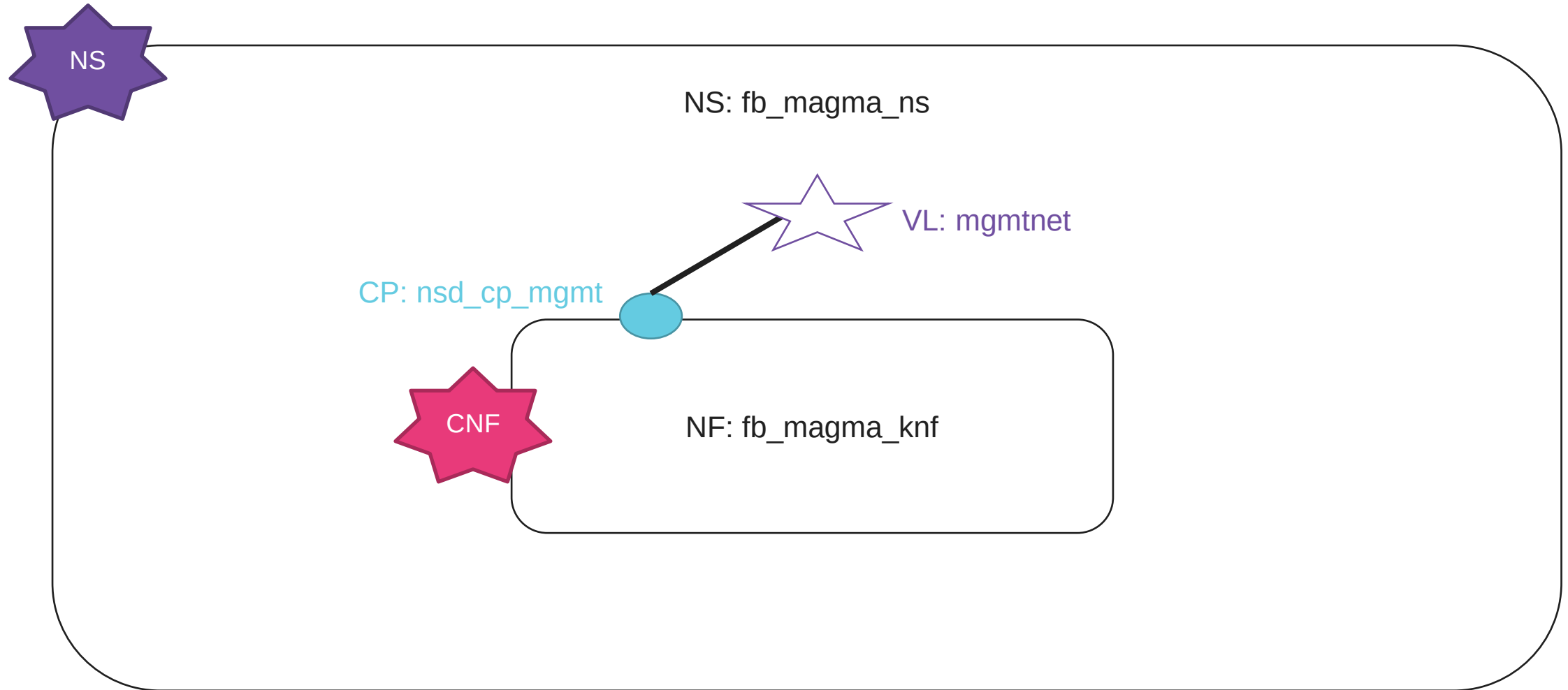
```
osm nfpkg-create --repo vnfrepo fb_magma_knf  
osm nspkg-create --repo vnfrepo fb_magma_ns
```




Open Source
MANO

Structure of the packages





NS descriptor

```
$ osm nspkg-repo-show --repo vnfrepo fb_magma_ns --literal
```

```
nsd-catalog:
```

```
nsd:
```

```
- connection-point:
```

```
- name: nsd_cp_mgmt  
  vld-id-ref: mgmtnet
```

Connection to other NSS in the Network Slice

```
constituent-vnfd:
```

```
- member-vnf-index: orc8r  
  vnfd-id-ref: fb_magma_knf
```

One CNF: fb_magma_knf

```
description: NS consisting of a CNF fb_magma_knf connected to mgmt network
```

```
id: fb_magma_ns
```

```
name: fb_magma_ns
```

```
short-name: fb magma ns
```

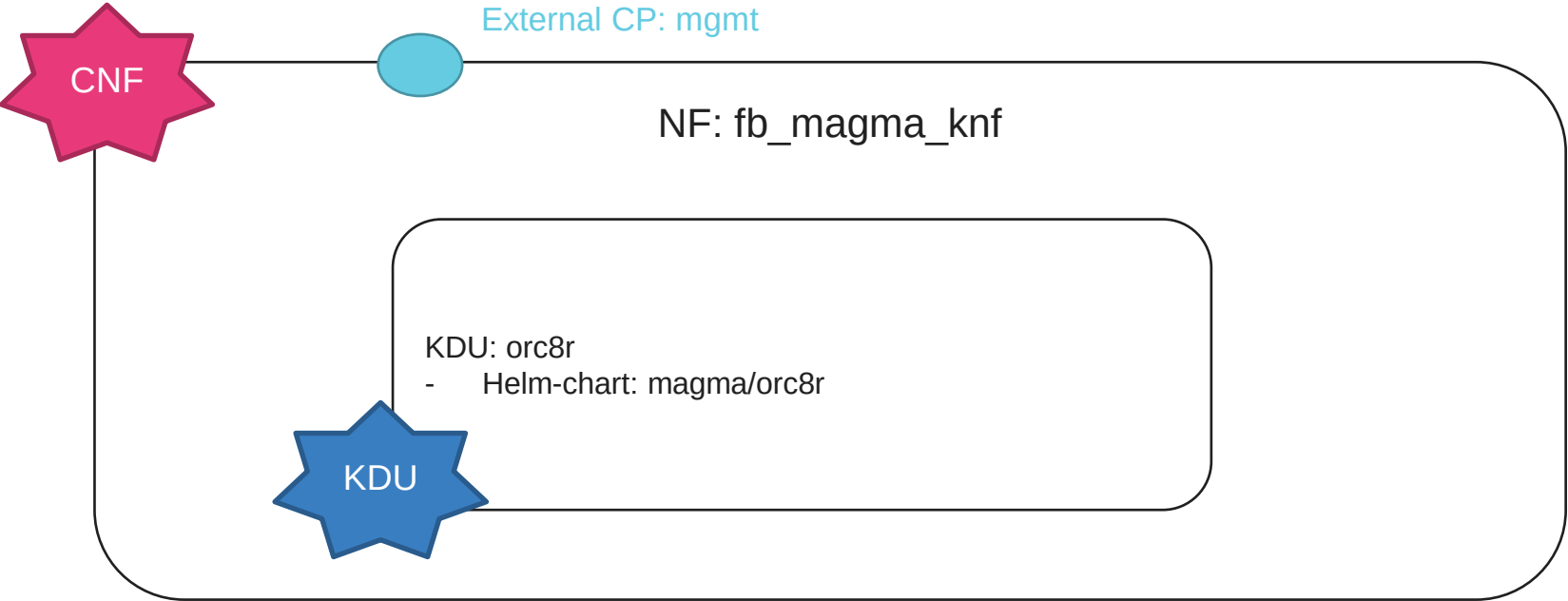
```
vld:
```

```
- id: mgmtnet  
  mgmt-network: true  
  name: mgmtnet  
  type: ELAN  
  vnfd-connection-point-ref:  
  - member-vnf-index-ref: orc8r  
    vnfd-connection-point-ref: mgmt  
    vnfd-id-ref: fb_magma_knf
```

One VL where the CNF will expose its services

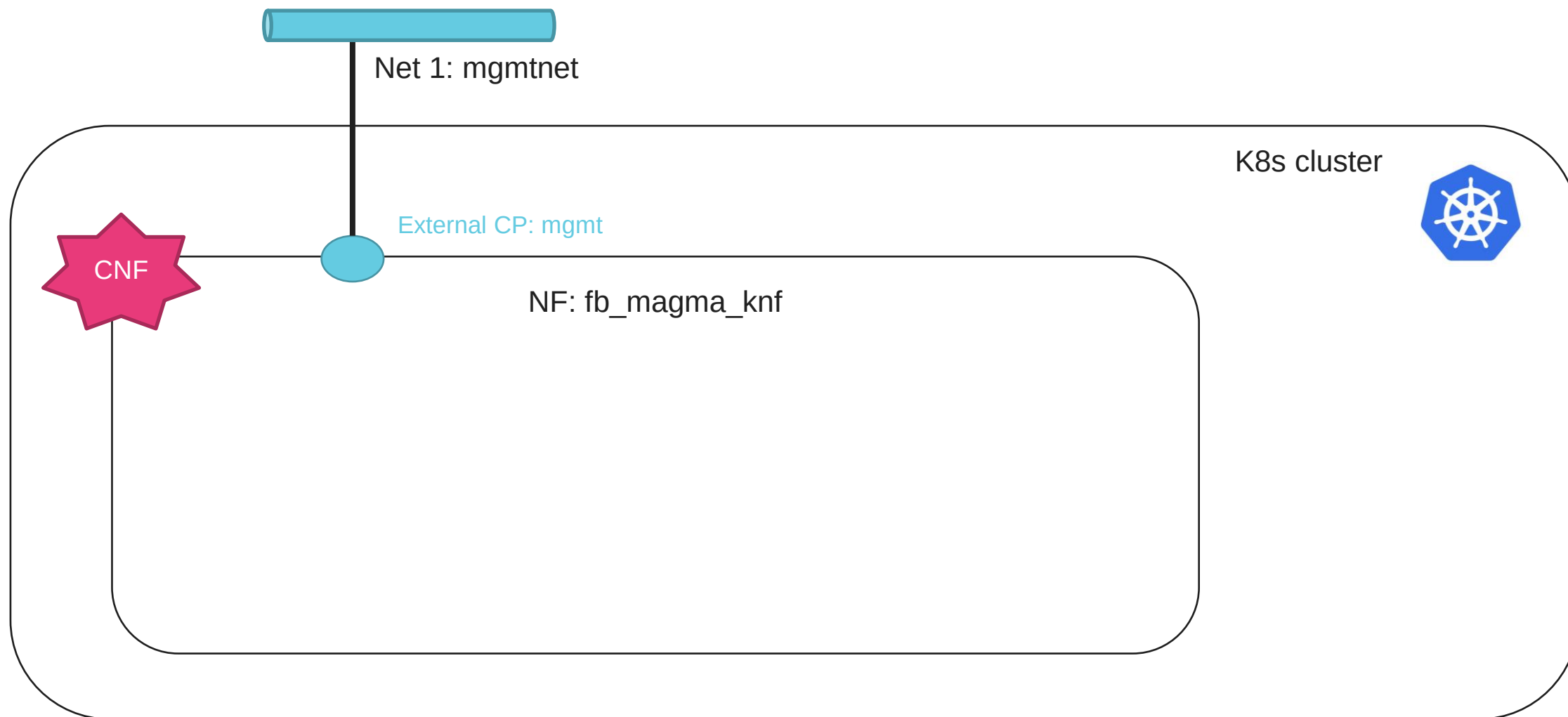
CNF structure

The CNF consists of a single KDU (a helm chart)



CNF structure

K8s cluster requirements: services exposed by single network



VNF descriptor

```
$ osm nfpkg-repo-show --repo vnfrepo fb_magma_knf --literal
```

```
vnfd-catalog:
```

```
  schema-version: '3.0'
```

```
  vnfd:
```

```
    - connection-point:
```

```
      - name: mgmt
```

```
      description: CNF with KDU using a helm-chart for Facebook magma orc8r
```

```
      id: fb_magma_knf
```

```
      k8s-cluster:
```

```
        nets:
```

```
          - external-connection-point-ref: mgmt
```

```
            id: mgmtnet
```

```
      kdu:
```

```
        - helm-chart: magma/orc8r
```

```
          name: orc8r
```

```
      mgmt-interface:
```

```
        cp: mgmt
```

```
      name: fb_magma_knf
```

```
      short-name: fb_magma_knf
```

K8s cluster requirements: 1 network to expose services

One KDU based on helm chart magma/orc8r (repo:magma)



Open Source
MANO

Helm charts



- Concepts:
 - Helm-chart: a packaged format for deploying K8s applications
 - Helm: a package manager
- Documentation:
 - Helm: <https://helm.sh/>
 - Charts: <https://helm.sh/docs/topics/charts/>
- Well known repos bringing access to more than 20,000 K8s applications:
 - Stable: <https://charts.helm.sh/stable>
 - Incubator: <https://charts.helm.sh/incubator>
 - Bitnami: <https://charts.bitnami.com/bitnami>

Helm chart structure

Chart File Structure

```
wordpress/  
  Chart.yaml # A YAML file containing information about the chart  
  LICENSE    # OPTIONAL: A plain text file containing the license for the chart  
  README.md  # OPTIONAL: A human-readable README file  
  values.yaml # The default configuration values for this chart  
  values.schema.json # OPTIONAL: A JSON Schema for imposing a structure on the values.yaml file  
  charts/    # A directory containing any charts upon which this chart depends.  
  crds/      # Custom Resource Definitions  
  templates/ # A directory of templates that, when combined with values,  
              # will generate valid Kubernetes manifest files.  
  templates/NOTES.txt # OPTIONAL: A plain text file containing short usage notes
```

Helm chart structure

Template files (K8s manifest files + Go)

```
apiVersion: v1
kind: ReplicationController
metadata:
  name: deis-database
  namespace: deis
  labels:
    app.kubernetes.io/managed-by: deis
spec:
  replicas: 1
  selector:
    app.kubernetes.io/name: deis-database
  template:
    metadata:
      labels:
        app.kubernetes.io/name: deis-database
    spec:
      serviceAccount: deis-database
      containers:
        - name: deis-database
          image: {{ .Values.imageRegistry }}/postgres:{{ .Values.dockerTag }}
          imagePullPolicy: {{ .Values.pullPolicy }}
          ports:
            - containerPort: 5432
          env:
            - name: DATABASE_STORAGE
              value: {{ default "minio" .Values.storage }}
```

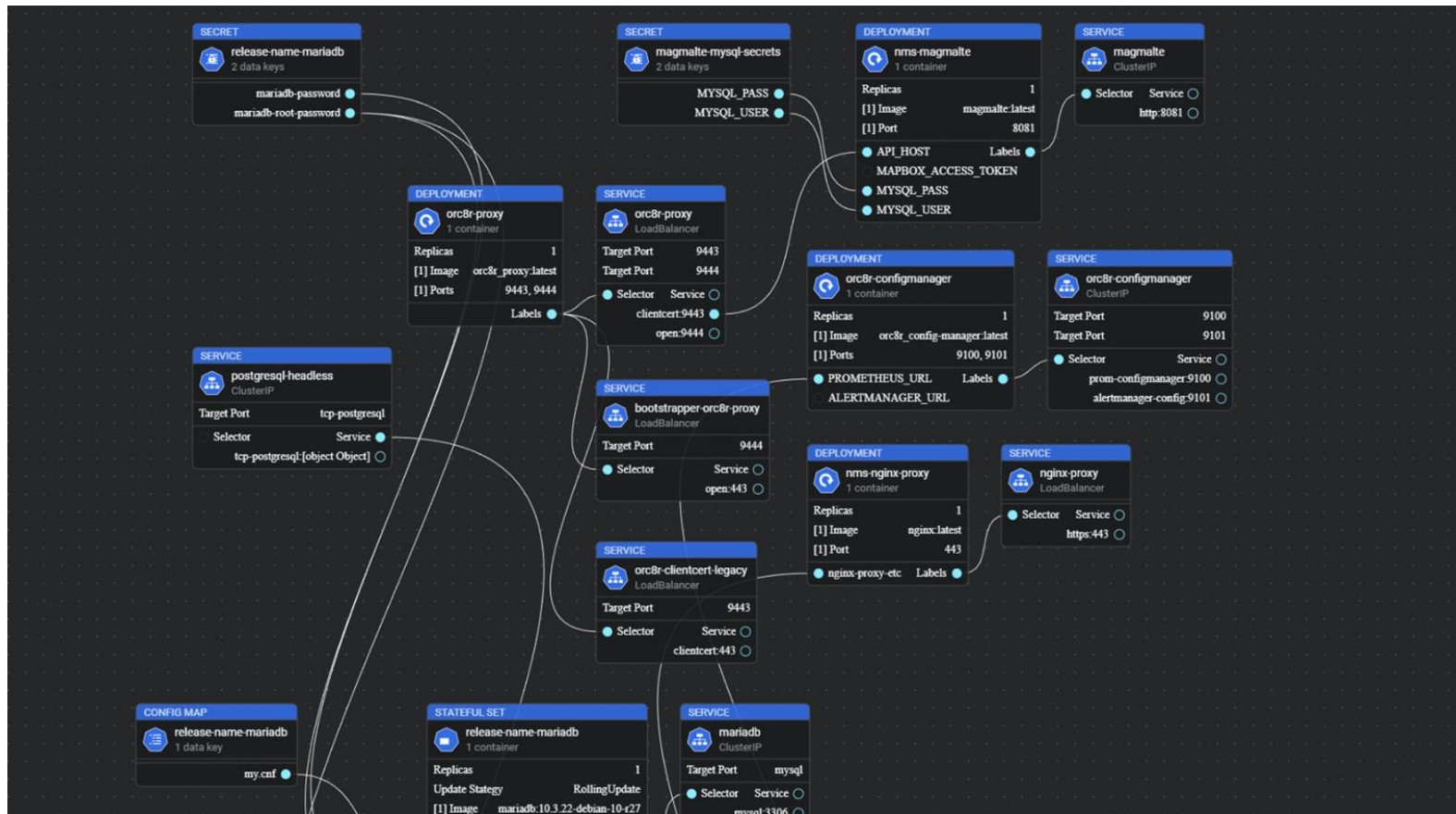
Helm chart structure

Values.yaml

```
imageRegistry: "quay.io/deis"  
dockerTag: "latest"  
pullPolicy: "Always"  
storage: "s3"
```

The Magma Orchestrator helm chart

- Magma Orchestrator helm chart structure



Do you remember our VNF descriptor?

```
$ osm nfpkg-repo-show --repo vnfrepo fb_magma_knf --literal
vnfd-catalog:
  schema-version: '3.0'
  vnfd:
    - connection-point:
        - name: mgmt
      description: CNF with KDU using a helm-chart for Facebook magma orc8r
      id: fb_magma_knf
      k8s-cluster:
        nets:
          - external-connection-point-ref: mgmt
            id: mgmtnet
      kdu:
        - helm-chart: magma/orc8r
          name: orc8r
      mgmt-interface:
        cp: mgmt
        name: fb_magma_knf
        short-name: fb_magma_knf
```

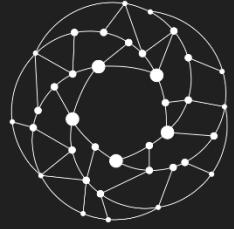
One KDU based on helm chart magma/orc8r (repo:magma)

Adding a helm-chart repo

- OSM needs to know where to obtain the 'orc8r' helm chart from:
 - Repo name: magma
 - URI: <http://osm-download.etsi.org/ftp/Packages/vnf-onboarding-tf/helm/>

```
osm repo-add --type helm-chart \  
             --description "Repository for Facebook Magma helm Chart" \  
             magma \  
             http://osm-download.etsi.org/ftp/Packages/vnf-onboarding-tf/helm/
```

- When the CNF is deployed, its helm chart will be downloaded from the repo



Open Source
MANO

Life Cycle Management of the NS



Instantiating the Network Slice

What you did yesterday

- Launch the Network Slice Instance

```
osm nsi-create --nsi_name magma_slice_${OSM_USER} \  
              --nst_name magma_slice_hackfest_nst \  
              --vim_account osm_hackfest_1 \  
              --config_file params.yaml
```

- Instantiation parameters (params.yaml)

```
netslice-subnet:  
- id: slice_hackfest_nsd_epc  
  additionalParamsForVnf:  
  - member-vnf-index: 'MagmaAGWsrsLTE'  
    additionalParams:  
      agw_id: 'agw_100'  
      agw_name: 'AGW100'  
      orch_ip: '172.21.251.XX'  
      orch_net: 'osmnet'  
  
- id: slice_hackfest_nsd_epcmgmt  
  additionalParamsForVnf:  
  - member-vnf-index: 'orc8r'  
    additionalParamsForKdu:  
      kdu_name: orc8r  
      additionalParams:  
        proxyloadBalancerIP: '172.21.251.XX'
```

Params related to the CNF

Instantiating the NS alone

- Launch the NS

```
osm ns-create --ns_name magma_orc8r \  
              --nsd_name fb_magma_ns \  
              --vim_account osm_hackfest_1 \  
              --config_file params.yaml
```

- Instantiation parameters (params.yaml)

```
additionalParamsForVnf:  
- member-vnf-index: 'orc8r'  
  additionalParamsForKdu:  
  - kdu_name: orc8r  
    additionalParams:  
      proxyserviceBalancerIP: '172.21.251.X' # orc8r-proxy IP Address
```

Checking status of the KDU instance

- Check status of NS in OSM

```
osm ns-list
```

- Once ready, get the details of the KDU

```
osm vnf-list --ns magma_orc8r  
VNF_ID=`osm vnf-list --ns magma_orc8r |grep orc8r |awk '{print $2}'`  
osm vnf-show $VNF_ID --kdu orc8r
```

- You can also check the status of the deployment in K8s (one namespace per OSM project):

```
osm project-list  
kubectl -n <PROJECT_ID> --kubeconfig ~/kube.yaml get all  
kubectl -n <PROJECT_ID> --kubeconfig ~/kube.yaml get svc/orc8r-proxy
```

Running implicit primitives in KDUs

(ONLY AN EXAMPLE, FOR ILLUSTRATIVE PURPOSES)

- You could upgrade your KDU to use a different IP address (172.21.251.X)

```
osm ns-action magma_orc8r --vnf_name orc8r --kdu_name orc8r \  
    --action_name upgrade \  
    --params '{"proxyserviceloadBalancerIP": "172.21.251.X"}'  
osm ns-op-list magma_orc8r
```

- Once completed, you can check that the IP address was updated

```
osm vnf-list --ns magma_orc8r  
VNF_ID=`osm vnf-list --ns magma_orc8r |grep orc8r |awk '{print $2}'`  
osm vnf-show $VNF_ID --kdu orc8r  
osm vnf-show $VNF_ID --kdu orc8r|grep ^orc8r-proxy|grep LoadBalancer
```

- Delete the NS

```
osm ns-delete magma_orc8r  
osm ns-list
```


More information

- [User Guide - Using Kubernetes-based VNFs](#)
- [VNF Onboarding Guidelines - KNF walkthrough](#)



Search docs

TABLE OF CONTENTS

1. OSM Quickstart
2. OSM Architecture and Functions
3. Installing OSM
4. Setup of Virtual Infrastructure Managers (VIMs)
5. OSM Usage
 - 5.1. Deploying your first Network Service
 - 5.2. Advanced instantiation: using instantiation parameters
 - 5.3. Understanding Day-1 and Day-2 Operations
 - 5.4. Monitoring and autoscaling
 - 5.5. Using Network Slices
 - 5.6. Using Kubernetes-based VNFs (KNFs)
 - 5.6.1. Kubernetes installation
 - 5.6.2. OSM Kubernetes requirements
 - 5.6.3. Adding Kubernetes cluster

5.6. Using Kubernetes-based VNFs (KNFs)

From Release SEVEN, OSM supports Kubernetes-based Network Functions (KNF). This feature unlocks more than 20.000 packages that can be deployed besides VNFs and PNFs. This section guides you to deploy your first KNF, from the installation of multiple ways of Kubernetes clusters until the selection of the package and deployment.

5.6.1. Kubernetes installation

KNFs feature requires an operative Kubernetes cluster. There are several ways to have that Kubernetes running. From the OSM perspective, the cluster is not an isolated element, but it is a technology that enables the deployment of microservices in a cloud-native way. To facilitate the connection to the infrastructure, the cluster has to be associated to a VIM. There is a special case where the cluster is deployed in a baremetal environment without the management of the networking part but in general, OSM considers that the Kubernetes cluster is managed by a VIM.

For OSM you can use one of these three different ways to install your Kubernetes cluster:

1. OSM Kubernetes cluster Network Service
2. Self-managed Kubernetes cluster in a VIM
3. Kubernetes baremetal installation

5.6.2. OSM Kubernetes requirements

After the Kubernetes installation is completed, you need to check if you have the following components in your cluster.

1. **Kubernetes Loadbalancer**: to expose your KNFs to the network
2. **Kubernetes default Storageclass**: to support persistent volumes.

5.6.3. Adding Kubernetes cluster to OSM

In order to test Kubernetes-based VNF (KNF), you require a K8s cluster connected to a network in the VIM (e.g. `vim-net`). If you have already installed Kubernetes, you will need to add a VIM in order to add the Kubernetes cluster.



Search docs

TABLE OF CONTENTS

- Introduction
- Onboarding Requirements
- Day 0: VNF Instantiation & Management setup
- Day 1: VNF Services Initialization
- Day 2: VNF Runtime Operations
- Reference NSD/VNFD & Charms
- VNF Onboarding Walkthrough
- KNF Onboarding Walkthrough (Work in Progress)
 - Introduction
 - KNF Requirements
 - Building the KNF Descriptor
 - Launching the KNF Package
 - Testing the KNF Package
 - Testing functionality
- Advanced Charm Development
- Additional References

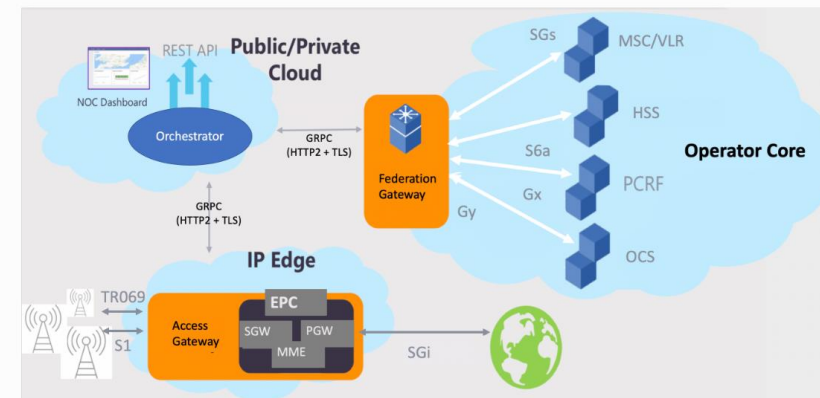
Docs » KNF Onboarding Walkthrough (Work in Progress)

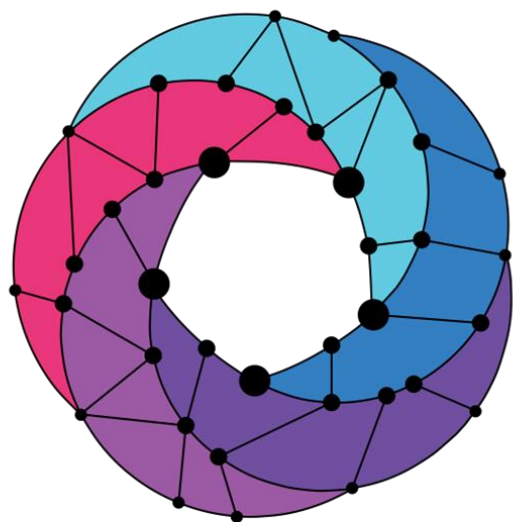
[View page source](#)

KNF Onboarding Walkthrough (Work in Progress)

Introduction

This section uses Facebook's Magma, an open-source software platform that gives network operators an open, flexible and extendable mobile core network solution.





Open Source MANO

Find us at:

osm.etsi.org
osm.etsi.org/docs
osm.etsi.org/wikipub

