

Open Source  
**MANO**

## OSM Hackfest – Guidelines for VNF builders

Adrián Candel (Altran)  
Guillermo Calviño (Altran)

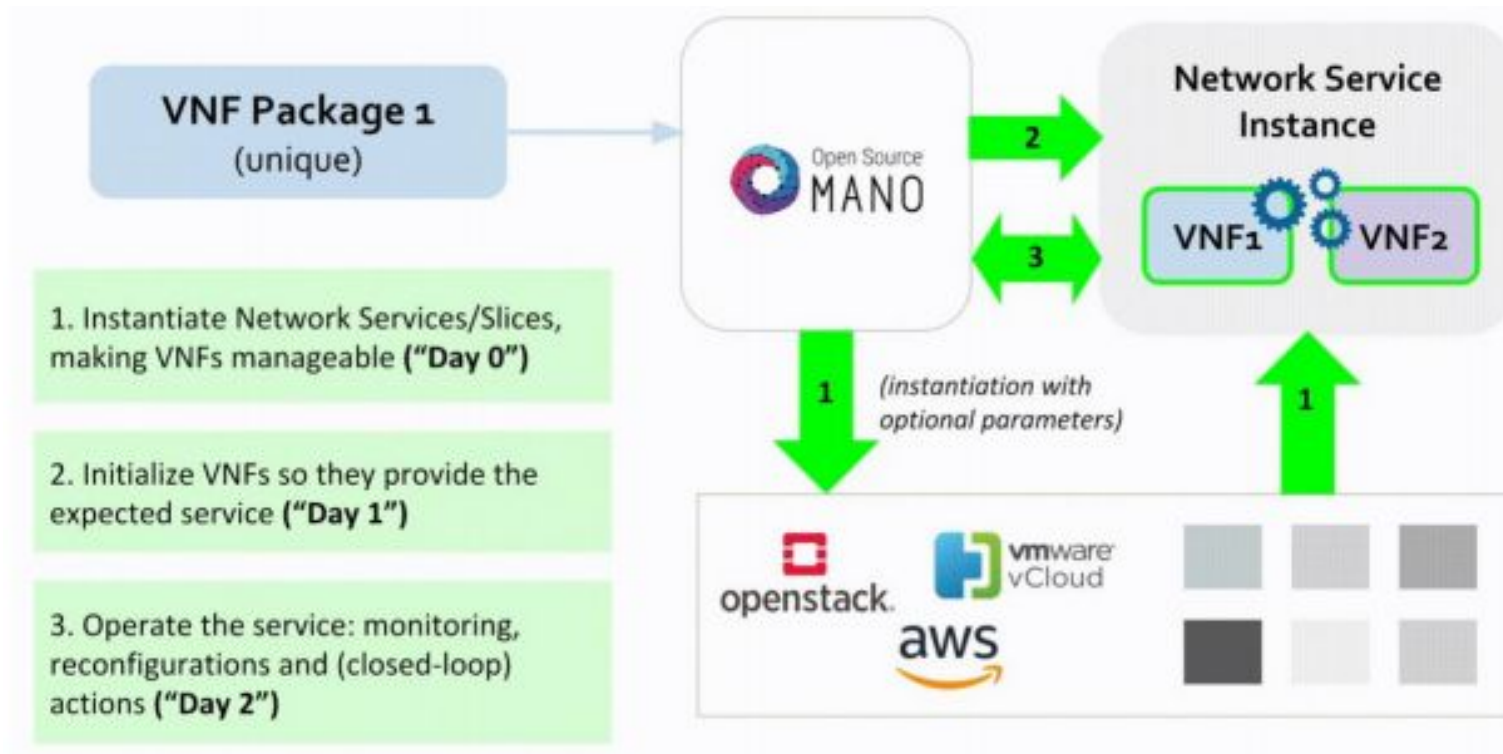


# VNF configurations for lifecycle stages

- Basic Instantiation (Day 0): the VNF is instantiated and the management access is established.
- Service Initialization (Day 1): configure the VNF so it starts providing the expected service.
- Runtime Operations (Day 2): re-configure the VNF so its behaviour can be modified during runtime, as well as be able to monitor its main KPIs and run scaling actions over it.

# VNF configurations for lifecycle stages

- Lifecycle stages



# Day 0 - Basic Instantiation

- Description of each VNF component
- Definition of NFVI requirements
  - Compute performance attributes:
    - CPU Pinning
    - NUMA Topology Awareness
    - Memory Page Size
  - Data plane performance attributes:
    - PCI-Passthrough
    - SR-IOV

# Day 0 – Basic instantiation

VNF name	VNF Description	VDU name	Image name	Flavor			N° ifaces	GUEST EPA INFO				
				vCPU	vMem	vDisk		Mempage size	Dedicated CPUs (YES/NO)	Cores or HW threads	Strict NUMA Mem Policy (YES/NO)	SR-IOV

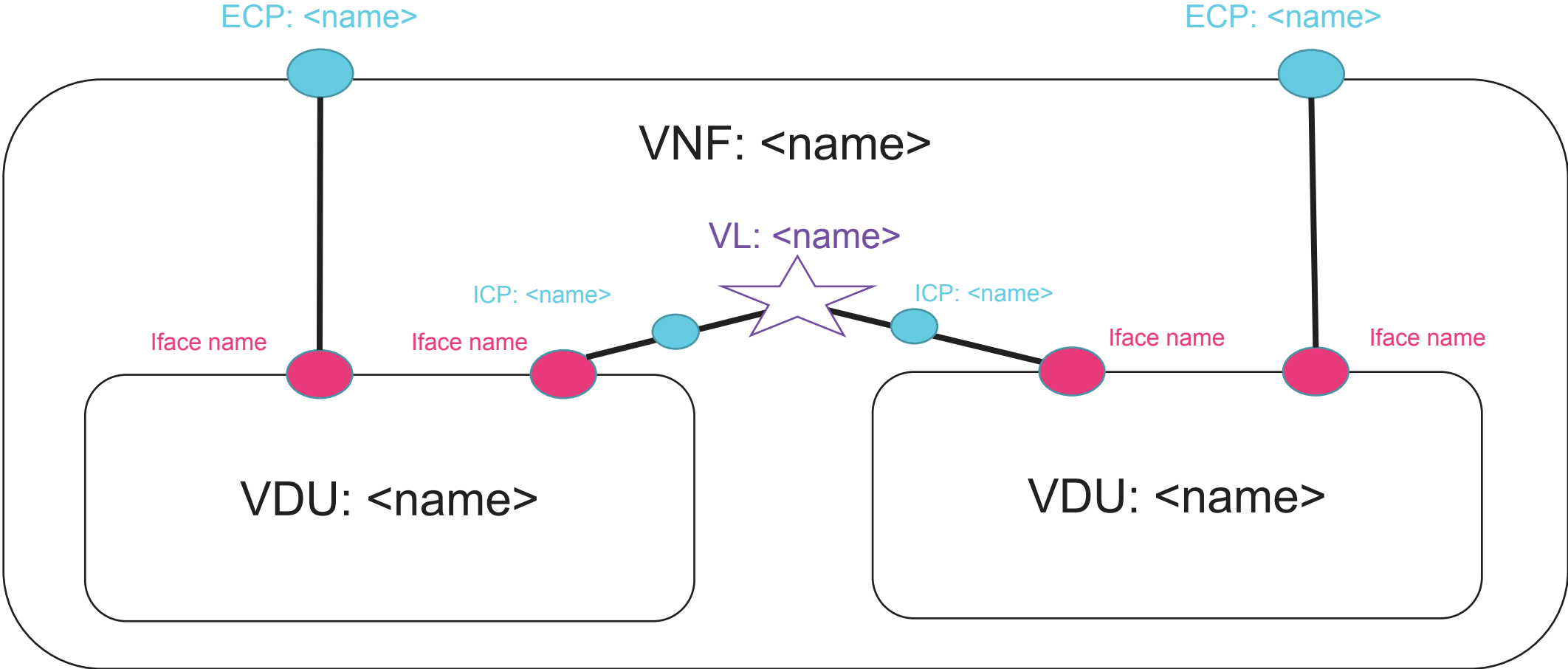
This table is a reference table. Other parameters could be added, such as the existence of a cloud-init file for each VDU, if the VDU has a charm, etc.

More information: [OSM Day 0 Guidelines](#)

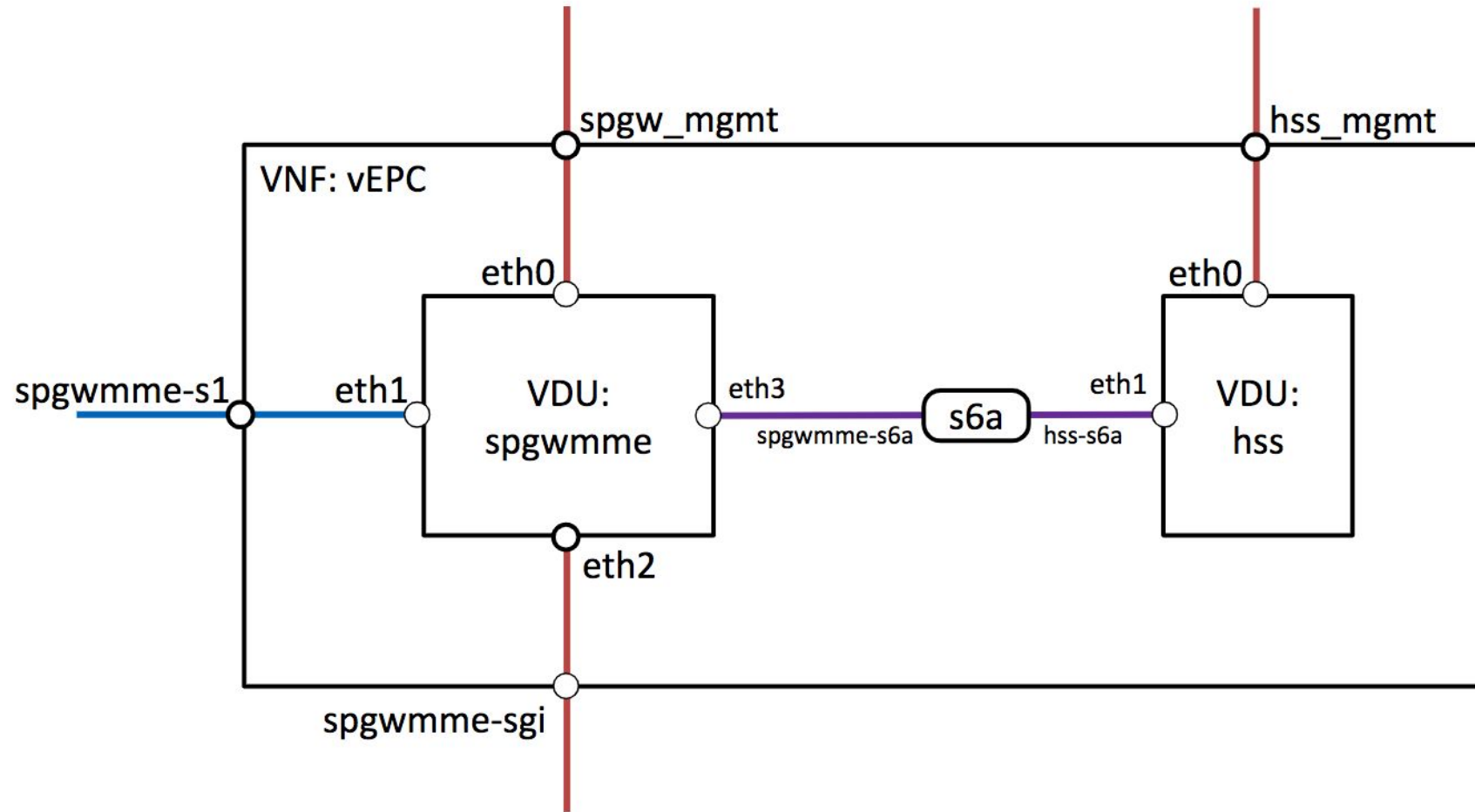
# vEPC VNF Day 0 - collected information Example

VNF name	VNF Description	VDU name	Image name	Flavor			N° ifaces	GUEST EPA INFO				
				vCPU	vMem	vDisk		Mempage size	Dedicated CPUs (YES/NO)	Cores or HW threads	Strict NUMA Mem Policy (YES/NO)	SR-IOV
vEPC	Single VDU containing SGW, PGW and MME	spgwmm	nextepc-spgwmm-e-base	2	4	10	4	LARGE	YES	NO	NO	YES
vEPC	HSS VDU	hss	nextepc-hss-base	1	2	10	2	-	-	-	-	-

# Build your VNF diagram

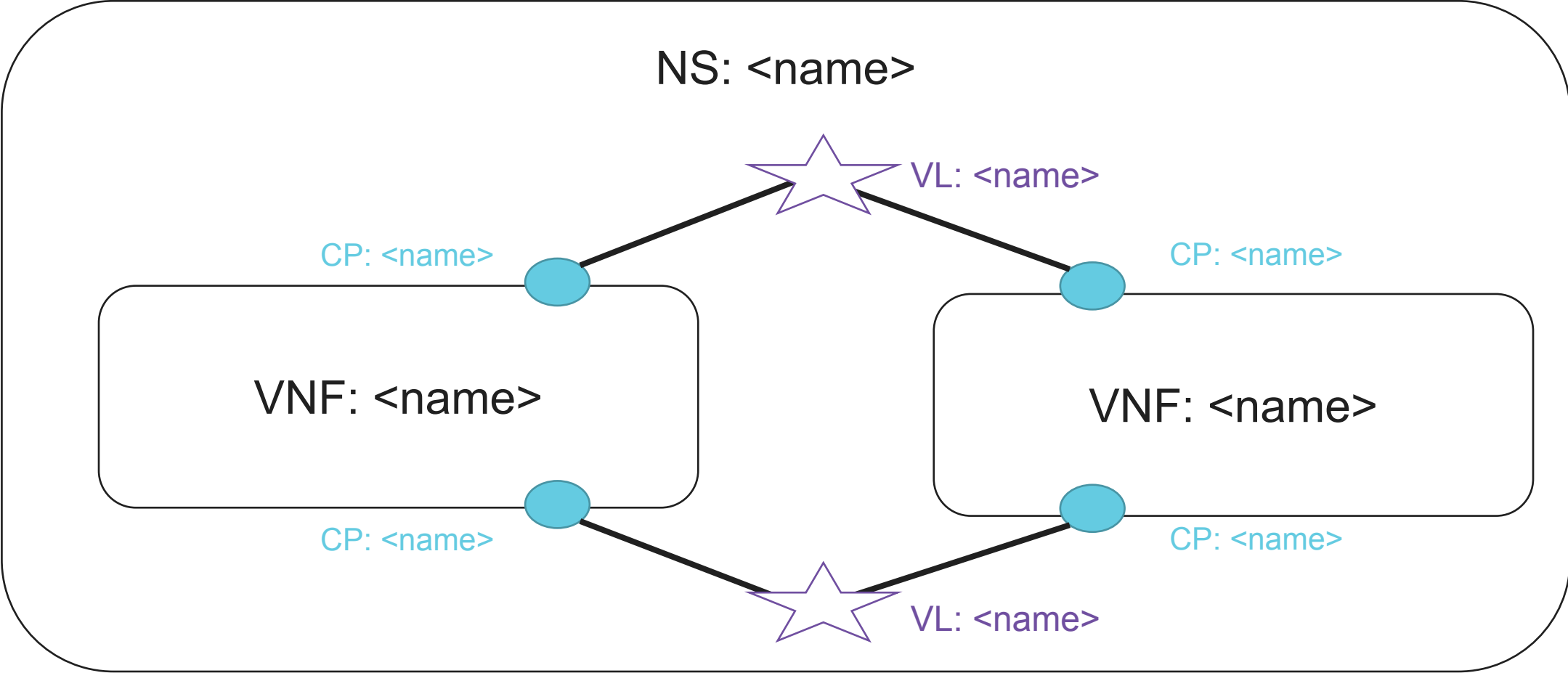


# vEPC Diagram example





# Build your NS diagram



# Day 0 - Basic Instantiation - configurations

- Minimal configuration of the VNFs can be injected via cloud-init
- Example:

```
#cloud-config
  hostname: my_first_vnf
  password: osm4u
  chpasswd: { expire: False }
  ssh_pwauth: True
```

- Identifying the instantiation parameters
- Associate cloud-init-file in corresponding VDU

# Day 1 - Service Initialization

The goal of Day 1 is the automatic initialization of VNF services right after the instantiation

# Day 1 - Service Initialization

- Identifying dependencies between components
  - IP address for connectivity
- Defining the required configuration for service initialization
  - Start some interfaces
  - Replace values in configuration files
  - Start services inside the VNF
- Identifying the need for instantiation parameters
  - External endpoints to configure

# Day 1 - Service Initialization

- The process after identification comprises:
  - Building a Proxy Charm
    - **Method 1: Building a Proxy Charm the traditional way**
    - **Method 2: Using Proxy Charm Generators**

# Day 1 - Service Initialization

## Example:

### Collection of commands

```
sudo ip link set ens4 up && sudo dhclient ens4  
sudo ip link set ens5 up && sudo dhclient ens5  
sudo ip link set ens6 up && sudo dhclient ens6
```

```
sudo sed -i 's/$hss_ip/HSS_IP/g' /etc/nextepc/freeDiameter/mme.conf  
sudo sed -i 's/$spgw_ip/SPGW_IP/g' /etc/nextepc/freeDiameter/mme.conf
```

### Charm definition

```
@when('actions.configure-spgw')  
def configure_spgw():  
    hss_ip = action_get('hss-ip')  
    spgw_ip = action_get('spgw-ip')  
    cmd1 = "sudo ip link set ens4 up && sudo dhclient ens4"  
    charms.sshproxy._run(cmd1)  
    cmd2 = "sudo ip link set ens5 up && sudo dhclient ens5"  
    charms.sshproxy._run(cmd2)  
    cmd3 = "sudo ip link set ens6 up && sudo dhclient ens6"  
    charms.sshproxy._run(cmd3)  
    cmd3='sudo sed -i "\s/$hss_ip/{}g\" /etc/nextepc/freeDiameter/mme.conf'.format(hss_ip)  
    charms.sshproxy._run(cmd3)  
    cmd4='sudo sed -i "\s/$spgw_ip/{}g\" /etc/nextepc/freeDiameter/mme.conf'.format(spgw_ip)  
    charms.sshproxy._run(cmd4)  
    remove_flag('actions.configure-spgw')
```

# Day 1 - Service Initialization

## Example:

Day 1 = initial-config-primitives

### Charm definition

```
@when('actions.configure-spgw')
def configure_spgw():
    hss_ip = action_get('hss-ip')
    spgw_ip = action_get('spgw-ip')
    cmd1 = "sudo ip link set ens4 up && sudo dhclient ens4"
    charms.sshproxy._run(cmd1)
    cmd2 = "sudo ip link set ens5 up && sudo dhclient ens5"
    charms.sshproxy._run(cmd2)
    cmd3 = "sudo ip link set ens6 up && sudo dhclient ens6"
    charms.sshproxy._run(cmd3)
    cmd3='sudo sed -i "\s/$hss_ip/{}/g\" /etc/nextepc/freeDiameter/mme.conf'.format(hss_ip)
    charms.sshproxy._run(cmd3)
    cmd4='sudo sed -i "\s/$spgw_ip/{}/g\" /etc/nextepc/freeDiameter/mme.conf'.format(spgw_ip)
    charms.sshproxy._run(cmd4)
    remove_flag('actions.configure-spgw')
```

### VNF Descriptor definition

```
vnf-configuration:
  initial-config-primitive:
  - seq: '1'
    name: config
    parameter:
    - name: ssh-hostname
      value: <rw_mgmt_ip>
    - name: ssh-username
      value: ubuntu
    - name: ssh-password
      value: <password>
  - seq: '2'
    name: configure-spgw
    parameter:
    - name: spgw-ip
      data-type: STRING
      value: <spgw_ip>
    - name: hss-ip
      data-type: STRING
      value: <hss_ip>
```

# Day 2 – Runtime Operations

The goal of Day 2 is the reconfiguration of the services  
and service monitoring



# Day 2 – Runtime Operations

- Adding Day-2 primitives to the descriptor
  - Used to operate the service for example:
    - Clean a cache
    - Install a route
    - Restart a service
    - Create and restore a backup

# Day 2 – Runtime Operations

- Example

## Collection of commands

```
sudo route add -net $prefix gw $next_hop
```

Day 2 = config-primitives

## VNF Descriptor definition

```
vnf-configuration:  
  config-primitive:  
    name: add-route  
    parameter:  
      - name: external-prefix  
        data-type: STRING  
        default-value: '8.8.8.8/32'  
      - name: next-hop  
        data-type: STRING  
        default-value: '192.168.2.1'
```

## Charm definition

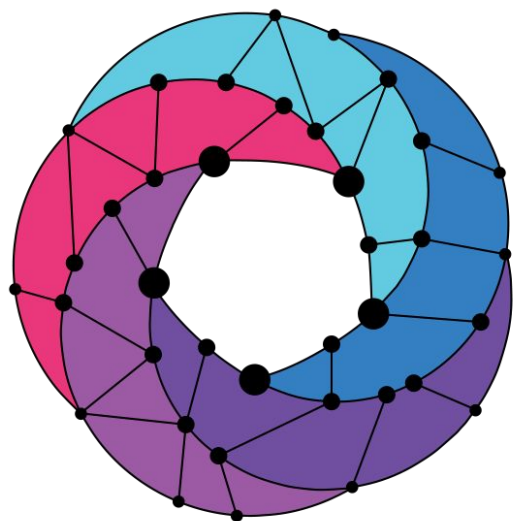
```
@when('actions.add-route')  
def add_route():  
    prefix = action_get('external-prefix')  
    next_hop = action_get('next-hop')  
    cmd = "sudo route add -net " + prefix + " gw " + next_hop  
    charms.sshproxy._run(cmd)  
    remove_flag('actions.add-route')
```

# Day 2 – Runtime Operations

- Monitoring metrics definition
  - [NFVI metrics](#)
  - [VNF Indicators - proxy charms with metrics layer](#)
- [Scaling Operators](#)

Example: nfvi metrics definition

```
vdu:  
...  
- id: spgwmme  
...  
  monitoring-param:  
  - id: "spgw_cpu_util"  
    nfvi-metric: "cpu_utilization"  
  - id: "spgw_memory_util"  
    nfvi-metric: "average_memory_utilization"  
...  
  monitoring-param:  
  - id: "spgw_cpu_util"  
    name: "spgw_cpu_util"  
    aggregation-type: AVERAGE  
    vdu-monitoring-param:  
      vdu-ref: "spgwmme"  
      vdu-monitoring-param-ref: "spgw_cpu_util"  
  - id: "spgw_memory_util"  
    name: "spgw_memory_util"  
    aggregation-type: AVERAGE  
    vdu-monitoring-param:  
      vdu-ref: "spgwmme"  
      vdu-monitoring-param-ref: "spgw_memory_util"
```



# Open Source MANO

Find us at:

[osm.etsi.org](https://osm.etsi.org)  
[osm.etsi.org/wikipub](https://osm.etsi.org/wikipub)