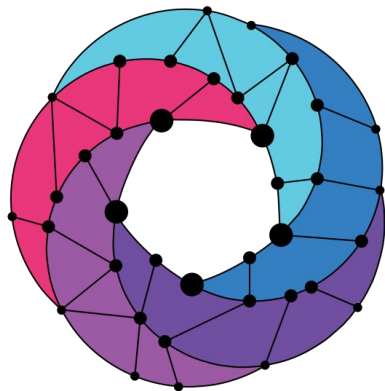




Open Source
MANO

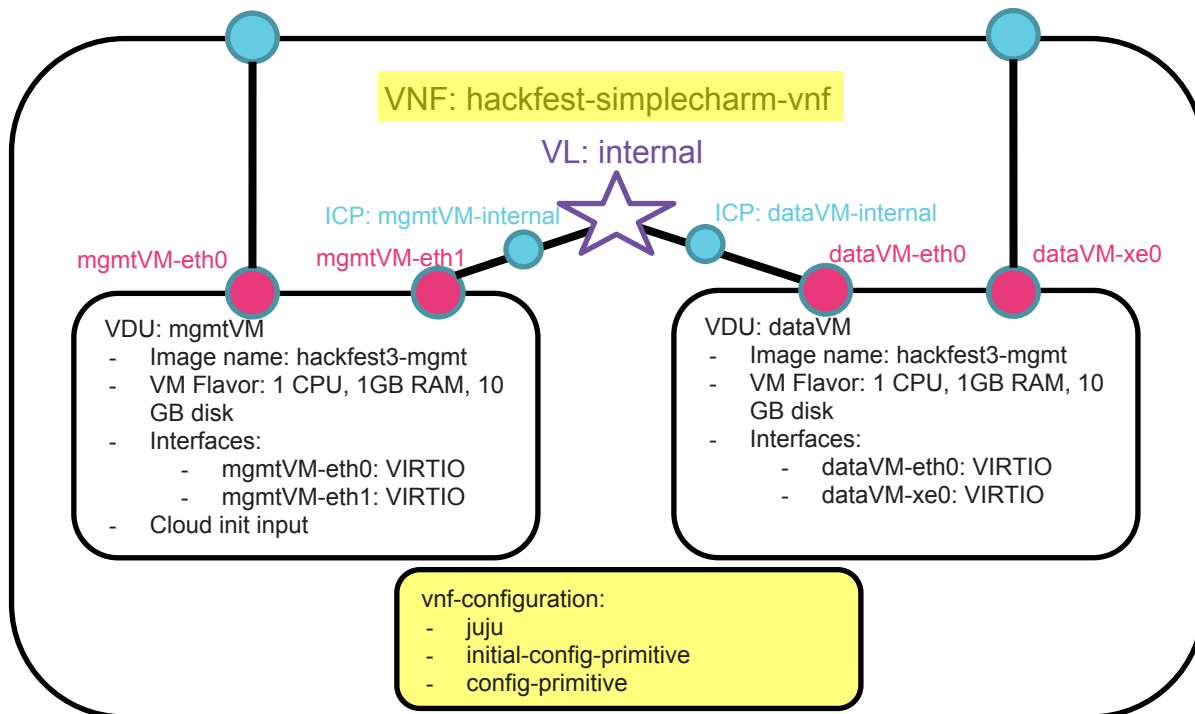
OSM Hackfest – Session 7b
Adding Charms to your VNF Descriptor
Eduardo Sousa (Whitestack)

VNF diagram

Changes with respect to 'hackfest_cloudinit-vnf'

External Connection point: vnf-mgmt

External Connection point: vnf-data



Generate the skeleton of the VNF Package and write the VNF descriptor

Copy VNF `hackfest_cloudinit-vnf` to `hackfest_simplecharm-vnf`

- Rename descriptor file
- Edit the descriptor
- Name: `hackfest-simplecharm-vnf`

Go into the VNF folder and write the VNF descriptor '`hackfest_simplecharm_vnfd.yaml`'.

Charms and Descriptors

Add the vnf-configuration section, as seen to the right, to the end of your descriptor, with the same level of indentation as the name of the VNF.

`initial-config-primitive` defines the primitives run at day-1, when the charm is instantiated.

`config-primitive` defines the primitives available to run as day-2 configuration.

```
name: 'myvnf'  
...  
vnf-configuration:  
  juju:  
    charm: simple  
  initial-config-primitive:  
  config-primitive:
```

Charms and Descriptors

Fill in the `initial-config-primitive` section. The `<rw_mgmt_ip>` token will be replaced with the IP address of your VM, allowing the charm to ssh to it.

```
initial-config-primitive:
-   seq: '1'
    name: config
    parameter:
      -   name: ssh-hostname
          value: <rw_mgmt_ip>
      -   name: ssh-username
          value: ubuntu
      -   name: ssh-password
          value: osm4u
-   seq: '2'
    name: touch
    parameter:
      -   name: filename
          value: '/home/ubuntu/first-touch'
```

Charms and Descriptors

Fill in the config-primitive section. This defines the primitive(s) available to run by the operator.

```
config-primitive:  
-   name: touch  
    parameter:  
    -   name: filename  
        data-type: STRING  
        default-value: '/home/ubuntu/touched'
```

Validate your VNF descriptor

Use the validation tool to check that the descriptor is syntactically correct:

```
devops/descriptor-packages/tools/validate_descriptor.py  
<VNF_DESCRIPTOR_FILE>
```

If an error message is shown, review the descriptor and validate again. Otherwise, the descriptor is syntactically correct.

Complete your VNF Package with the charm, the cloud-init file and the logo

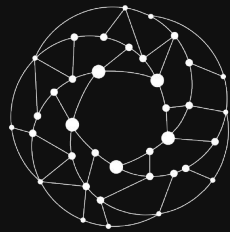
- Copy your compiled charm to descriptor folder (e.g. ~/hackfest_simplecharm_vnfd)
 - `cp -r ~/charms/builds/simple ~/hackfest_simplecharm_vnfd/charms`

Generate the VNF package and upload it to the UI

- Generate the VNF Package .tar.gz
 - `devops/descriptor-packages/tools/generate_descriptor_pkg.sh -t vnfd -N hackfest_simplecharm_vnfd`

Note: the argument -N is important if you want to keep the package folder and files after creating the package.

- Upload `hackfest_simplecharm_vnfd.tar.gz` to OSM UI

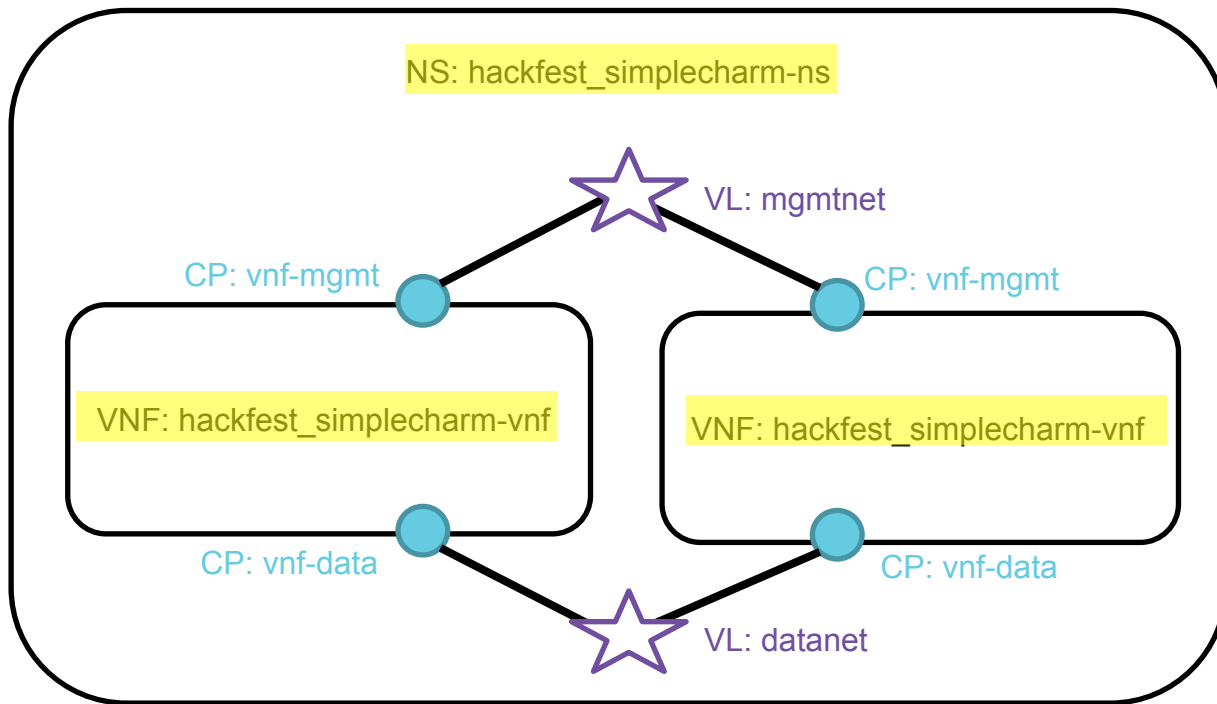


Open Source
MANO

Create NS, instantiate
and run config primitives
With subtitle

NS diagram

Changes highlighted in yellow



Deploying NS

- Go to UI > Packages > NS Packages
- Find hackfest-simplecharm-ns and click *Instantiate NS* Action
- Complete the form
 - Add a name to the NS
 - Select the Datacenter where the NS will be deployed
 - Add SSH key
- Go to UI > VNF Instances to see the instance.
- Click *Show Info* Action to see the mgmt IP address of the VNF
- Connect to each VNF:
 - `ssh ubuntu@<IP>`
- Check that the cloud-config file was executed;
 - The file `/home/ubuntu/first-touch` should exist

Testing VNF primitives

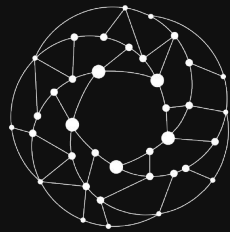
- Go to UI > NS instances > Actions: Exec primitive
 - Vnf-member-index
 - Action
 - Parameter name
 - Parameter value

Relevant links

- Juju
 - <https://jujucharms.com/>
- Charm Developers Guide
 - <https://jujucharms.com/docs/2.4/developer-getting-started>
- Creating a VNF Charm
 - [https://osm.etsi.org/wikipub/index.php/Creating_your_own_VNF_charm_\(Release_FOUR\)](https://osm.etsi.org/wikipub/index.php/Creating_your_own_VNF_charm_(Release_FOUR))
- Creating a VNF Package
 - [https://osm.etsi.org/wikipub/index.php/Creating_your_own_VNF_package_\(Release_FOUR\)](https://osm.etsi.org/wikipub/index.php/Creating_your_own_VNF_package_(Release_FOUR))
- Session 5 charm and descriptors
 - <https://github.com/AdamIsrael/osm-hackfest>

Example VNF Charms

- Using Ansible
 - <https://github.com/5GinFIRE/mano/tree/master/charms/ansible-charm>
- vpe-router, demoed at MWC 2016
 - <https://github.com/AdamIsrael/vpe-router>
- Hackfest examples
 - <https://github.com/AdamIsrael/osm-hackfest>



Open Source
MANO

The End