

Open Source
MANO

OSM-MR#9 Hackfest
Orchestrating a PNF in OSM
Gianpietro Lavado (Whitestack)



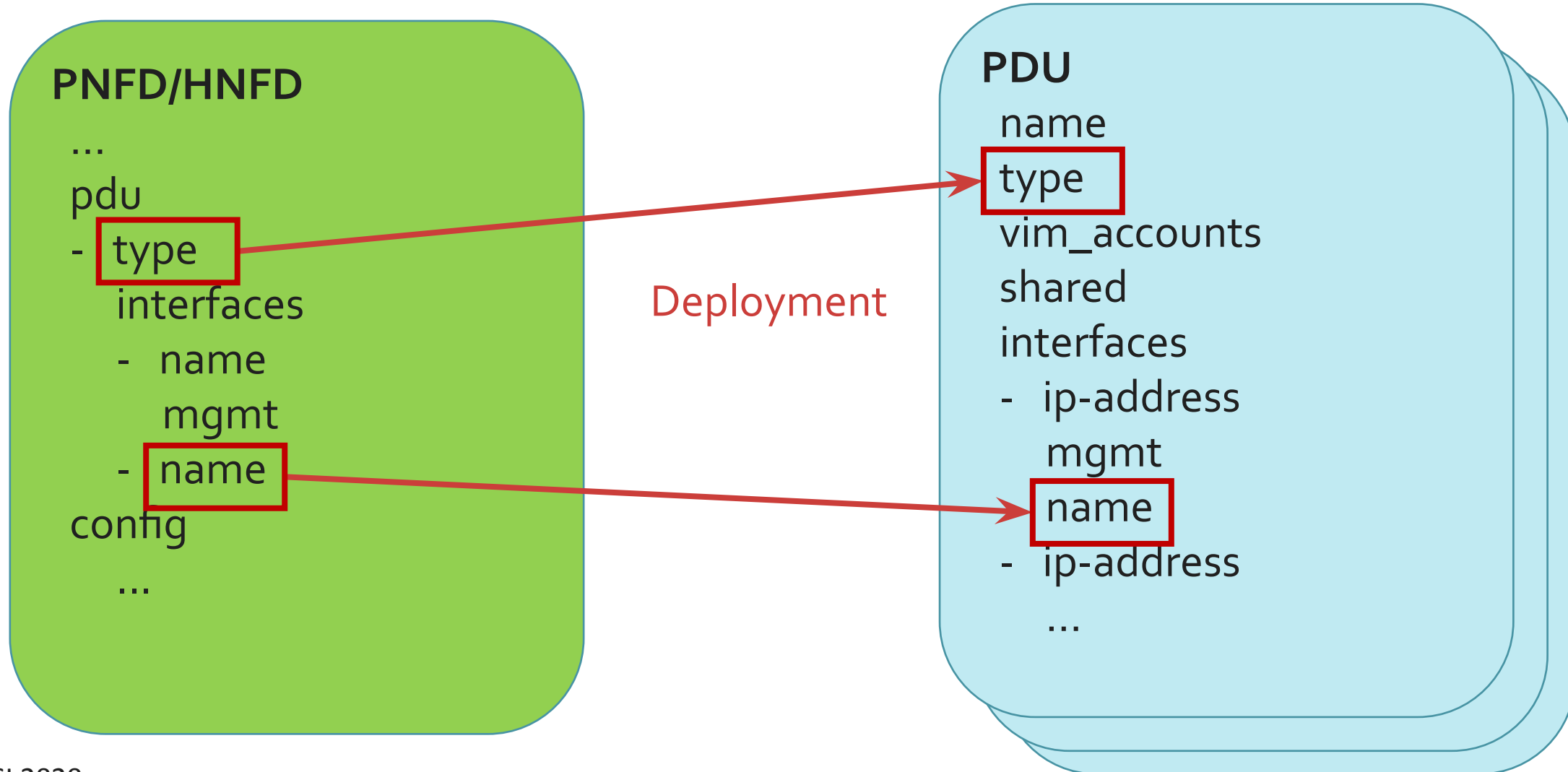
PNF / HNF - Definitions

- **PNF:** Physical network function. It refers to a HW box that provides a networking function. Routers, firewalls, load balancers, etc.
- **PDU:** Physical deployment unit. It refers to the “instance” of the physical appliance that will be incorporated to a Network Service instance.
- **HNF:** Hybrid network function: Network function composed of both physical and virtual elements.

In OSM, there are no fundamental differences between a VNF, a PNF or a Hybrid Network Function (HNF)

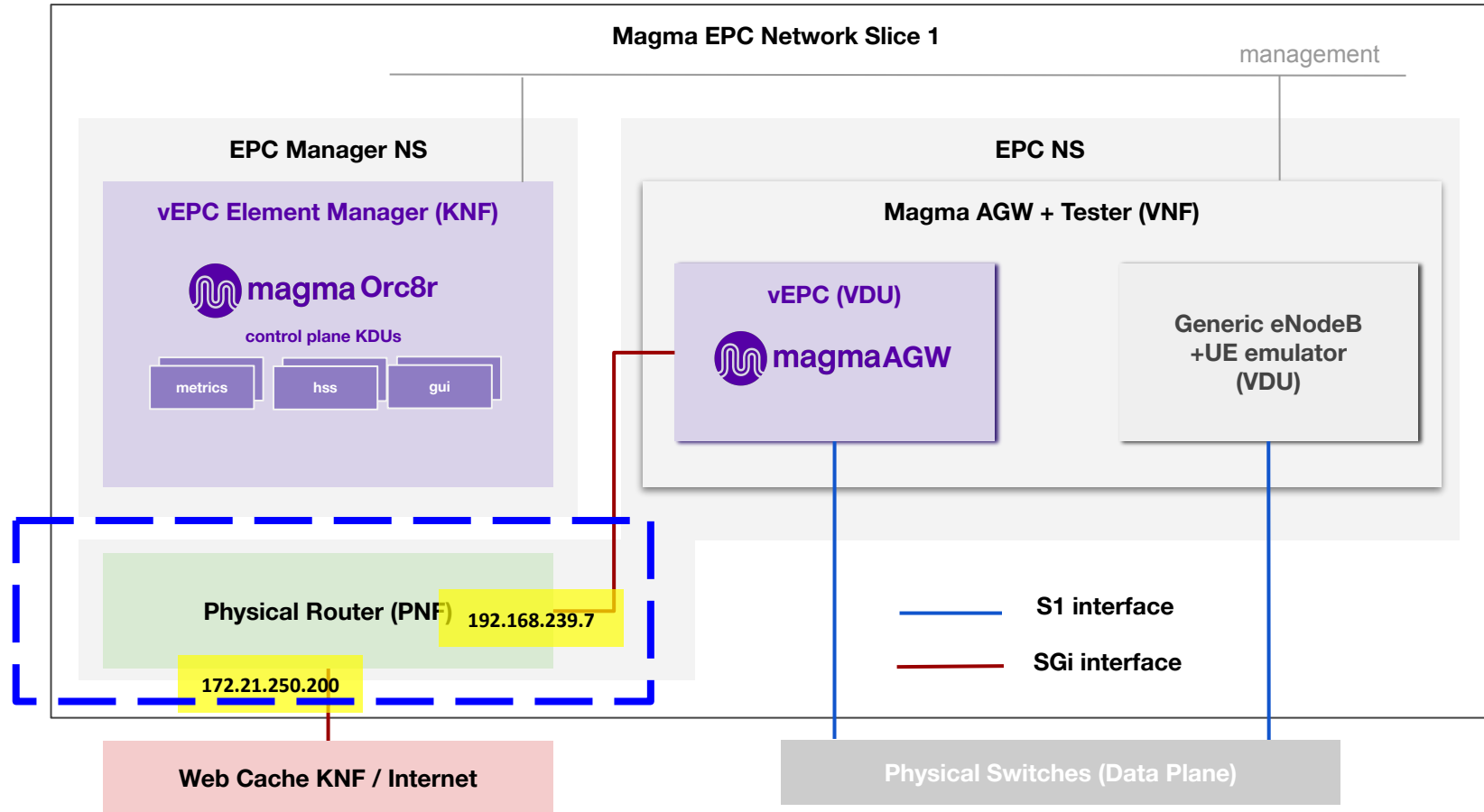
In those cases where we want to define NS packages consisting of PNF packages or HNF packages, **OSM needs to be instructed about the available PDUs.**

PNFD/HNFD vs PDU



Let's orchestrate the PNF!

Both Internet
exit and
management



(1) Tell OSM about the PDU instance(s)

Via the OSM CLI

a) Create a file that describes the PDU (for example, pdu.yaml)

```
name:      router01
description:  VyOS Router
type:      gateway
vim_accounts: [ YOUR_VIM_ID_HERE ]
shared:    false
interfaces:
- name:    eth0
  ip-address: 172.21.250.200
  vim-network-name: osm-ext
  mgmt:    true
- name:    eth1
  ip-address: 192.168.239.7
  mgmt:    false
```

b) Create the PDU in OSM:

```
osm pdu-create --descriptor_file pdu.yaml
```

(1) Tell OSM about the PDU instance(s)

Or via the OSM GUI

New PDU ×

Name * **PDU type ***

Vim Accounts *

Interfaces:

Name	<input type="text" value="eth0"/>	IP	<input type="text" value="172.21.250.200"/>	<input type="button" value="-"/>
Mgmt	<input type="text" value="True"/> ▼	Net name	<input type="text" value="osm-ext"/>	
Name	<input type="text" value="eth1"/>	IP	<input type="text" value="192.168.239.7"/>	<input type="button" value="+"/>
Mgmt	<input type="text" value="False"/> ▼	Net name	<input type="text" value="sgi"/>	

(2) Now create the PNFD

You will create it in the next section. The PNF uses the same model as a VNF and will look like this:

```
vnfd-catalog:
  vnfd:
    - connection-point:
      - name: gateway_public
        type: VPORT
      description: Gateway PNF
      id: hackfest_gateway_vnfd
      mgmt-interface:
        cp: gateway_public
        name: hackfest_gateway_vnfd
        short-name: hackfest_gateway_vnfd
      vdu:
        - description: gateway_pdu
          id: gateway_pdu
          interface:
            - external-connection-point-ref: gateway_public
              name: eth0
              type: EXTERNAL
          pdu-type: gateway
      vnf-configuration:
```

...

(3) Next, include it in your NSD

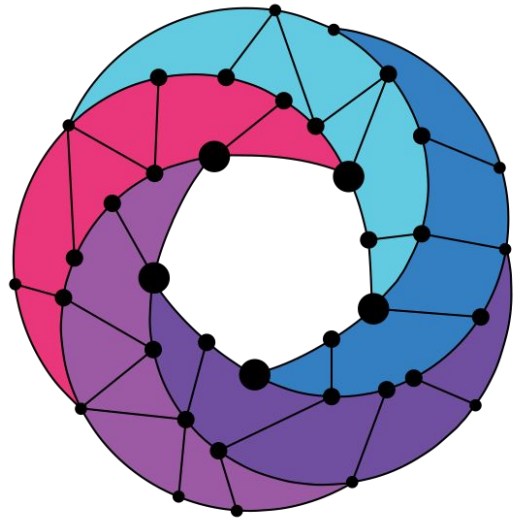
You will need to modify your NSD to include your new PNF. You will do it in the next section and it will become an Hybrid NS!

```
constituent-vnfd:
- member-vnf-index: 'MagmaAGW+srsLTE'
  vnfd-id-ref: hackfest_magma-agw-enb_vnfd
# - member-vnf-index: 'VYOS-PNF'
#   vnfd-id-ref: hackfest_gateway_vnfd
connection-point:
- name: nsd_cp_mgmt
  vld-id-ref: mgmt
- name: nsd_cp_sgi
  vld-id-ref: sgi
vld:
- id: mgmt
  name: mgmt
  short-name: mgmt
  type: ELAN
  mgmt-network: true
  vnfd-connection-point-ref:
- member-vnf-index-ref: 'MagmaAGW+srsLTE'
  vnfd-id-ref: hackfest_magma-agw-enb_vnfd
  vnfd-connection-point-ref: agw-mgmt
- member-vnf-index-ref: 'MagmaAGW+srsLTE'
  vnfd-id-ref: hackfest_magma-agw-enb_vnfd
  vnfd-connection-point-ref: srsLTE-mgmt
# - member-vnf-index-ref: 'VYOS-PNF'
#   vnfd-id-ref: hackfest_gateway_vnfd
#   vnfd-connection-point-ref: gateway_public
```


(4) Don't forget to add automation!

PNF orchestration, since is an element that already exists in the environment, is all about automating is configuration in the context of our Network Service.

Stay tuned for the **Automating Day 1 & 2 PNF Operations with OSM Primitives** session!



Open Source MANO

Find us at:

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