i2CAT OFELIA MANUAL

Introduction

Provides important information about the overall architecture of the OFELIA island at i2CAT, Barcelona, as well as a step-by-step guide for using the different resources of the OFELIA island. This document is mainly aimed at experienced experimenters who are familiar with the different concepts of an OpenFlow network and are comfortable with configuring and using command line (CLI) tools to operate the testbed.

The OFELIA testbed is a networking testbed where experimenters can perform experiments based on OpenFlow. The OFELIA testbed at i2CAT started with the European project OFELIA in September 2010. More details about this project can be found on http://www.fp7-ofelia.eu/ . The role of i2CAT is to offer an enhanced version of the OFELIA testbed to the Fed4FIRE project (http://fed4fire.eu/). Fed4FIRE is a federation of existing testbeds which may not be based on OpenFlow, in contrast with OFELIA.

Component	**Public IP address**
VTAM	https://137.222.204.27:5001/xmlrpc/sfa/ https://137.222.204.27:5001/xmlrpc/3/ (GENI v3)
OFAM	https://137.222.204.27:5005/xmlrpc/sfa/ https://137.222.204.27:5005/xmlrpc/3/ (GENI v3)

Feedback is welcome!

Please send your comments about this document (errors, misunderstandings, etc.) to carlos.bermudo@i2cat.net

Fed4FIRE First Level Support (FLS)

The basic status of the state of the aggregate managers of i2CAT OFELIA can be found at https://flsmonitor.fed4fire.eu/ . The rows with Testbed Name “Ofelia (i2CAT openflow)” and “Ofelia (i2CAT vtam)” refer to the aggregate managers of the OpenFlow networking and Compute resources respectively. The “GetVersion” field indicates whether the relevant aggregate manager can be contacted (green colour) or not (red colour). The “Free Resources” field indicates whether the Virtual Wall 2 certificate authority can be used to interact with the aggregate managers. If the authority is valid, the field will be green or orange otherwise it will be red. The numeric value of the “Free Resources” field does not provide any meaningful information in the case of i2CAT OFELIA since the resources at the testbed are virtualized and sliced for experimenters.

General Architecture

Figure 1 shows the general architecture of the i2CAT OFELIA island for Fed4FIRE from an experimenter point of view. In terms of OpenFlow networking resources, experimenters have at their disposal 5 OpenFlow-enabled switches that they can reserve and use for their experiments. The testbed also offers compute resources in terms of virtual machines that can be created on the three virtualization servers in the testbed. These virtual machines can be created and instantiated by the experimenters and act as the source and sink of traffic for the OpenFlow network.

A short description of the role of each component of the testbed is described in the following table Table 1 and more advanced description on how to use the different components will be given in the next chapters.

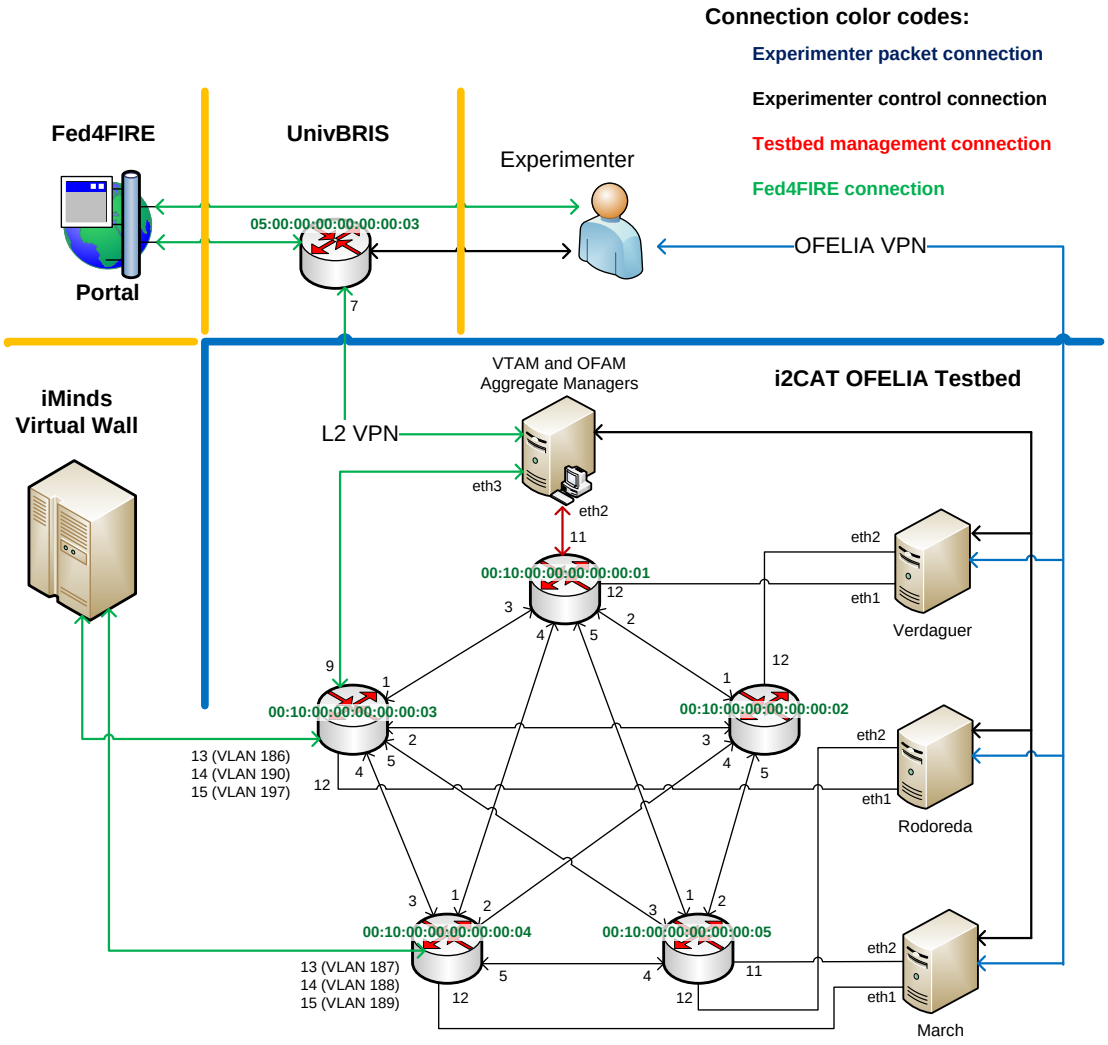

Figure 1: General architecture of i2CAT OFELIA for Fed4FIRE

Table 1: Description of each component inside the i2CAT OFELIA island

Component	Symbol	Description
OpenFlow Switch		OpenFlow packet switch (model NEC IP8800/S3640-24T2XW) on which selected OpenFlow ports can be reserved and used by experimenters.
Virtualization server		Virtualization servers (model SuperMicro SYS-6010T-T) on which experimenters create and instantiate their virtual machines.
VTAM and OFAM Aggregate managers		VTAM and OFAM aggregate managers are software components which are used to manage virtualization servers and network resources that are available for experimenters.
Fed4FIRE portal		Fed4FIRE portal allows experimenters to list and reserve resources on the testbed as well as other testbeds inside the Fed4FIRE consortium.

Pre-requisites

- OMNI installed with Fed4FIRE/PlanteLab/GENI GPO credentials. (ClearingHouse running)
- Optional: OFELIA registration and access to the VPN (https://alpha.fp7-ofelia.eu/cms/ofelia-facility-and-islands/how-to-experiment/) to access the VMs or use Bristol GateWay to access Bristol VMs (http://univbrisofeliaf4f.blogs.ilrt.org/user-manual-introduction/reservation/virtualization-aggregate-manager-vtam/logging-inside-virtual-machines/).

Provisioning

This section shows the provisioning process for the different types of resources on the testbed. Currently, there are two aggregate managers on the testbed: VTAM and OFAM. VTAM is used to control and manage the virtualization servers while OFAM is used for network resources.

Both the aggregate managers at i2CAT OFELIA speak GENI API v2 and in the near future, the aggregate managers will be compatible with GENI API v3. There are several Fed4FIRE-supported tools: OMNI (patched), SFI, jFed and Fed4FIRE portal, which can be used to list and reserve resources on the testbed. The table below Table 1 shows the compatibility/ability of these reservation tools with respect with the aggregate managers at i2CAT OFELIA.

Table 2: Status of the compatibility/ability of different reservation tools with i2CAT OFELIA

Provisioning tool	Tested on i2CAT	Available at	Type
OMNI	Fully tested & working	http://trac.gpolab.bbn.com/gcf/wiki/Omni	CLI
SFI	Fully tested & working	http://trac.myslice.info/wiki/InstallSfa	CLI
jFed	Partially integrated	http://jfed.iminds.be/	CLI/GUI
Portal	Partially integrated	https://portal.fed4fire.eu/	GUI

OMNI and SFI are basic CLI tools and therefore, they can work “directly” with the aggregate managers of the i2CAT OFELIA. jFed and the Fed4FIRE portal are mainly GUI tools and work by providing an intuitive GUI on top of the CLI tool which is integrated inside them. Hence, the reason why jFed and the portal were not originally fully compatible with OFELIA is the development of the GUI has not been finished for OFELIA. At this moment jFed Experimenter can now work with OFELIA by using the RSpecs Editor button, and it is expected that portal development should be finished in the near future.

The remaining of this website section will describe the use of OMNI to list and reserve resources on i2CAT OFELIA. It is assumed that the experimenter has configured the OMNI client appropriately; especially the clearinghouse/registry configuration after the experimenter has obtained the appropriate credential from one of the many Fed4FIRE authority providers.

Creation of a slice

A slice is a container of resources for an experiment.

sudo python `omni.py createslice sfatest`

The last argument is the name of the slice.

Virtualization Aggregate Manager (VTAM)

VTAM is the aggregate manager responsible for controlling and managing the virtualization servers at i2CAT OFELIA. VTAM can be contacted using its public internet address at: https://137.222.204.27:5001/xmlrpc/sfa/

Viewing available resources to reserve

In GENI API v2, it is possible to see which resources are available to book on i2CAT OFELIA by using the API call `listresources`. For example in OMNI, an experimenter can use the following command:

```
sudo python omni.py -a https://137.222.204.27:5001/xmlrpc/sfa/ -V2 listresources -o
```

The result of this API call is an XML file called RSpec which describes the resources available from this particular aggregate manager (the –o argument makes the output to be stored in an .xml file with a name similar to rspec-137-222-204-27-5001-xmlrpc-sfa.xml). More specifically, this RSpec is called an Ad RSpec or Advertisement RSpec. An example of such an RSpec from i2CAT OFELIA VTAM is given in the code examples section *Example of an Advertisement RSpec from VTAM*.

At lines 9, 49 and 89 in the example RSpec (VTAM_advertisement_RSPEC.rspec) in section *Example of an Advertisement RSpec from VTAM*, three “node” xml elements are present. They represent the three virtualization servers at i2CAT OFELIA: Rodoreda, March and Verdaguer which experimenters can use to host the virtual machines for their experiments.

This rspec also shows the links between the servers and the switches. For example, lines 38-42 show that interface eth1 of the VMs located in server Rodoreda are linked to switch 00:10:00:00:00:00:00:03 at port 12.

It should be noted that only Debian Squeeze (6.0) is currently supported as OS for the virtual machines (in full virtualization mode) that Fed4FIRE experimenters create in i2CAT OFELIA (lines 13-14).

Reserving resources (creating virtual machines)

The creation of a new virtual machine on i2CAT OFELIA is done through the GENI API v2 call `createsliver`. In OMNI, this is done by using the following command:

```
sudo python omni.py -a https://137.222.204.27:5001/xmlrpc/sfa/ -V2 createsliver sfatest vtam_request_i2cat.rspec
```

The before-last argument in the command above gives the name of the slice. The last argument in the command above is the file name of the reservation RSpec. A reservation RSpec provides all the details that an aggregate manager needs to create/reserve resources on

the aggregate manager. An example of a reservation RSpec (vtam_request_i2cat.rspec) for i2CAT OFELIA VTAM is in the code section *Example of a Request RSpec for VTAM (vtam_request_i2cat.rspec)*. Experimenters can modify and use this RSpec to reserve resources on the island.

The VMs created in i2CAT OFELIA testbed are behind OFELIA VPN and not reachable from the internet. There are two ways to reach them: through the OFELIA VPN, so you should register in the OFELIA project through this link (https://alpha.fp7-ofelia.eu/cms/ofelia-facility-and-islands/how-to-experiment/) or through a VirtualWall VM, which are already linked to the OFELIA VPN and can access the OFELIA VMs.

The most important information in the example RSpec is located on line 4. Line 4 is the start of the "node" element which is the equivalent of a virtual machine in i2CAT OFELIA. The "node" element has a "name" attribute in line 6 which is the name the experimenter wants to give to the virtual machine. It is important that the name of the virtual machine is alphanumeric and that no 2 virtual machines within the same sliver at i2CAT OFELIA have the same name.

Another important attribute of the node element is the "component_name" (and also "component_id") which specifies on which virtualization server the virtual machine is to be created (in the example, the server named Verdaguer). It should be noted that experimenters can retrieve this information from the advertisement RSpec as mentioned in the previous subsection. The last important attribute of the node element is "component_manager_id" which is the URN of the VTAM aggregate manager at i2CAT OFELIA. The other fields in the reservation RSpec should not be modified by the experimenter.

The following GENI API v2 call can be used to know at what creation stage is the virtual machine in:

```
sudo python omni.py -a https://137.222.204.27:5001/xmlrpc/sfa/ -V2 sliverstatus sfatest -o
```

Examples of the output provided by VTAM in OMNI are given in the code section *Example of a Request RSpec for OFAM (ofam_*request_i2cat.rspec)

```
1.  <rspec                          xmlns="https://github.com/fp7-
    ofelia/ocf/blob/ocf.rspecs/openflow/schemas/request.xsd"
2.          xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
3.
    xmlns:openflow="http://www.geni.net/resources/rspec/ext/openflo
    w/3"
4.
    xs:schemaLocation="http://www.geni.net/resources/rspec/3
5.              https://github.com/fp7-
    ofelia/ocf/blob/ocf.rspecs/openflow/schemas/request.xsd
6.              http://www.geni.net/resources/rspec/3/request.xsd
7.
    http://www.geni.net/resources/rspec/ext/openflow/3
```

```
8.
    http://www.geni.net/resources/rspec/ext/openflow/3/of-resv.xsd"
9.       type="request">
10.    <openflow:sliver  email="a@b.com"  description="OF  request
    example">
11.       <openflow:controller          url="tcp:10.216.12.29:6633"
    type="primary"/>
12.        <openflow:group name="fs1">
13.           <openflow:datapath
    component_id="urn:publicid:IDN+openflow:optin:i2cat.of_ocf+data
    path:00:10:00:00:00:00:00:01"
14.
    component_manager_id="urn:publicid:IDN+openflow:optin:i2cat.of_
    optin+authority+am"
15.                                dpid="00:10:00:00:00:00:00:01">
16.              <openflow:port name="GBE0/4" num="4"/>
17.              <openflow:port name="GBE0/12" num="12"/>
18.           </openflow:datapath>
19.           <openflow:datapath
    component_id="urn:publicid:IDN+openflow:optin:i2cat.of_ocf+data
    path:00:10:00:00:00:00:00:03"
20.
    component_manager_id="urn:publicid:IDN+openflow:optin:i2cat.of_
    optin+authority+am"
21.                                dpid="00:10:00:00:00:00:00:03">
22.              <openflow:port name="GBE0/3" num="4"/>
23.              <openflow:port name="GBE0/3" num="9"/>
24.              <openflow:port name="GBE0/13" num="13"/>
25.              <openflow:port name="GBE0/14" num="14"/>
26.              <openflow:port name="GBE0/15" num="15"/>
27.           </openflow:datapath>
28.           <openflow:datapath
    component_id="urn:publicid:IDN+openflow:optin:i2cat.of_ocf+data
    path:00:10:00:00:00:00:00:04"
29.
    component_manager_id="urn:publicid:IDN+openflow:optin:i2cat.of_
    optin+authority+am"
30.                                dpid="00:10:00:00:00:00:00:04">
31.              <openflow:port name="GBE0/1" num="1"/>
32.              <openflow:port name="GBE0/3" num="3"/>
33.              <openflow:port name="GBE0/13" num="13"/>
34.              <openflow:port name="GBE0/14" num="14"/>
35.              <openflow:port name="GBE0/15" num="15"/>
36.           </openflow:datapath>
37.        </openflow:group>
38.        <openflow:match>
39.           <openflow:use-group name="fs1" />
```

```
40.             <openflow:packet>
41.                <openflow:dl_vlan value="755" />
42.             </openflow:packet>
43.          </openflow:match>
44.      </openflow:sliver>
45.  </rspec>
```

Different states reported by VTAM during virtual machine creation. In first listing, the line 10 shows the name of the slice for which the "sliverstatus" is requested. Lines 4-7 show the state of the geni_resources (i.e. the virtual machines) within the slice at the aggregate manager. Line 5 indicates the name of the virtual machine being described. The different "geni_status" states that a virtual machine can be in are detailed below in Table 3. After the virtual machine is created through one of the Fed4FIRE tools, it will be in "running" state and can be logged into immediately. The "geni_status" in line 11 shows the status of the whole sliver at the aggregate manager.

Table 3: Different states that the virtual machines can be in on i2CAT OFELIA

Virtual Machine State	Description
creating…	The virtual machine is being created by the virtualization agent on the virtualization server. This involves copying and configuring a pre-configured image. Usually takes around 5 minutes to finish if there are no other virtual machine in the creation queue.
running	The virtual machine is now running and the experimenter can log inside his virtual machine.

After the virtual machines get into the "running" state, you can get the IP allocated to the virtual machines by using the following command:

```
sudo python omni.py -a https://137.222.204.27:5001/xmlrpc/sfa/ -V2 listresources sfatest
```

It is important to append the slice name to the `listresources` command so as to get all the virtual machines in the slice at the aggregate manager. *Example of a Manifest RSpec from VTAM* in the code section shows a sample of the manifest RSpecs which is returned by the aggregate manager following the command above. Lines 22-30 of the list give information about the virtual machine name and lines 11-21 give info about the virtualization server on which it is located. Within the node element, there is the "login" element (line 23) which provides the IP of the virtual machine. This is used to log into the virtual machine by SSH using the port number and username as mentioned in line 23. The next section provides details about how to log into the virtual machines which are on a private subnet.

Logging inside virtual machines

All the virtual machines created on i2CAT OFELIA have a private IP address (usually in the 10.216.12.0/24 subnet), which means that the virtual machines cannot be accessed from the internet directly. There are two ways to ssh into an i2CAT OFELIA VM, one is to be registered and connected to OFELIA VPN (more info at https://alpha.fp7-ofelia.eu/cms/ofelia-facility-and-islands/how-to-experiment/). The other way is to ssh directly to the VMs from a VirtualWall VM. VirtualWall is already connected to OFELIA VPN, so its VMs can ssh to OFELIA VMs.

Configuring virtual machines for experimentation

After an experimenter logs into his virtual machine, he needs to perform some tasks to configure the experiment network interface (experimenters can use information in the item (d) on "Reserving Packet OpenFlow resources" page to verify which interface of the VM is connected to which switch) of his virtual machine so that he can send traffic through his flowspace on the OpenFlow network. Since on i2CAT OFELIA experimenters are allocated on VLANs to separate their traffic from other traffic, experimenters need to create a VLAN interface on the experiment network interface ethX. This is done by using the following commands:

```
ifconfig ethX up
```

```
vconfig add eth1 VLAN_ALLOCATED
```

The experimenter needs to replace the VLAN_ALLOCATED with the VLAN which was assigned to them by i2CAT OFELIA OFAM.

The experimenter can then give the VLAN interface an IP by using the following command:

```
ifconfig ethX.VLAN_ALLOCATED IP/SUBNET up
```

It is desirable that experimenters do not give an IP in the 10. network because this may cause traffic to go to the control interface rather than the experiment interface.

An additional configuration for the experiment interface is to set the Maximum Transmission Unit (MTU) to 1496 bytes because of the network configuration in the testbed.

```
ifconfig ethX.VLAN_ALLOCATED mtu 1496
```

Software available on default virtual machine

Currently, the following software is available on the default Fed4FIRE virtual machine at i2CAT:

1. **NOX OpenFlow controller** at: "/opt/ofelia/software/nox".
2. **POX OpenFlow controller** is easily installable:
    a. apt-get update
    b. apt-get install git
    c. git clone https://github.com/noxrepo/pox.git
    d. cd pox/

e. git checkout dart ← Otherwise won't work
f. ./pox.py forwarding.l2_learning (example of controller acting as a L2 learning switch)

Deleting/renewing the virtual machines/sliver

Renewing the virtual machines/sliver

An experimenter can renew his virtual machines/sliver before expiration by using the command:

```
sudo python omni.py -a https://137.222.204.27:5001/xmlrpc/sfa/ -V2 renewslice -n sfatest 20141227T12:00:00Z
```

The format of the time is <year><day><month>T<hour>:<minute>:<second>Z

The maximum period to extend the life of a sliver is 30 days.

Deleting the virtual machines/sliver

The following commands can be used to delete all virtual machines in the sliver at the VTAM aggregate manager in OMNI:

```
sudo python omni.py -a https://137.222.204.27:5001/xmlrpc/sfa/ -V2 deletesliver sfatest
```

It is important to note that the deletion of all virtual machines inside a sliver is the only option offered by the GENI API v2 for unprovisioning. Therefore, if experimenters want to use the same virtual machine for a future experiment, they should not delete their sliver. Another important point is if the sliver expires, the virtual machines will be deleted at the aggregate manager but it is possible to renew the sliver to delay the deletion of the virtual machines.

OpenFlow Aggregate Manager (OFAM)

OFAM is the aggregate manager responsible for controlling and managing the OpenFlow switches. OFAM can be contacted using its public internet address at:

https://137.222.204.27:5005/xmlrpc/sfa/

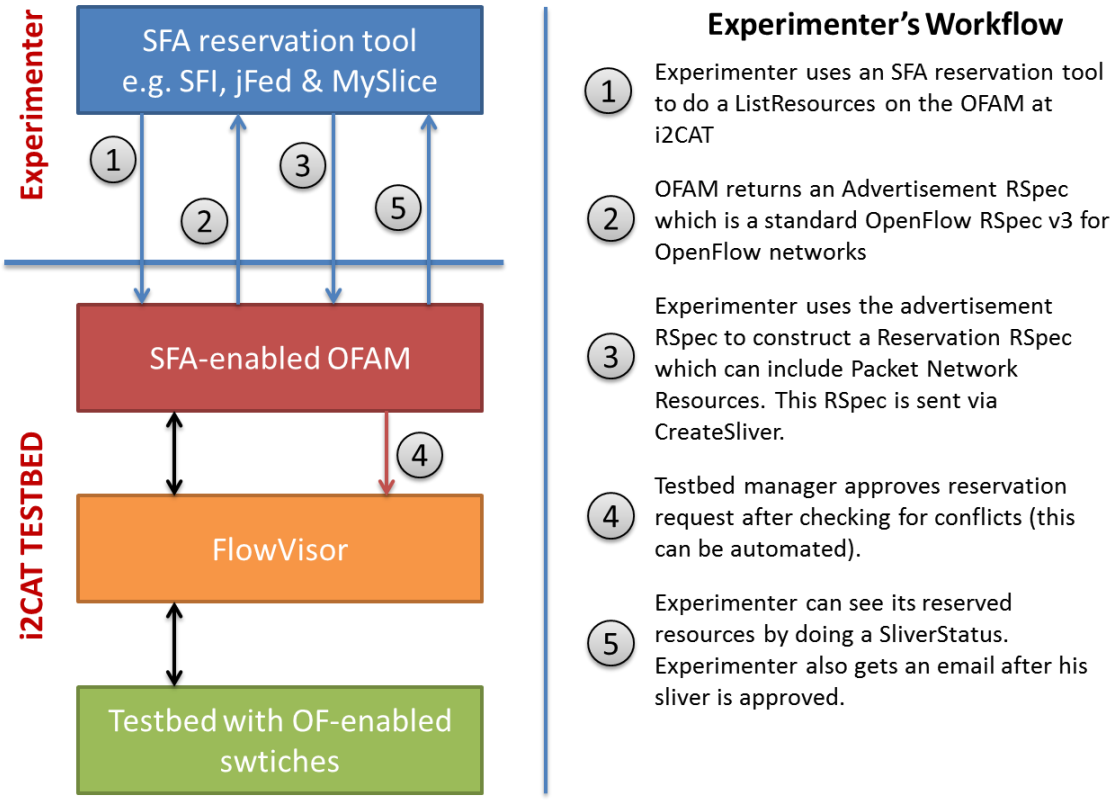

Figure 2: Overview of the reservation of network resources on i2CAT OFELIA testbed.

Viewing available OpenFlow resources to reserve

Similar to VTAM, the experimenter will use the following command to see which network resources are available to reserve:

```
sudo python omni.py -a https://137.222.204.27:5001/xmlrpc/sfa/ -V2
listresources
```

An example of the advertisement RSpec that can be retrieved is shown in the code section *Example of an Advertisement RSpec from OFAM* (OFAM_advertisement_RSPEC). The first child element in the RSpec is the "openflow:datapath" (line 8) which describes the switch DPID as well as the different OpenFlow ports which are available from that switch. These elements are quite important as they need to be specified in the reservation RSpec.

The links between the switches start to be defined at line 98 in the example advertisement RSpec with the element “openflow:link”. The value of the “component_id” attribute of this element gives an indication how the switches are connected. For example, the element <openflow:link srcDPID="00:10:00:00:00:00:00:03" srcPort="2" dstDPID="00:10:00:00:00:00:00:02" dstPort="3"/> means that the switch with DPID 00:10:00:00:00:00:00:03 is connected at port 2 to the switch with DPID 00:10:00:00:00:00:00:02 at port 3. It should be noted that all the links defined in the advertisement RSpec are bi-directional.

The links between switches and servers are not presented in this rspec, they must be consulted in the VTAM advertisement rspec.

Reserving Packet OpenFlow resources

The reservation of a flowspace on i2CAT OFELIA is done through the GENI API v2 call `createsliver`.

In OMNI, this is done by using the following command:

```
sudo python omni.py -a https://137.222.204.27:5005/xmlrpc/sfa/ -V2 createsliver sfatest ofam_request_i2cat.rspec
```

The before-last argument in the command above gives the name of the slice. The last argument in the command above is the file name of the reservation RSpec. A reservation RSpec provides all the details that an aggregate manager needs to create/reserve resources on the aggregate manager.

Once the flowspace has been requested, it has to be approved by the testbed manager.

An example of a packet-only RSpec for i2CAT OFELIA is included in the code section *Example of a Request RSpec for OFAM (ofam_request_i2cat.rspec)*. It should be noted that this is an example only and experimenters are encouraged to use the `listresources` API call to verify that the switches and links are still the same.

There are several important parameters that need to be included in the reservation RSpec:

a) **Controller IP (line 11):** It is important to specify a controller which is reachable from within the i2CAT OFELIA, e.g. 10.216.12.x IP that is obtained when an experimenter creates a virtual machine on the testbed. This is because of the firewall around the i2CAT OFELIA makes it impossible for FlowVisor to receive any messages from controllers which are found outside the OFELIA private network (10.216. network).
b) **VLAN Specification (line 41):** A VLAN should be specified in the “openflow:dl_vlan” attribute of the “openflow:match” element of the reservation RSpec. It is very likely that the testbed manager will have to allocate another VLAN than the one requested since the VLAN requested could have already been allocated. The flowspace finally allocated to the experimenter can be checked by the experimenter by using the following command once the sliver has been created:

```
sudo python omni.py -a https://137.222.204.27:5005/xmlrpc/sfa/ -V2
sliverstatus sfatest
```

where `sfatest` is the name of his slice.

It should be noted that the result of the command shows the status of all the resources of the AM, that is, the port numbers of the switches and the flowspace reserved, including the VLAN, as can be seen in line 87 of the sample file of section *Different states reported by OFAM during request*.

c) **Compute ports:** It is important to include the OpenFlow ports of the switches which are connected to the virtualization servers into the reservation RSpec. Without which, the traffic coming from the experimenters' virtual machines will not be included in the flowspace of the experimenter and can't be controlled by the controller of the experimenter.

It is easy to know what are these OpenFlow ports by doing a `listresources` on VT AM (`listresources` on OF AM only shows links between switches)

`sudo python omni.py` -a https://137.222.204.27:5001/xmlrpc/sfa/ -V2 listresources

and identifying the relevant "NetworkInterface" elements.

For example in the VT AM advertisement RSpec in the *Example of an Advertisement RSpec from VTAM*, the lines 38 to 42 show that the virtualization server Rodoreda is connected through Eth interface eth1 to the switch with DPID 00:10:00:00:00:00:00:03 on port 12. An experimenter should only send to the i2CAT OFELIA OFAM aggregate manager the network end point of the link and not the end point representing the server/port end of the link. There is no need to reserve the interface of the server since this is shared by default among all experimenters when they have a virtual machine on the server.

d) **Federation ports (inter-island links):** the i2CAT OFELIA testbed is connected to other testbeds which can be part of either Fed4FIRE or other projects such as virtualWall. Similar to the process of identifying compute ports, an experimenter can identify the federation links by looking at the "openflow:link". For example in the Ad Rspecs of OFAM in section *Example of an Advertisement RSpec from OFAM*, line 143 shows a federation link between i2CAT OFELIA (switch with DPID 00:10:00:00:00:00:00:03 with port 9) and UNIVBRIS OFELIA (switch with DPID 05:00:00:00:00:00:00:03 with port 7). The owner of one end-point of the link can be identified by looking at the child elements of the "openflow:link" tag of the link, they specify the component manager id of each link. An experimenter should only send to the i2CAT OFELIA OFAM aggregate manager the end point in the i2CAT OFELIA island and separately reserve the other end point in the other island by using the appropriate different aggregate manager.

Approval of FlowSpace

Experimenters can check the approval status of their flowspace by using the following command:

```
sudo python omni.py -a https://137.222.204.27:5005/xmlrpc/sfa/ -V2
sliverstatus sfatest
```

where `sfatest` is the name of his slice.

Deleting/Renewing the allocated flowspace

Renewing the allocated flowspace before expiration

An experimenter can renew his flowspace before expiration by using the command:

```
sudo python omni.py -a https://137.222.204.27:5005/sfa/2/ -V2
renewsliver -n sfatest 20141007T12:00:00Z
```

The format of the time is <year><day><month>T<hour>:<minute>:<second>Z

Deleting the allocated flowspace

An experimenter can delete his flowspace by using the command:

```
sudo python omni.py -a https://137.222.204.27:5005/sfa/2/ -V2
deletesliver sfatest
```

The deletion of the flowspace will happen immediately.

Code examples

1- Example of an Advertisement RSpec from VTAM

```
<?xml version="1.0"?>
  <!-- Resources at AM:
      URN: unspecified_AM_URN
      URL: https://137.222.204.27:5001/xmlrpc/sfa/
 -->

<rspec xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" type="advertisement"
xsi:schemaLocation="http://www.protogeni.net/resources/rspec/2 https://github.com/fp7-
ofelia/ocf/blob/ocf.rspecs/server_schema.xsd http://www.protogeni.net/resources/rspec/2/ad.xsd" expires="2014-11-
20T17:33:10Z" generated="2014-11-20T16:33:10Z">
  <network name=".i2cat">
    <node component_manager_id="urn:publicid:IDN+i2cat:vtam+cm" component_id="urn:publicid:IDN+i2cat:vtam+node+Rodoreda"
exclusive="false" component_name="urn:publicid:IDN+i2cat:vtam+node+Rodoreda">
          <hostname>rodoreda.ctx.i2cat.net</hostname>
          <name>Rodoreda</name>
          <operating_system_type>GNU/Linux</operating_system_type>
          <operating_system_distribution>Debian</operating_system_distribution>
          <operating_system_version>6.0</operating_system_version>
          <virtualization_technology>xen</virtualization_technology>
          <cpus_number>None</cpus_number>
          <cpu_frequency>None</cpu_frequency>
          <memory>None</memory>
          <hdd_space_GB>None</hdd_space_GB>
          <agent_url>https://rodoreda.ctx.i2cat.net:9229/</agent_url>
          <location country="Spain" longitude="2.11191386334" latitude="41.3873269495"/>
          <service type="Range">
            <type>IP_Range</type>
            <name>i2cat</name>
            <start_value>10.216.12.25</start_value>
            <end_value>10.216.15.254</end_value>
```

```
27.            </service>
28.            <service type="Range">
29.              <type>MAC_Range</type>
30.              <name>i2cat</name>
31.              <start_value>02:03:00:00:00:00</start_value>
32.              <end_value>02:03:FF:FF:FF:FF</end_value>
33.            </service>
34.            <service type="NetworkInterface">
35.              <from_server_interface_name>eth1.999</from_server_interface_name>
36.              <to_network_interface_port>None</to_network_interface_port>
37.            </service>
38.            <service type="NetworkInterface">
39.              <from_server_interface_name>eth1</from_server_interface_name>
40.              <to_network_interface_id>00:10:00:00:00:00:00:03</to_network_interface_id>
41.              <to_network_interface_port>12</to_network_interface_port>
42.            </service>
43.            <service type="NetworkInterface">
44.              <from_server_interface_name>eth2</from_server_interface_name>
45.              <to_network_interface_id>00:10:00:00:00:00:00:05</to_network_interface_id>
46.              <to_network_interface_port>12</to_network_interface_port>
47.            </service>
48.          </node>
49.          <node component_manager_id="urn:publicid:IDN+i2cat:vtam+cm"
   component_id="urn:publicid:IDN+i2cat:vtam+node+March" exclusive="false"
   component_name="urn:publicid:IDN+i2cat:vtam+node+March">
50.            <hostname>march.ctx.i2cat.net</hostname>
51.            <name>March</name>
52.            <operating_system_type>GNU/Linux</operating_system_type>
53.            <operating_system_distribution>Debian</operating_system_distribution>
54.            <operating_system_version>6.0</operating_system_version>
55.            <virtualization_technology>xen</virtualization_technology>
56.            <cpus_number>None</cpus_number>
57.            <cpu_frequency>None</cpu_frequency>
58.            <memory>None</memory>
59.            <hdd_space_GB>None</hdd_space_GB>
```

```
60.            <agent_url>https://10.216.140.3:9229/</agent_url>
61.            <location country="Spain" longitude="2.11191386334" latitude="41.3873269495"/>
62.            <service type="Range">
63.              <type>IP_Range</type>
64.              <name>i2cat</name>
65.              <start_value>10.216.12.25</start_value>
66.              <end_value>10.216.15.254</end_value>
67.            </service>
68.            <service type="Range">
69.              <type>MAC_Range</type>
70.              <name>i2cat</name>
71.              <start_value>02:03:00:00:00:00</start_value>
72.              <end_value>02:03:FF:FF:FF:FF</end_value>
73.            </service>
74.            <service type="NetworkInterface">
75.              <from_server_interface_name>eth1.999</from_server_interface_name>
76.              <to_network_interface_port>None</to_network_interface_port>
77.            </service>
78.            <service type="NetworkInterface">
79.              <from_server_interface_name>eth1</from_server_interface_name>
80.              <to_network_interface_id>00:10:00:00:00:00:00:04</to_network_interface_id>
81.              <to_network_interface_port>12</to_network_interface_port>
82.            </service>
83.            <service type="NetworkInterface">
84.              <from_server_interface_name>eth2</from_server_interface_name>
85.              <to_network_interface_id>00:10:00:00:00:00:00:05</to_network_interface_id>
86.              <to_network_interface_port>11</to_network_interface_port>
87.            </service>
88.          </node>
89.          <node component_manager_id="urn:publicid:IDN+i2cat:vtam+cm"
   component_id="urn:publicid:IDN+i2cat:vtam+node+Verdaguer" exclusive="false"
   component_name="urn:publicid:IDN+i2cat:vtam+node+Verdaguer">
90.            <hostname>verdaguer.ctx.i2cat.net</hostname>
91.            <name>Verdaguer</name>
92.            <operating_system_type>GNU/Linux</operating_system_type>
```

```
93.           <operating_system_distribution>Debian</operating_system_distribution>
94.           <operating_system_version>6.0</operating_system_version>
95.           <virtualization_technology>xen</virtualization_technology>
96.           <cpus_number>None</cpus_number>
97.           <cpu_frequency>None</cpu_frequency>
98.           <memory>None</memory>
99.           <hdd_space_GB>None</hdd_space_GB>
100.          <agent_url>https://verdaguer.ctx.i2cat.net:9229/</agent_url>
101.          <location country="Spain" longitude="2.11191386334" latitude="41.3873269495"/>
102.          <service type="Range">
103.            <type>IP_Range</type>
104.            <name>i2cat</name>
105.            <start_value>10.216.12.25</start_value>
106.            <end_value>10.216.15.254</end_value>
107.          </service>
108.          <service type="Range">
109.            <type>MAC_Range</type>
110.            <name>i2cat</name>
111.            <start_value>02:03:00:00:00:00</start_value>
112.            <end_value>02:03:FF:FF:FF:FF</end_value>
113.          </service>
114.          <service type="NetworkInterface">
115.            <from_server_interface_name>eth1.999</from_server_interface_name>
116.            <to_network_interface_port>None</to_network_interface_port>
117.          </service>
118.          <service type="NetworkInterface">
119.            <from_server_interface_name>eth1</from_server_interface_name>
120.            <to_network_interface_id>00:10:00:00:00:00:00:01</to_network_interface_id>
121.            <to_network_interface_port>12</to_network_interface_port>
122.          </service>
123.          <service type="NetworkInterface">
124.            <from_server_interface_name>eth2</from_server_interface_name>
125.            <to_network_interface_id>00:10:00:00:00:00:00:02</to_network_interface_id>
126.            <to_network_interface_port>12</to_network_interface_port>
127.          </service>
```

```
        </node>
      </network>
    </rspec>
```

2- Example of an Advertisement RSpec from OFAM

```
<?xml version="1.0"?>
  <!-- Resources at AM:
      URN: unspecified_AM_URN
      URL: https://137.222.204.27:5005/xmlrpc/sfa/
 -->

<rspec xmlns="https://github.com/fp7-ofelia/ocf/tree/ocf.rspecs/openflow/schemas"
  xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xmlns:openflow="https://github.com/fp7-
  ofelia/ocf/blob/ocf.rspecs/openflow/schemas" type="advertisement" xs:schemaLocation="https://github.com/fp7-
  ofelia/ocf/blob/ocf.rspecs/openflow/schemas/ https://github.com/fp7-ofelia/ocf/blob/ocf.rspecs/openflow/schemas/ad.xsd
  http://www.geni.net/resources/rspec/3/ad.xsd https://github.com/fp7-ofelia/ocf/blob/ocf.rspecs/openflow/schemas/ad.xsd
  http://www.geni.net/resources/rspec/ext/openflow/3/of-ad.xsd https://github.com/fp7-
  ofelia/ocf/blob/ocf.rspecs/openflow/schemas/network_schema.xsd" expires="2014-11-21T15:27:50Z" generated="2014-11-
  21T14:27:50Z">
  <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:05"
  component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:05">
    <openflow:port num="1" name="GBE0/1"/>
        <openflow:port num="2" name="GBE0/2"/>
        <openflow:port num="3" name="GBE0/3"/>
        <openflow:port num="4" name="GBE0/4"/>
        <openflow:port num="5" name="GBE0/5"/>
        <openflow:port num="6" name="GBE0/6"/>
        <openflow:port num="7" name="GBE0/7"/>
        <openflow:port num="8" name="GBE0/8"/>
        <openflow:port num="9" name="GBE0/9"/>
        <openflow:port num="10" name="GBE0/10"/>
        <openflow:port num="11" name="GBE0/11"/>
        <openflow:port num="12" name="GBE0/12"/>
        <openflow:port num="13" name="GBE0/13"/>
```

```
22.          <openflow:port num="14" name="GBE0/14"/>
23.          <openflow:port num="15" name="GBE0/15"/>
24.          <openflow:port num="16" name="GBE0/16"/>
25.        </openflow:datapath>
26.        <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:01"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:01">
27.          <openflow:port num="1" name="GBE0/1"/>
28.          <openflow:port num="2" name="GBE0/2"/>
29.          <openflow:port num="3" name="GBE0/3"/>
30.          <openflow:port num="4" name="GBE0/4"/>
31.          <openflow:port num="5" name="GBE0/5"/>
32.          <openflow:port num="6" name="GBE0/6"/>
33.          <openflow:port num="7" name="GBE0/7"/>
34.          <openflow:port num="8" name="GBE0/8"/>
35.          <openflow:port num="9" name="GBE0/9"/>
36.          <openflow:port num="10" name="GBE0/10"/>
37.          <openflow:port num="11" name="GBE0/11"/>
38.          <openflow:port num="12" name="GBE0/12"/>
39.          <openflow:port num="13" name="GBE0/13"/>
40.          <openflow:port num="14" name="GBE0/14"/>
41.          <openflow:port num="15" name="GBE0/15"/>
42.          <openflow:port num="16" name="GBE0/16"/>
43.        </openflow:datapath>
44.        <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:04">
45.          <openflow:port num="1" name="GBE0/1"/>
46.          <openflow:port num="2" name="GBE0/2"/>
47.          <openflow:port num="3" name="GBE0/3"/>
48.          <openflow:port num="4" name="GBE0/4"/>
49.          <openflow:port num="5" name="GBE0/5"/>
50.          <openflow:port num="6" name="GBE0/6"/>
51.          <openflow:port num="7" name="GBE0/7"/>
52.          <openflow:port num="8" name="GBE0/8"/>
53.          <openflow:port num="9" name="GBE0/9"/>
54.          <openflow:port num="10" name="GBE0/10"/>
```

```
55.          <openflow:port num="11" name="GBE0/11"/>
56.          <openflow:port num="12" name="GBE0/12"/>
57.          <openflow:port num="13" name="GBE0/13"/>
58.          <openflow:port num="14" name="GBE0/14"/>
59.          <openflow:port num="15" name="GBE0/15"/>
60.          <openflow:port num="16" name="GBE0/16"/>
61.        </openflow:datapath>
62.        <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03">
63.          <openflow:port num="1" name="GBE0/1"/>
64.          <openflow:port num="2" name="GBE0/2"/>
65.          <openflow:port num="3" name="GBE0/3"/>
66.          <openflow:port num="4" name="GBE0/4"/>
67.          <openflow:port num="5" name="GBE0/5"/>
68.          <openflow:port num="6" name="GBE0/6"/>
69.          <openflow:port num="7" name="GBE0/7"/>
70.          <openflow:port num="8" name="GBE0/8"/>
71.          <openflow:port num="9" name="GBE0/9"/>
72.          <openflow:port num="10" name="GBE0/10"/>
73.          <openflow:port num="11" name="GBE0/11"/>
74.          <openflow:port num="12" name="GBE0/12"/>
75.          <openflow:port num="13" name="GBE0/13"/>
76.          <openflow:port num="14" name="GBE0/14"/>
77.          <openflow:port num="15" name="GBE0/15"/>
78.          <openflow:port num="16" name="GBE0/16"/>
79.        </openflow:datapath>
80.        <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:02"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:02">
81.          <openflow:port num="1" name="GBE0/1"/>
82.          <openflow:port num="2" name="GBE0/2"/>
83.          <openflow:port num="3" name="GBE0/3"/>
84.          <openflow:port num="4" name="GBE0/4"/>
85.          <openflow:port num="5" name="GBE0/5"/>
86.          <openflow:port num="6" name="GBE0/6"/>
87.          <openflow:port num="7" name="GBE0/7"/>
```

```
88.          <openflow:port num="8" name="GBE0/8"/>
89.          <openflow:port num="9" name="GBE0/9"/>
90.          <openflow:port num="10" name="GBE0/10"/>
91.          <openflow:port num="11" name="GBE0/11"/>
92.          <openflow:port num="12" name="GBE0/12"/>
93.          <openflow:port num="13" name="GBE0/13"/>
94.          <openflow:port num="14" name="GBE0/14"/>
95.          <openflow:port num="15" name="GBE0/15"/>
96.          <openflow:port num="16" name="GBE0/16"/>
97.        </openflow:datapath>
98.        <openflow:link srcDPID="00:10:00:00:00:00:00:03" srcPort="2" dstDPID="00:10:00:00:00:00:00:02" dstPort="3"/>
99.        <openflow:link srcDPID="00:10:00:00:00:00:00:02" srcPort="5" dstDPID="00:10:00:00:00:00:00:05" dstPort="2"/>
100.       <openflow:link srcDPID="00:10:00:00:00:00:00:03" srcPort="4" dstDPID="00:10:00:00:00:00:00:04" dstPort="3"/>
101.       <openflow:link srcDPID="00:10:00:00:00:00:00:03" srcPort="1" dstDPID="00:10:00:00:00:00:00:01" dstPort="3"/>
102.       <openflow:link srcDPID="05:00:00:00:00:00:00:03" srcPort="7" dstDPID="00:10:00:00:00:00:00:03" dstPort="9"/>
103.       <openflow:link srcDPID="00:10:00:00:00:00:00:01" srcPort="4" dstDPID="00:10:00:00:00:00:00:04" dstPort="1"/>
104.       <openflow:link srcDPID="00:10:00:00:00:00:00:04" srcPort="2" dstDPID="00:10:00:00:00:00:00:02" dstPort="4"/>
105.       <openflow:link srcDPID="00:10:00:00:00:00:00:04" srcPort="1" dstDPID="00:10:00:00:00:00:00:01" dstPort="4"/>
106.       <openflow:link srcDPID="00:10:00:00:00:00:00:01" srcPort="3" dstDPID="00:10:00:00:00:00:00:03" dstPort="1"/>
107.       <openflow:link srcDPID="00:10:00:00:00:00:00:04" srcPort="3" dstDPID="00:10:00:00:00:00:00:03" dstPort="4"/>
108.       <openflow:link srcDPID="00:10:00:00:00:00:00:04" srcPort="5" dstDPID="00:10:00:00:00:00:00:05" dstPort="4"/>
109.       <openflow:link srcDPID="00:10:00:00:00:00:00:03" srcPort="5" dstDPID="00:10:00:00:00:00:00:05" dstPort="3"/>
110.       <openflow:link srcDPID="00:10:00:00:00:00:00:05" srcPort="1" dstDPID="00:10:00:00:00:00:00:01" dstPort="5"/>
111.       <openflow:link srcDPID="00:10:00:00:00:00:00:01" srcPort="2" dstDPID="00:10:00:00:00:00:00:02" dstPort="1"/>
112.       <openflow:link srcDPID="00:10:00:00:00:00:00:05" srcPort="3" dstDPID="00:10:00:00:00:00:00:03" dstPort="5"/>
113.       <openflow:link srcDPID="00:10:00:00:00:00:00:02" srcPort="4" dstDPID="00:10:00:00:00:00:00:04" dstPort="2"/>
114.       <openflow:link srcDPID="00:10:00:00:00:00:00:05" srcPort="2" dstDPID="00:10:00:00:00:00:00:02" dstPort="5"/>
115.       <openflow:link srcDPID="00:10:00:00:00:00:00:05" srcPort="4" dstDPID="00:10:00:00:00:00:00:04" dstPort="5"/>
116.       <openflow:link srcDPID="00:10:00:00:00:00:00:02" srcPort="1" dstDPID="00:10:00:00:00:00:00:01" dstPort="2"/>
117.       <openflow:link srcDPID="00:10:00:00:00:00:00:02" srcPort="3" dstDPID="00:10:00:00:00:00:00:03" dstPort="2"/>
118.       <openflow:link srcDPID="00:10:00:00:00:00:00:01" srcPort="5" dstDPID="00:10:00:00:00:00:00:05" dstPort="1"/>
119.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:03_2_00:10:00:00:00:00:00:02_3">
120.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03"/>
```

```
121.         <openflow:port port_num="2"/>
122.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:02"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:02"/>
123.         <openflow:port port_num="3"/>
124.       </openflow:link>
125.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:02_5_00:10:00:00:00:00:00:05_2">
126.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:02"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:02"/>
127.         <openflow:port port_num="5"/>
128.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:05"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:05"/>
129.         <openflow:port port_num="2"/>
130.       </openflow:link>
131.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:03_4_00:10:00:00:00:00:00:04_3">
132.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03"/>
133.         <openflow:port port_num="4"/>
134.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:04"/>
135.         <openflow:port port_num="3"/>
136.       </openflow:link>
137.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:03_1_00:10:00:00:00:00:00:01_3">
138.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03"/>
139.         <openflow:port port_num="1"/>
140.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:01"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:01"/>
141.         <openflow:port port_num="3"/>
142.       </openflow:link>
143.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+05:00:00:00:00:00:00:03_7_00:10:00:00:00:00:00:03_9">
```

```
144.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+05:00:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="05:00:00:00:00:00:00:03"/>
145.         <openflow:port port_num="7"/>
146.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03"/>
147.         <openflow:port port_num="9"/>
148.       </openflow:link>
149.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:01_4_00:10:00:00:00:00:00:04_1">
150.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:01"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:01"/>
151.         <openflow:port port_num="4"/>
152.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:04"/>
153.         <openflow:port port_num="1"/>
154.       </openflow:link>
155.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:04_2_00:10:00:00:00:00:00:02_4">
156.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:04"/>
157.         <openflow:port port_num="2"/>
158.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:02"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:02"/>
159.         <openflow:port port_num="4"/>
160.       </openflow:link>
161.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:04_1_00:10:00:00:00:00:00:01_4">
162.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:04"/>
163.         <openflow:port port_num="1"/>
164.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:01"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:01"/>
165.         <openflow:port port_num="4"/>
166.       </openflow:link>
```

```
167.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:01_3_00:10:00:00:00:00:00:03_1">
168.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:01"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:01"/>
169.         <openflow:port port_num="3"/>
170.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03"/>
171.         <openflow:port port_num="1"/>
172.       </openflow:link>
173.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:04_3_00:10:00:00:00:00:00:03_4">
174.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:04"/>
175.         <openflow:port port_num="3"/>
176.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03"/>
177.         <openflow:port port_num="4"/>
178.       </openflow:link>
179.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:04_5_00:10:00:00:00:00:00:05_4">
180.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:04"/>
181.         <openflow:port port_num="5"/>
182.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:05"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:05"/>
183.         <openflow:port port_num="4"/>
184.       </openflow:link>
185.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:03_5_00:10:00:00:00:00:00:05_3">
186.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03"/>
187.         <openflow:port port_num="5"/>
188.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:05"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:05"/>
189.         <openflow:port port_num="3"/>
```

```
190.       </openflow:link>
191.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:05_1_00:10:00:00:00:00:00:01_5">
192.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:05"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:05"/>
193.         <openflow:port port_num="1"/>
194.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:01"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:01"/>
195.         <openflow:port port_num="5"/>
196.       </openflow:link>
197.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:01_2_00:10:00:00:00:00:00:02_1">
198.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:01"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:01"/>
199.         <openflow:port port_num="2"/>
200.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:02"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:02"/>
201.         <openflow:port port_num="1"/>
202.       </openflow:link>
203.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:05_3_00:10:00:00:00:00:00:03_5">
204.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:05"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:05"/>
205.         <openflow:port port_num="3"/>
206.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03"/>
207.         <openflow:port port_num="5"/>
208.       </openflow:link>
209.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:02_4_00:10:00:00:00:00:00:04_2">
210.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:02"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:02"/>
211.         <openflow:port port_num="4"/>
212.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:04"/>
```

```
213.         <openflow:port port_num="2"/>
214.       </openflow:link>
215.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:05_2_00:10:00:00:00:00:00:02_5">
216.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:05"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:05"/>
217.         <openflow:port port_num="2"/>
218.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:02"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:02"/>
219.         <openflow:port port_num="5"/>
220.       </openflow:link>
221.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:05_4_00:10:00:00:00:00:00:04_5">
222.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:05"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:05"/>
223.         <openflow:port port_num="4"/>
224.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:04"/>
225.         <openflow:port port_num="5"/>
226.       </openflow:link>
227.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:02_1_00:10:00:00:00:00:00:01_2">
228.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:02"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:02"/>
229.         <openflow:port port_num="1"/>
230.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:01"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:01"/>
231.         <openflow:port port_num="2"/>
232.       </openflow:link>
233.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:02_3_00:10:00:00:00:00:00:03_2">
234.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:02"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:02"/>
235.         <openflow:port port_num="3"/>
```

```
236.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:03"/>
237.         <openflow:port port_num="2"/>
238.       </openflow:link>
239.       <openflow:link
   component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:01_5_00:10:00:00:00:00:00:05_1">
240.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:01"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:01"/>
241.         <openflow:port port_num="5"/>
242.         <openflow:datapath component_id="urn:publicid:IDN+openflow:i2cat.ofam+datapath+00:10:00:00:00:00:00:05"
   component_manager_id="urn:publicid:IDN+openflow:i2cat.ofam+cm" dpid="00:10:00:00:00:00:00:05"/>
243.         <openflow:port port_num="1"/>
244.       </openflow:link>
245.       <openflow:link srcDPID="00:10:00:00:00:00:00:03" srcPort="10" dstDevice="VirtualWall-GW-A" dstPort=""/>
246.       <openflow:link srcDPID="00:10:00:00:00:00:00:03" srcPort="11" dstDevice="VirtualWall-GW-B" dstPort=""/>
247.       <openflow:link srcDPID="00:10:00:00:00:00:00:04" srcPort="10" dstDevice="VirtualWall-GW-C" dstPort=""/>
248.       <openflow:link srcDPID="00:10:00:00:00:00:00:04" srcPort="11" dstDevice="VirtualWall-GW-D" dstPort=""/>
249.       <openflow:link component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:03_10_VirtualWall-GW-
   A_">
250.         <openflow:device component_id="urn:publicid:IDN+federation:i2cat.ofam+device+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+federation:i2cat.ofam+cm"/>
251.         <openflow:port port_num="10"/>
252.         <openflow:device component_id="urn:publicid:IDN+federation:i2cat.ofam+device+VirtualWall-GW-A"
   component_manager_id="urn:publicid:IDN+federation:i2cat.ofam+cm"/>
253.         <openflow:port port_num=""/>
254.       </openflow:link>
255.       <openflow:link component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:03_11_VirtualWall-GW-
   B_">
256.         <openflow:device component_id="urn:publicid:IDN+federation:i2cat.ofam+device+00:10:00:00:00:00:00:03"
   component_manager_id="urn:publicid:IDN+federation:i2cat.ofam+cm"/>
257.         <openflow:port port_num="11"/>
258.         <openflow:device component_id="urn:publicid:IDN+federation:i2cat.ofam+device+VirtualWall-GW-B"
   component_manager_id="urn:publicid:IDN+federation:i2cat.ofam+cm"/>
259.         <openflow:port port_num=""/>
260.       </openflow:link>
```

```
261.       <openflow:link component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:04_10_VirtualWall-GW-
   C_">
262.         <openflow:device component_id="urn:publicid:IDN+federation:i2cat.ofam+device+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+federation:i2cat.ofam+cm"/>
263.         <openflow:port port_num="10"/>
264.         <openflow:device component_id="urn:publicid:IDN+federation:i2cat.ofam+device+VirtualWall-GW-C"
   component_manager_id="urn:publicid:IDN+federation:i2cat.ofam+cm"/>
265.         <openflow:port port_num=""/>
266.       </openflow:link>
267.       <openflow:link component_id="urn:publicid:IDN+openflow:i2cat.ofam+link+00:10:00:00:00:00:00:04_11_VirtualWall-GW-
   D_">
268.         <openflow:device component_id="urn:publicid:IDN+federation:i2cat.ofam+device+00:10:00:00:00:00:00:04"
   component_manager_id="urn:publicid:IDN+federation:i2cat.ofam+cm"/>
269.         <openflow:port port_num="11"/>
270.         <openflow:device component_id="urn:publicid:IDN+federation:i2cat.ofam+device+VirtualWall-GW-D"
   component_manager_id="urn:publicid:IDN+federation:i2cat.ofam+cm"/>
271.         <openflow:port port_num=""/>
272.       </openflow:link>
273.     </rspec>
```

3- Example of a Request RSpec for VTAM (vtam_request_i2cat.rspec)

```
1. <?xml version="1.0"?>
2. <RSpec type="SFA" expires="2015-02-18T11:21:29Z" generated="2013-02-18T10:21:29Z">
3.   <network name="vt_ocf" slice="">
4.       <node component_manager_id="urn:publicid:IDN+i2cat:vtam+cm"
   component_id="urn:publicid:IDN+i2cat:vtam+node+Verdaguer" exclusive="false"
   component_name="urn:publicid:IDN+i2cat:vtam+node+Verdaguer">
5.             <sliver>
6.                   <name>tutorialVM</name>
7.                   <uuid>myuuid</uuid>
8.                   <project-id>project4</project-id>
9.                   <slice-id>slice2</slice-id>
10.                        <slice-name>slice2</slice-name>
11.                        <operating-system-type>GNU/Linux</operating-system-type>
```

```
                        <operating-system-version>5.0</operating-system-version>
                        <operating-system-distribution>Debian</operating-system-distribution>
                        <server-id>serverid</server-id>
                        <hd-setup-type>file-image</hd-setup-type>
                        <hd-origin-path>default/test/lenny</hd-origin-path>
                        <virtualization-setup-type>paravirtualization</virtualization-setup-type>
                        <memory-mb>128</memory-mb>
                    </sliver>
                </node>
            </network>
        </RSpec>
```

4- Example of a Request RSpec for OFAM (ofam_request_i2cat.rspec)

```
<rspec xmlns="https://github.com/fp7-ofelia/ocf/blob/ocf.rspecs/openflow/schemas/request.xsd"
         xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
         xmlns:openflow="http://www.geni.net/resources/rspec/ext/openflow/3"
         xs:schemaLocation="http://www.geni.net/resources/rspec/3
               https://github.com/fp7-ofelia/ocf/blob/ocf.rspecs/openflow/schemas/request.xsd
               http://www.geni.net/resources/rspec/3/request.xsd
               http://www.geni.net/resources/rspec/ext/openflow/3
               http://www.geni.net/resources/rspec/ext/openflow/3/of-resv.xsd"
      type="request">
    <openflow:sliver email="a@b.com" description="OF request example">
       <openflow:controller url="tcp:10.216.12.29:6633" type="primary"/>
        <openflow:group name="fs1">
           <openflow:datapath
component_id="urn:publicid:IDN+openflow:optin:i2cat.of_ocf+datapath:00:10:00:00:00:00:00:01"
                              component_manager_id="urn:publicid:IDN+openflow:optin:i2cat.of_optin+authority+am"
                              dpid="00:10:00:00:00:00:00:01">
              <openflow:port name="GBE0/4" num="4"/>
              <openflow:port name="GBE0/12" num="12"/>
           </openflow:datapath>
```

```
19.            <openflow:datapath
     component_id="urn:publicid:IDN+openflow:optin:i2cat.of_ocf+datapath:00:10:00:00:00:00:00:03"
20.                               component_manager_id="urn:publicid:IDN+openflow:optin:i2cat.of_optin+authority+am"
21.                               dpid="00:10:00:00:00:00:00:03">
22.               <openflow:port name="GBE0/3" num="4"/>
23.               <openflow:port name="GBE0/3" num="9"/>
24.               <openflow:port name="GBE0/13" num="13"/>
25.               <openflow:port name="GBE0/14" num="14"/>
26.               <openflow:port name="GBE0/15" num="15"/>
27.            </openflow:datapath>
28.            <openflow:datapath
     component_id="urn:publicid:IDN+openflow:optin:i2cat.of_ocf+datapath:00:10:00:00:00:00:00:04"
29.                               component_manager_id="urn:publicid:IDN+openflow:optin:i2cat.of_optin+authority+am"
30.                               dpid="00:10:00:00:00:00:00:04">
31.               <openflow:port name="GBE0/1" num="1"/>
32.               <openflow:port name="GBE0/3" num="3"/>
33.               <openflow:port name="GBE0/13" num="13"/>
34.               <openflow:port name="GBE0/14" num="14"/>
35.               <openflow:port name="GBE0/15" num="15"/>
36.            </openflow:datapath>
37.         </openflow:group>
38.         <openflow:match>
39.            <openflow:use-group name="fs1" />
40.            <openflow:packet>
41.               <openflow:dl_vlan value="755" />
42.            </openflow:packet>
43.         </openflow:match>
44.      </openflow:sliver>
45.   </rspec>
```

5- Different states reported by VTAM during virtual machine creation

RSpec showing virtual machine in "creating" state

```
1.   {
2.     "geni_resources": [
```

```
      {
        "node-name": "Verdaguer",
        "vm-name": "tutorialVM",
        "vm-ip": "10.216.12.55",
        "vm-state": "creating..."
      }
    ],
    "geni_urn": "urn:publicid:IDN+geni:gpo:gcf+slice+sfatest",
    "geni_status": "ready"
  }
```

RSpec showing virtual machine in "running" state

```
{
  "geni_resources": [
    {
      "node-name": "Verdaguer",
      "vm-name": "tutorialVM",
      "vm-ip": "10.216.12.55",
      "vm-state": "running"
    }
  ],
  "geni_urn": "urn:publicid:IDN+geni:gpo:gcf+slice+sfatest",
  "geni_status": "ready"
}
```

RSpec showing virtual machine in "stopped" state

```
{
  "geni_resources": [
    {
      "node-name": "Verdaguer",
      "vm-name": "tutorialVM",
      "vm-ip": "10.216.12.55",
      "vm-state": "stopped"
    }
  ],
  "geni_urn": "urn:publicid:IDN+geni:gpo:gcf+slice+sfatest",
```

```
  "geni_status": "ready"
}
```

6- Example of a Manifest RSpec from VTAM

```
<?xml version="1.0" ?>
  <!-- Reserved resources for:
      Slice: sfatest
      at AM:
      URN: unspecified_AM_URN
      URL: https://137.222.204.27:5001/xmlrpc/sfa/
 -->
  <rspec expires="2014-11-25T17:06:39Z" generated="2014-11-25T16:06:39Z" type="manifest"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.protogeni.net/resources/rspec/2
https://github.com/fp7-ofelia/ocf/blob/ocf.rspecs/server_schema.xsd http://www.protogeni.net/resources/rspec/2/ad.xsd">
    <network name=".i2cat">
            <node component_id="urn:publicid:IDN+i2cat:vtam+node+Verdaguer"
component_manager_id="urn:publicid:IDN+i2cat:vtam+cm" component_name="urn:publicid:IDN+i2cat:vtam+node+Verdaguer"
exclusive="false">
                <hostname>verdaguer.ctx.i2cat.net</hostname>
                <name>Verdaguer</name>
                <operating_system_type>GNU/Linux</operating_system_type>
                <operating_system_distribution>Debian</operating_system_distribution>
                <operating_system_version>6.0</operating_system_version>
                <virtualization_technology>xen</virtualization_technology>
                <cpus_number>None</cpus_number>
                <cpu_frequency>None</cpu_frequency>
                <memory>None</memory>
                <hdd_space_GB>None</hdd_space_GB>
                <agent_url>https://verdaguer.ctx.i2cat.net:9229/</agent_url>
                <services>
                    <login authentication="ssh-keys" hostname="10.216.12.55" port="22" username="root"/>
                </services>
                <sliver type="VM">
                    <name>tutorialVM1</name>
```

```
                    <state>creating...</state>
                    <ip>10.216.12.55</ip>
                </sliver>
            </node>
        </network>
    </rspec>
```

7- Different states reported by OFAM during request

RSpec showing ready state

```
{
  "geni_status": "ready",
  "geni_urn": "urn:publicid:IDN+geni:gpo:gcf+slice+sfatest",
  "geni_resources": [
    {
      "granted_flowspaces": [
        {
          "openflow": [
            {
                    "port_number_s": 4,
                    "direction": 2,
                    "port_number_e": 4,
                    "dpid": "00:10:00:00:00:00:00:01"
                },
                {
                  "port_number_s": 12,
                  "direction": 2,
                  "port_number_e": 12,
                  "dpid": "00:10:00:00:00:00:00:01"
                },
                {
                  "port_number_s": 4,
                  "direction": 2,
```

```
                  "port_number_e": 4,
                  "dpid": "00:10:00:00:00:00:00:03"
                },
                {
                  "port_number_s": 9,
                  "direction": 2,
                  "port_number_e": 9,
                  "dpid": "00:10:00:00:00:00:00:03"
                },
                {
                  "port_number_s": 13,
                  "direction": 2,
                  "port_number_e": 13,
                  "dpid": "00:10:00:00:00:00:00:03"
                },
                {
                  "port_number_s": 14,
                  "direction": 2,
                  "port_number_e": 14,
                  "dpid": "00:10:00:00:00:00:00:03"
                },
                {
                  "port_number_s": 15,
                  "direction": 2,
                  "port_number_e": 15,
                  "dpid": "00:10:00:00:00:00:00:03"
                },
                {
                  "port_number_s": 1,
                  "direction": 2,
                  "port_number_e": 1,
                  "dpid": "00:10:00:00:00:00:00:04"
                },
                {
                  "port_number_s": 3,
```

```
                    "direction": 2,
                    "port_number_e": 3,
                    "dpid": "00:10:00:00:00:00:00:04"
                  },
                  {
                    "port_number_s": 13,
                    "direction": 2,
                    "port_number_e": 13,
                    "dpid": "00:10:00:00:00:00:00:04"
                  },
                  {
                    "port_number_s": 14,
                    "direction": 2,
                    "port_number_e": 14,
                    "dpid": "00:10:00:00:00:00:00:04"
                  },
                  {
                    "port_number_s": 15,
                    "direction": 2,
                    "port_number_e": 15,
                    "dpid": "00:10:00:00:00:00:00:04"
                  }
                ],
                "flowspace": {
                  "tp_dst_e": 65535,
                  "ip_proto_e": 255,
                  "mac_src_e": "ff:ff:ff:ff:ff:ff",
                  "mac_dst_e": "ff:ff:ff:ff:ff:ff",
                  "vlan_id_s": 754,
                  "ip_src_e": "255.255.255.255",
                  "tp_src_e": 65535,
                  "ip_dst_s": "0.0.0.0",
                  "eth_type_e": 65535,
                  "mac_src_s": "00:00:00:00:00:00",
                  "vlan_id_e": 754,
```

```
                "mac_dst_s": "00:00:00:00:00:00",
                "ip_proto_s": 0,
                "tp_dst_s": 0,
                "eth_type_s": 0,
                "ip_dst_e": "255.255.255.255",
                "tp_src_s": 0,
                "ip_src_s": "0.0.0.0"
              }
            }
          ]
        }
      ]
    }
```