

Open Source MANO

Modelling Multi-VDU VNFs
Preethika P(Tata Elxsi)

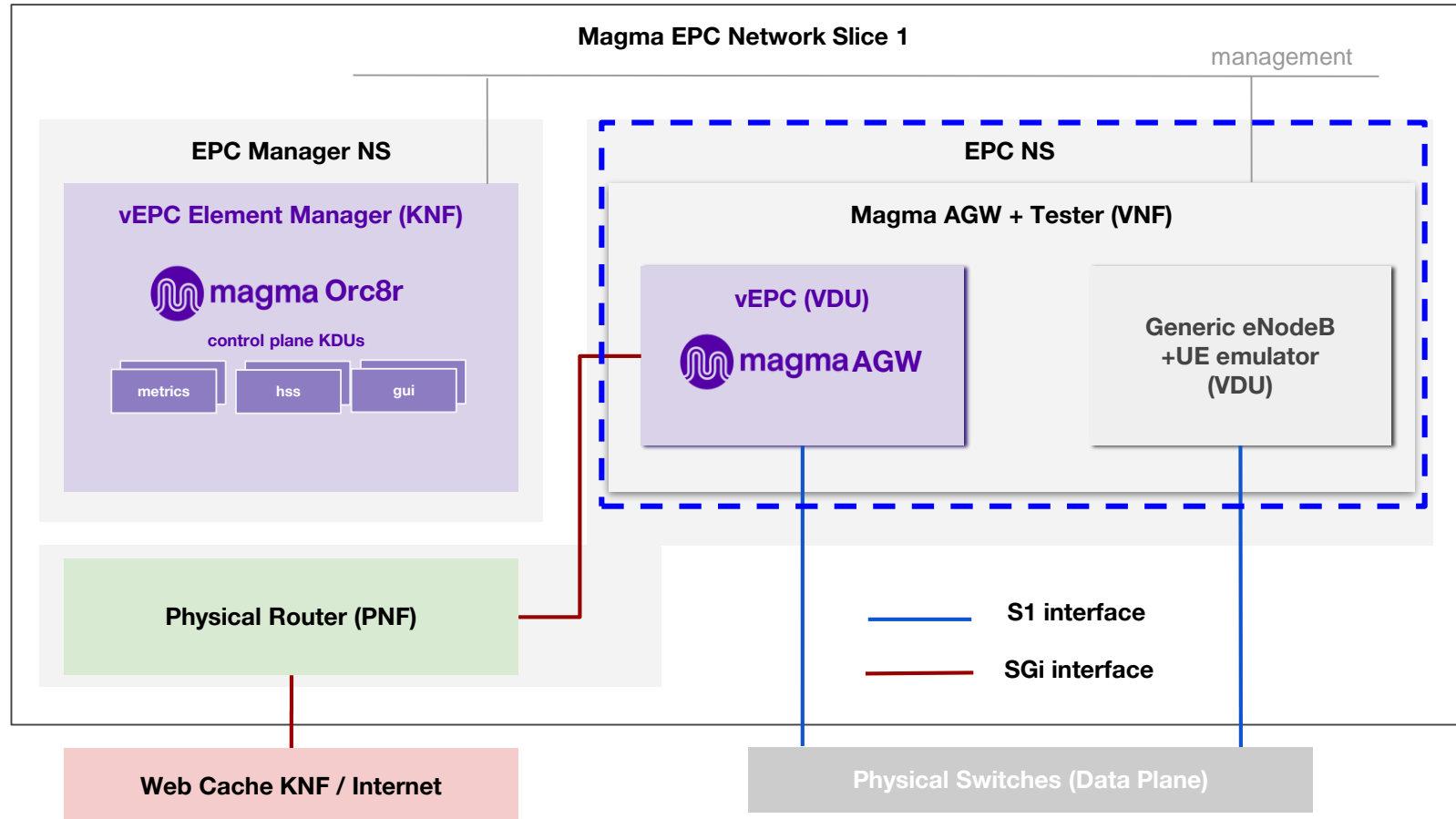


Open Source
MANO

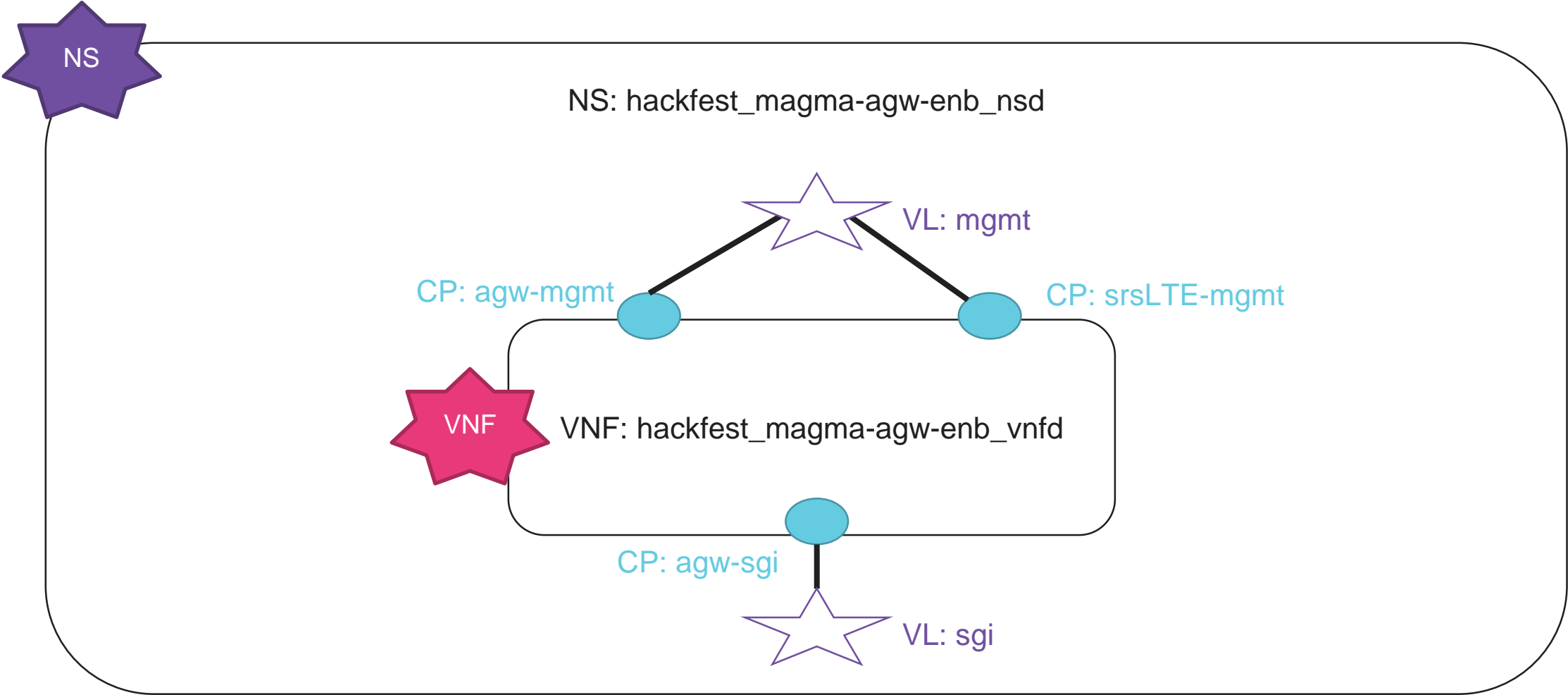
Modelling Multi- VDU VNFs



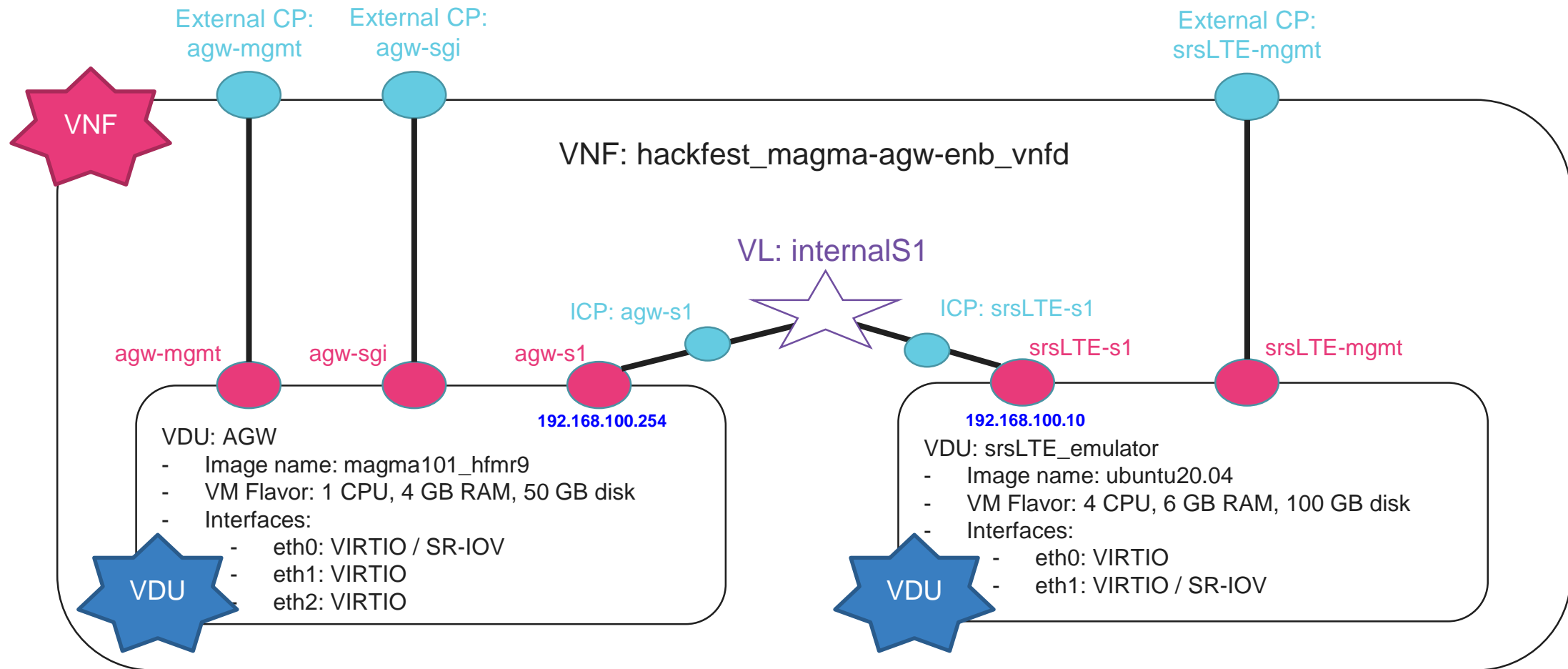
Let's start with the VNF



NS diagram

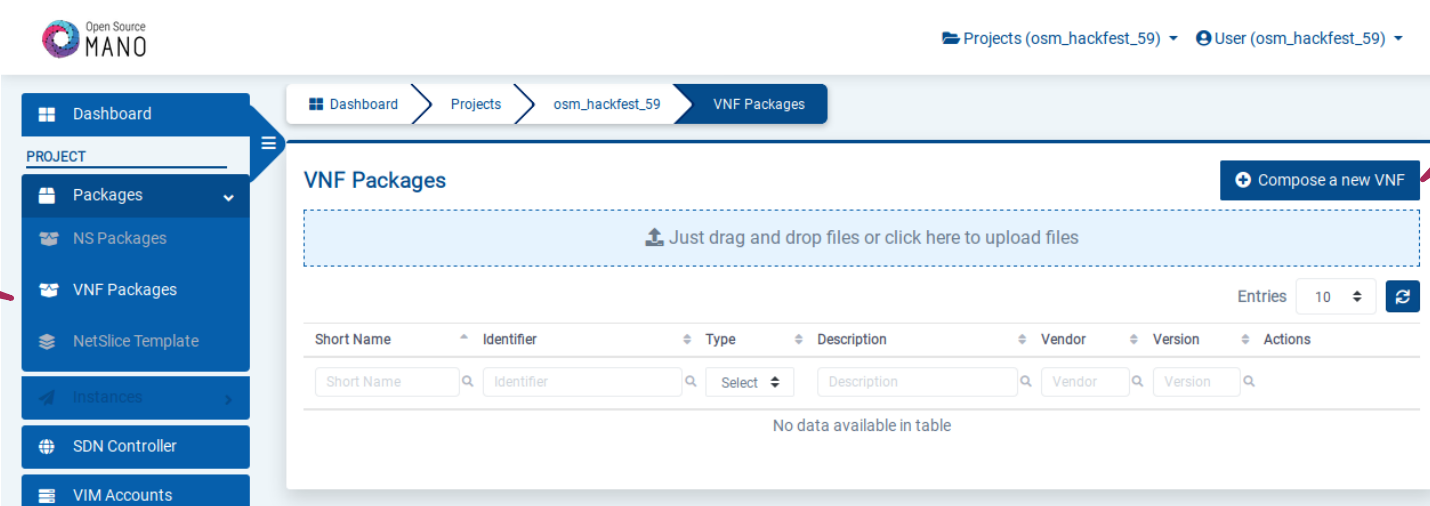


VNF diagram



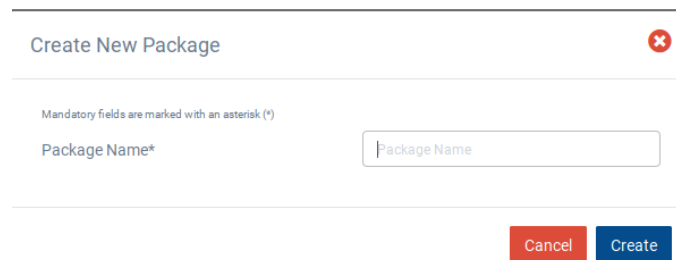
Building a Multi-VDU VNF Package from scratch

- We can use the graphical composer for the VNFD, then download the package to add other artifacts, but it will be faster through the CLI
 - Compose a new VNF



The screenshot shows the Open Source MANO interface. The left sidebar contains a menu with 'VNF Packages' highlighted. The main content area is titled 'VNF Packages' and features a 'Compose a new VNF' button. Below this is a dashed box for file uploads and a table with columns for Short Name, Identifier, Type, Description, Vendor, Version, and Actions. The table is currently empty, displaying 'No data available in table'. A pink callout bubble points to the 'Compose a new VNF' button with the text 'COMPOSE A NEW VNF'. Another pink callout bubble points to the 'VNF PACKAGES' menu item with the text 'VNF PACKAGES'.

- Create new Package



The screenshot shows the 'Create New Package' form. It includes a title bar with a close button, a note that 'Mandatory fields are marked with an asterisk (*)', and a text input field for 'Package Name*'. At the bottom, there are 'Cancel' and 'Create' buttons.

Creating a new VNF Package from CLI

- Use the command line to create the complete structure of the package, modify as desired with an editor

```
osm package-create --base-directory ~/magma --image magma101_hfmr9 --vcpu 1  
--memory 4096 --storage 50 --interfaces 2 --vendor OSM vnf hackfest_magma-  
agw-enb
```

- The final contents we need for this section

```
git clone --recurse-submodules -j8 https://osm.etsi.org/gitlab/vnf-  
onboarding/osm-packages.git
```

```
cd osm-packages/magma  
vi hackfest_magma-agw-enb_vnfd/magma-agw-enb_vnfd.yaml
```

- In our first VDU, interfaces section, we will make sure we have our internal “s1” interface first

```
vdu:  
- id: magma-agw-vdu  
  ...  
  interface:  
  - name: eth0  
    type: INTERNAL  
    position: 1  
    virtual-interface:  
      type: PARAVIRT  
    internal-connection-point-ref: agw-s1  
  - name: eth1  
    type: EXTERNAL  
    position: 2  
    virtual-interface:  
      type: PARAVIRT  
    external-connection-point-ref: agw-sgi
```

```
- name: eth2  
  type: EXTERNAL  
  position: 3  
  virtual-interface:  
    type: PARAVIRT  
  external-connection-point-ref: agw-mgmt  
internal-connection-point:  
- id: agw-s1  
  name: agw-s1  
  short-name: agw-s1  
  port-security-enabled: false
```


Magma-agw VNF Package

- The management interface for our VNF will be the agw-mgmt CP

```
mgmt-interface:  
  cp: agw-mgmt
```

- Our Magma AGW VDU needs some information to be passed via a cloud-init file, which we will review later

```
vdu:  
- id: magma-agw-vdu  
  ...  
  cloud-init-file: magmaagw_init
```

- A second VDU is added, for the srsLTE eNodeB/UE emulator

```
vdu:  
  ...  
- id: srsLTE-vdu  
  name: srsLTE-vdu  
  description: srsLTE-vdu  
  count: 1  
  cloud-init-file: srslte_init  
  vm-flavor:  
    vcpu-count: 4  
    memory-mb: 6144  
    storage-gb: 100  
  image: 'ubuntu20.04'
```

```
interface:  
- name: eth0  
  type: EXTERNAL  
  virtual-interface:  
    type: PARAVIRT  
  external-connection-point-ref: srsLTE-mgmt  
  mgmt-interface: true  
- name: eth1  
  type: INTERNAL  
  virtual-interface:  
    type: PARAVIRT  
  internal-connection-point-ref: srsLTE-s1  
internal-connection-point:  
- id: srsLTE-s1  
  name: srsLTE-s1  
  short-name: srsLTE-s1
```

- The internal VLD, for the S1 network, must be defined in the VNFD. An IP Profile is used to force a specific IP addressing

```
internal-vld:
- id: internalS1
  name: internalS1
  short-name: internalS1
  type: ELAN
  ip-profile-ref: internalS1
  internal-connection-point:
  - id-ref: agw-s1
    ip-address: 192.168.100.254
  - id-ref: srsLTE-s1
    ip-address: 192.168.100.10
ip-profiles:
- name: internalS1
  description: S1 test network
  ip-profile-params:
    ip-version: ipv4
    subnet-address: 192.168.100.0/24
  dhcp-params:
    enabled: true
```

- Finally, the external connection points that the VNF will expose, are defined

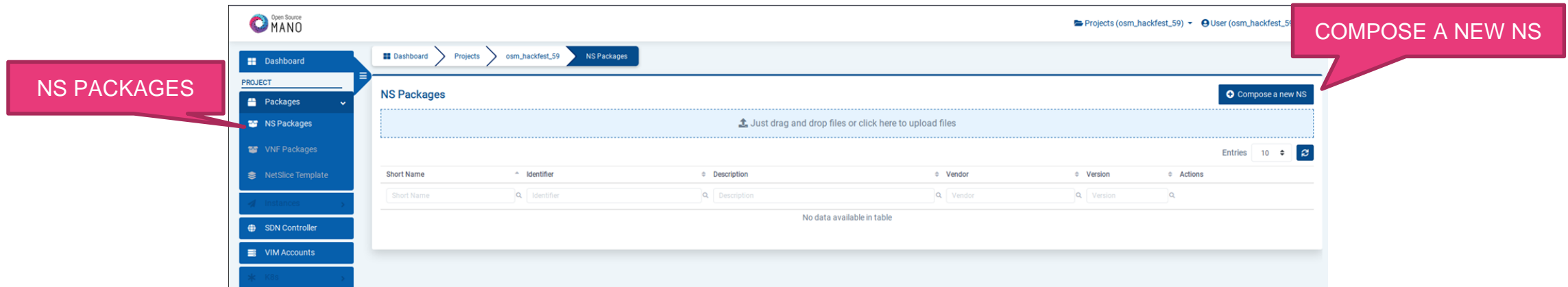
```
connection-point:  
- name: agw-mgmt  
- name: agw-sgi  
- name: srsLTE-mgmt
```

We are exposing the two management ports of both VDUs, and the SGi interface, to the Network Service

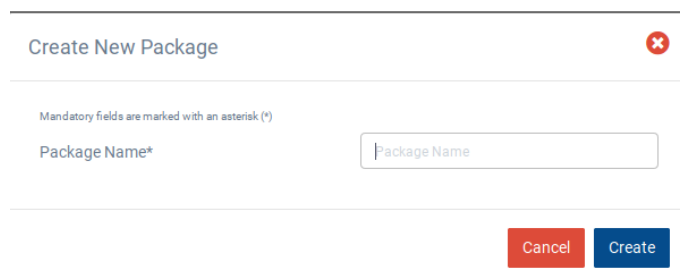
Building a NS Package from scratch

We can use the graphical composer for the NSD, then download the package to add other artifacts, but it will be faster through the CLI

- Compose a new NS



- Create new Package



Creating a new NS Package from CLI

- Use the command line to create the complete structure of the package, modify as desired with an editor

```
osm package-create --base-directory ~/magma --vendor OSM ns hackfest_magma-  
agw-enb
```

- The final contents needed for this section

```
cd osm-packages/magma  
vi hackfest_magma-agw-enb_nsd/magma-agw-enb_nsd.yaml
```

Magma-agw NS Package

```
nsd-catalog:
  nsd:
  - id: hackfest_magma-agw-enb_nsd
    name: hackfest_magma-agw-enb_nsd
    short-name: hackfest_magma-agw-enb_nsd
    description: Magma AGW 1.0.0 with tools & srsLTE connected to PNF Gateway
    vendor: Whitestack
    version: '1.0'
    constituent-vnfd:
    - member-vnf-index: 'MagmaAGWsrsLTE'
      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: 'MagmaAGWsrsLTE'
    vnfd-id-ref: hackfest_magma-agw-enb_vnfd
    vnfd-connection-point-ref: agw-mgmt
  - member-vnf-index-ref: 'MagmaAGWsrsLTE'
    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
```

```
- id: sgi
  name: sgi
  short-name: sgi
  type: ELAN
  mgmt-network: false
  vim-network-name: sgi
  vnfd-connection-point-ref:
  - member-vnf-index-ref: 'MagmaAGWsrsLTE'
    vnfd-id-ref: hackfest_magma-agw-enb_vnfd
    vnfd-connection-point-ref: agw-sgi
  - member-vnf-index-ref: 'VYOS-PNF'
    vnfd-id-ref: hackfest_gateway_vnfd
    vnfd-connection-point-ref: gateway_public
```

- PNF will be covered in later session

- The first important part is the ‘constituent-vnfd’ section, which will specify which VNFs form our NS

```
constituent-vnfd:  
- member-vnf-index: 'MagmaAGWsrsLTE'  
  vnfd-id-ref: hackfest_magma-agw-enb_vnfd
```


- Our management VLD will connect all the external management CPs exposed at our VNF

```
vld:  
- id: mgmt  
  name: mgmt  
  short-name: mgmt  
  type: ELAN  
  mgmt-network: true  
  vnfd-connection-point-ref:  
  - member-vnf-index-ref: 'MagmaAGWsrsLTE'  
    vnfd-id-ref: hackfest_magma-agw-enb_vnfd  
    vnfd-connection-point-ref: agw-mgmt  
  - member-vnf-index-ref: 'MagmaAGWsrsLTE'  
    vnfd-id-ref: hackfest_magma-agw-enb_vnfd  
    vnfd-connection-point-ref: srsLTE-mgmt
```

- Finally, our SGI VLD will connect the Magma AGW VDU to a existing network called “sgi” in our VIM

```
vld:  
  ...  
- id: sgi  
  name: sgi  
  short-name: sgi  
  type: ELAN  
  mgmt-network: false  
  vim-network-name: sgi  
  vnfd-connection-point-ref:  
  - member-vnf-index-ref: 'MagmaAGWsrsLTE'  
    vnfd-id-ref: hackfest_magma-agw-enb_vnfd  
    vnfd-connection-point-ref: agw-sgi
```



Open Source
MANO

Automating Day-0
configuration
through cloud-init



What is cloud-init and what can it be used for?

- It is a Linux package used to automate initial configuration of a VM
- VM requirements:
 - Cloud-init package
 - Cloud-init configuration (data source) via `/etc/cloud/cloud.cfg`
 - Config drive
 - Openstack metadata server
 - ...
- What can be done?
 - Setting a default locale
 - Setting an instance hostname
 - Generating instance SSH private keys
 - Adding SSH keys to a user's `.ssh/authorized_keys` so they can log in
 - Setting up ephemeral mount points
 - Configuring network devices
 - Adding users and groups
 - Adding files
- Docs: <http://cloudinit.readthedocs.io/en/latest/>

- Cloud-init is available in Linux VMs and might be supported in other OS
- Not all VIMs support cloud-init via a metadata server

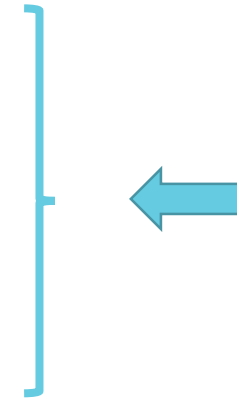
Let's explore the Cloud-init files

~/osm-packages/magma/hackfest_magma-agw-enb_vnfd/cloud_init/magmaagw_init

```
#cloud-config
```

```
runcmd:
```

```
# deleting default mgmt route to Internet
- route delete -net 0.0.0.0/0 gw 172.21.251.254
# adding specific ETSI HIVE mgmt segments through mgmt network
- route add -net 10.100.0.0/16 gw 172.21.251.254
- route add -net 10.101.0.0/16 gw 172.21.251.254
- route add -net 172.21.0.0/16 gw 172.21.251.254
- route add -net 172.22.0.0/16 gw 172.21.251.254
- route add -net 192.168.170.0/24 gw 172.21.251.254
# adding specific ORCH_IP through mgmt network
- route add -host {{ orch_ip }}/32 gw 172.21.251.254
# adding new default route to VyOS PNF
- route add -net 0.0.0.0/0 gw 192.168.239.7
# adding new specific routes to reach MetallB ranges (Squid and other svcs) through VyOS PNF
- route add -net 172.21.250.0/24 gw 192.168.239.7
- route add -net 172.21.251.0/24 gw 192.168.239.7
```

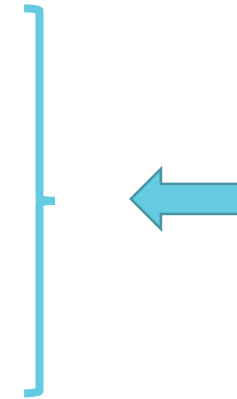


- From the AGW, we are removing the default route and pointing it towards the data plane interface (router at the SGI, at 192.168.239.7)
- We are also passing a route towards the Magma Orc8r, through the management port, the IP is parametrized!

Let's explore the Cloud-init files

~/osm-packages/magma/hackfest_magma-agw-enb_vnfd/cloud_init/srslte_init

```
#cloud-config
password: osm2020
chpasswd: { expire: False }
ssh_pwauth: True
packages:
  - net-tools
runcmd:
  - route add -net 10.0.0.0/8 gw 172.21.251.254
  - route add -net 172.21.0.0/16 gw 172.21.251.254
  - route add -net 172.22.0.0/16 gw 172.21.251.254
  - route add -net 192.168.170.0/24 gw 172.21.251.254
```



- From the srsLTE emulator, we are adding some specific management routes towards the management network.
- We are also setting a fixed password, 'osm2020', for the default 'ubuntu' user.



Open Source
MANO

Packaging and
instantiation



Building, validating and uploading packages

- Once finished, you can build and upload the NS/VNF Package to OSM with the following commands

```
osm nfpkg-create <path to vnf package>  
osm nspkg-create <path to ns package>
```

- This single command will:
 - **Validate** the package according to the Information Model.
 - **Build** the package.
 - **Upload** the package to OSM.

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


Uploaded Packages

```
$ osm nfpkg-list
```

```
+-----+-----+
| nfpkg name          | id                               |
+-----+-----+
| fb_magma_knf        | 8022da76-52d9-4d4c-b3ff-7a4f7a1692c5 |
| hackfest_magma-agw-enb_vnfd | 6a43a3f0-3a77-4900-beba-2d6f02e04d80 |
| hackfest_gateway_vnfd      | 49277219-a256-410b-919e-ffcb883a4c3e |
+-----+-----+
```

```
$ osm nspkg-list
```

```
+-----+-----+
| nsd name            | id                               |
+-----+-----+
| hackfest_magma-agw-enb_nsd | cf78e99a-2abb-4896-a766-db941e31a26f |
| fb_magma_ns         | 55746c75-278f-44b3-b750-929a7bbd3fc4 |
+-----+-----+
```

Uploaded Packages

```
$ osm netslice-template-list
+-----+-----+
| nst name          | id                               |
+-----+-----+
| magma_slice_hackfest_nst | 31e1ebb5-de12-486b-a69a-1f47b2001c57 |
+-----+-----+
```

Adding helm Repo, netslice-template and onboarding pdu(If you had missed in earlier session!!)

- `osm repo-add --type helm-chart --description "Repository for Facebook Magma helm Chart" magma https://felipevicens.github.io/fb-magma-helm-chart/`
`cd osm-packages/magma`
- `osm netslice-template-create magma_slice.yaml`
- `VIMID=`osm vim-list | grep "etsi-openstack " | awk '{ print $4 }'``
- `sed -i "s/vim_accounts: ./vim_accounts: [$VIMID]/" pdu.yaml`
- `osm pdu-create --descriptor_file pdu.yaml`

Instantiation parameters

- Prepare any parameter you want to pass during instantiation. In this case, we will prepare a **'params.yaml'** file that will pass some parameters we will need during the following tests

```
netslice-subnet:
- id: slice_hackfest_nsd_epc
  #placement-engine: PLA
  #wimAccountId: False
  additionalParamsForVnf:
  - member-vnf-index: 'MagmaAGWsrsLTE'
    additionalParams:
      agw_id: 'agw_100'
      agw_name: 'AGW100'
      orch_ip: '172.21.251.x' ## change this to the MetalLB IP address of your
orc8r_proxy service.
      orch_net: 'osmnet'

- id: slice_hackfest_nsd_epcmgmt
  additionalParamsForVnf:
  - member-vnf-index: 'orc8r'
    additionalParamsForKdu:
    - kdu_name: orc8r
      additionalParams:
        proxyserviceloadBalancerIP: '172.21.251.x' # MetalLB IP Address
```

Launch your instance

- With your Netslice template, NS and VNF package ready, you can proceed to instantiation.

```
osm nsi-create --nsi_name magma_slice_x --nst_name magma_slice_hackfest_nst --  
config_file params.yaml --vim_account etsi-openstack-x
```

- To verify

```
osm netslice-instance-list  
osm ns-list
```

If you have modelled network service rather than slice, command to instantiate Network service alone:

```
osm ns-create --ns_name <network-service-name> --nsd_name <nsd-package-name>  
--vim_account <vim-account-name> --config_file params.yaml
```

Verify your instance

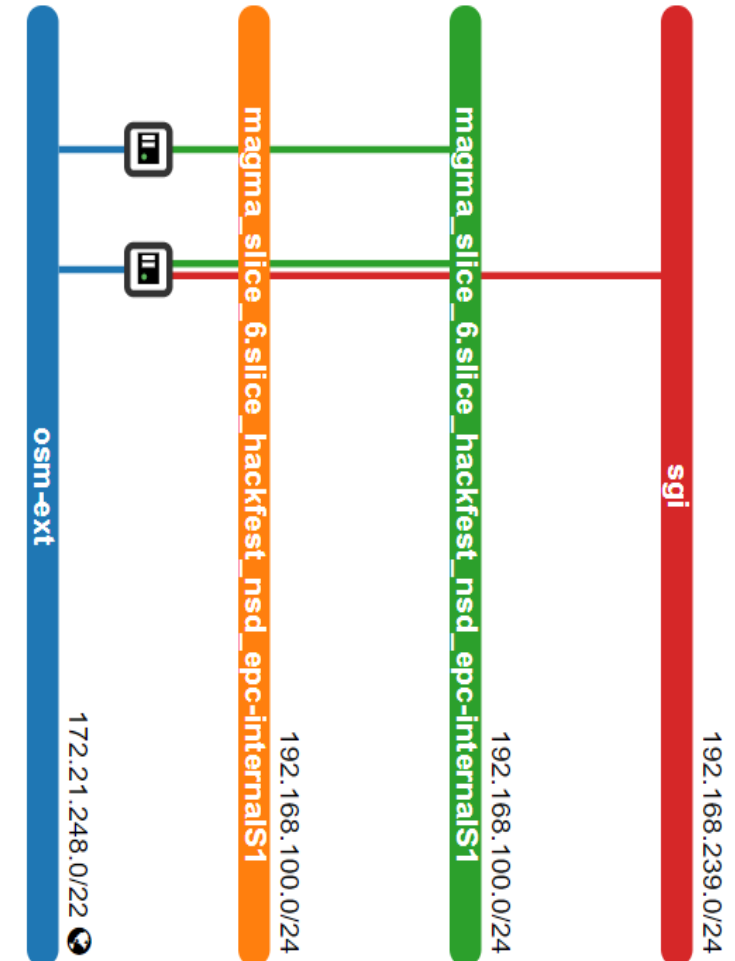
NetSlice Instances

🕒 init ✅ running / configured ❌ failed

[Create NSI](#)

Entries 10 [↻](#)

Name	Identifier	Nst name	Operational Status	Config Status	Detailed Status	Actions
<input type="text" value="Name"/>	<input type="text" value="Identifier"/>	<input type="text" value="Nst name"/>	<input type="text" value="Select"/>	<input type="text" value="Select"/>	<input type="text" value="Detailed Status"/>	
magma_slice_6	d203adb5-60bc-4f9b-b8b5-82566cf34b7f	magma_slice_hackfest_nst	✅	✅	done	i 🗑️ Action



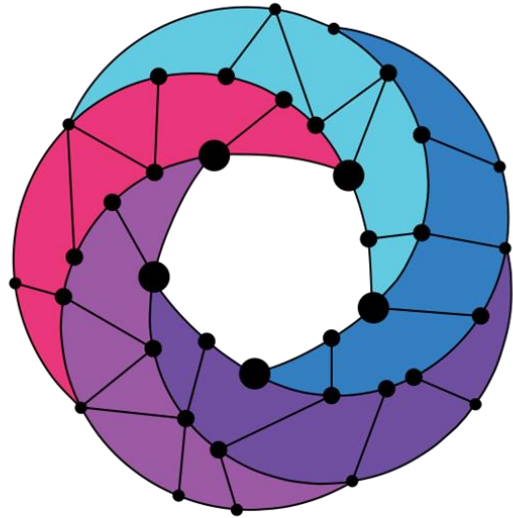
NS Instances

🕒 init ✅ running / configured ❌ failed

[New NS](#)

Entries 10 [↻](#)

Name	Identifier	Nsd name	Operational Status	Config Status	Detailed Status	Actions
<input type="text" value="Name"/>	<input type="text" value="Identifier"/>	<input type="text" value="Nsd name"/>	<input type="text" value="Select"/>	<input type="text" value="Select"/>	<input type="text" value="Detailed Status"/>	
magma_slice_6.slice_hackfest_est_nsd_epc	cc16a248-f82b-4be5-a5cf-07365b50335f	hackfest_magma-agw-enb_nsd	✅	✅	Done	📄 👤 🗑️ Action
magma_slice_6.slice_hackfest_est_nsd_epcgmt	8573872a-23e0-488a-bf55-0d9ccc469661	fb_magma_ns	✅	✅	Done	📄 👤 🗑️ Action



Open Source MANO

Find us at:

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