

# Static routing experiment

We describe here the setup of a simple static routing experiment using the OFELIA Control Framework (OCF). This experiment demonstrates how to create virtual machines and how to allocate a specific isolated network topology within the FIBRE network. The goal of the experiment is to setup the routing as indicated by Figure 1; i.e. packets from endhost1 sent to the IP address 192.168.2.30 on endhost3 should be routed via endhost2.

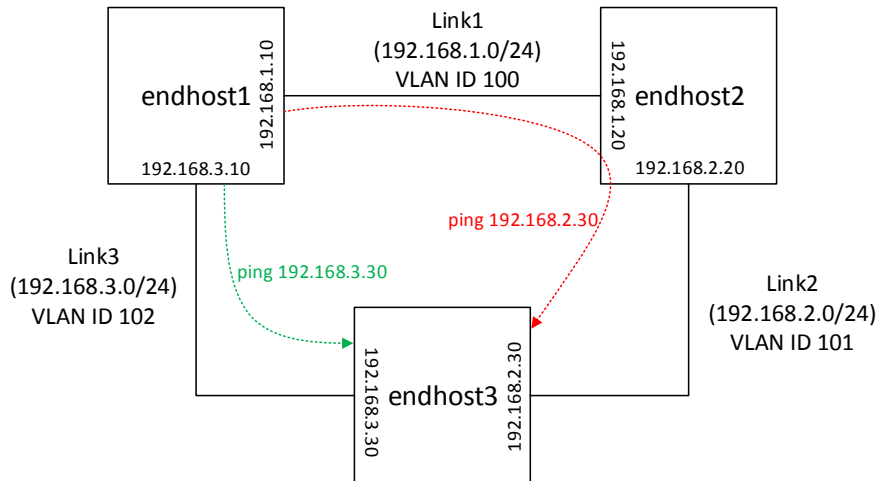


Figure 1. Static Routing Experiment.

In this experiment, we use three virtual machines in different sites of the testbed (end-hosts in Figure 3) connected through three isolated networks in a mesh topology. Each network link is an independent virtualized network that is dynamically allocated on top of the experiment data plane with a specific VLAN ID (100, 101 and 102 in this case). The endhost2 must be configured to forward packets coming through its network interfaces.

## Setting up the experiment

OCF uses the concept of “slice” to describe an experiment. Slices describe the resources to be used and the associated configurations, and encompass the state of the experiment. Once the student has permission to participate in an experiment, he or she can either use existing slices associated to this experiment, or create new ones. When creating a new slice, information such as name, description and expiration date (the slice life-time) must be provided. After slice expiry, the FIBRE island may deallocate the reserved resources.

To set up an experiment, a student must first request access to a project in the OCF interface. Students can see and modify configurations in projects for which they have been granted permission. These permissions are usually granted by the project owners (teachers).

The experimenter must begin the setup of the experiment by making a reservation of the desired resources on the appropriate OCF instance. For cross-island experiments, experimenters must access the OCF instance running at the NOC and add to their project the necessary aggregates from all desired islands; experiments configured by an OCF instance running at an island may use only its own local resources.

On the Slice Management page (Figure 2), the Topology panel shows the physical topology of

the resources in the aggregates available to the slice. These resources may comprise the available OpenFlow switches, virtualization servers, and the connections between switches and servers.

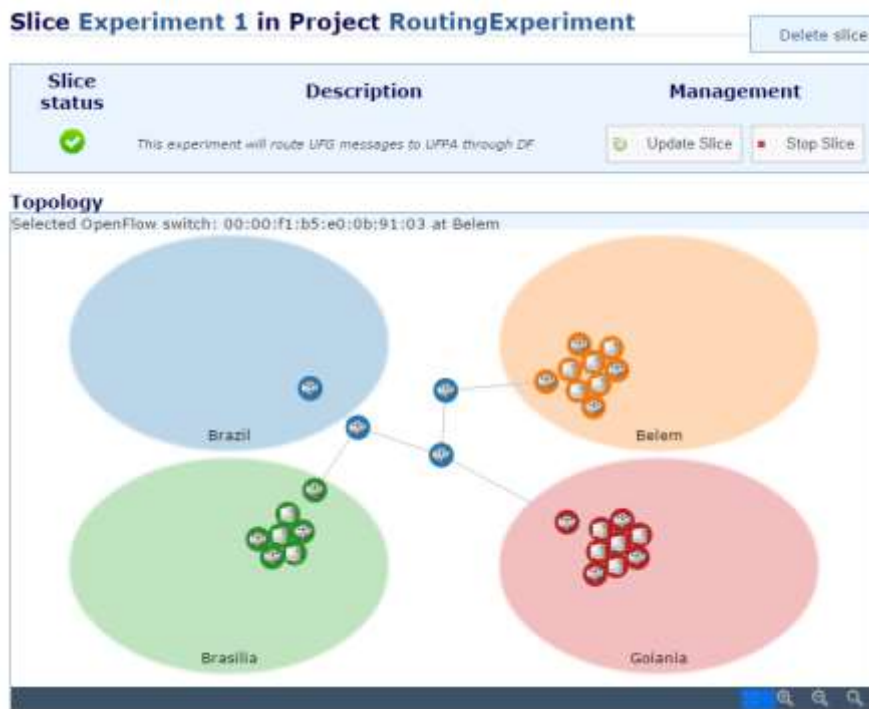


Figure 2. Physical topology of the experiment slice

Virtual Machines (VMs) for experiments are created in a Computational Resources area on the Slice Management page (Figure 3). During the creation process, the VM will be granted an IP, which will be displayed in the Topology Panel and the Computational Resources area. This IP is only reachable through FIBRE's VPN. VMs may be started, stopped, rebooted or deleted by clicking on the respective action link in the Computational Resources area. Experimenters can create as many VMs as needed for their experiment.



Figure 3. Computational Resources area.

To use OpenFlow resources, experimenters are required to add OpenFlow resources to their slice and specify an OpenFlow Controller for the experiment. On the Slice Management page

in the OpenFlow Aggregate area (Figure 4), data paths (consisting of id and port number) that are available in the slice can be selected in the Topology panel to define FlowSpaces for the experiment. When the slice starts, a specific VLAN ID is reserved for each defined FlowSpace on the data plane. Every virtual machine instantiated has a dedicated interface that is connected to the data plane network.



Figure 4. OpenFlow Aggregate area.

Once the FlowSpaces are selected and the VMs are created, the user must set the controller IP address to match the VM that hosts the OpenFlow controller. To carry out the experiment, the student starts the slice on the Slice Management page. This will trigger a FlowSpace request to the OpenFlow switches involved and make sure all the VMs in the slice are active.

Within the slice, the experimenter can use the VMs as end-hosts and the FlowSpaces (allocated on the OpenFlow switch fabric) as the network data-plane. Students can access the running VMs through SSH, using their FIBRE username and password over FIBRE's VPN. Students can then install on the VMs any OpenFlow controllers or arbitrary software that is needed for their experiment (a set of OpenFlow controllers are pre-installed in the VM images, but students can also provide their own controller implementation.).

## Configuring and running the experiment

In this section, we go through the network configuration procedures to configure the static routing experiment of Figure 1. For this experiment, we allocated three FlowSpaces and three virtual machines. The topology of each reserved FlowSpace is shown on Figure 5.

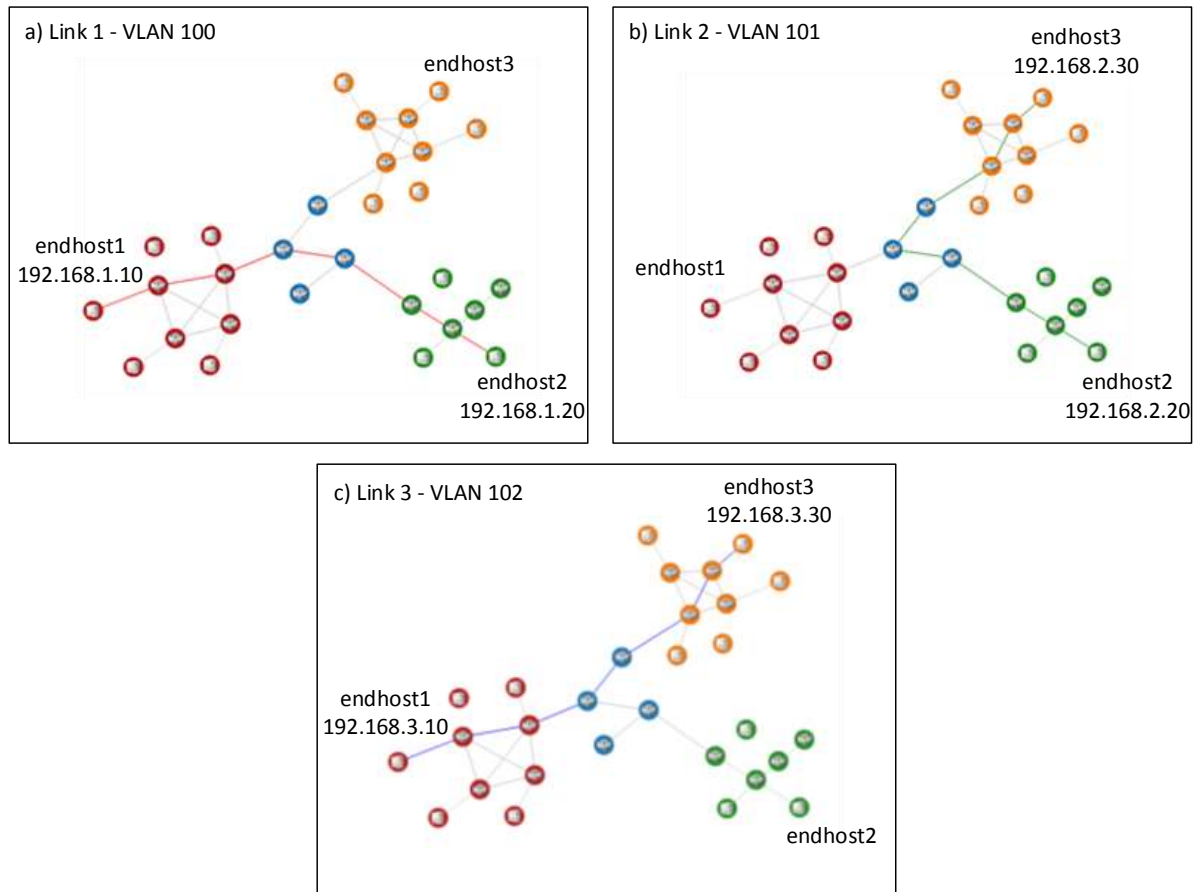


Figure 5. The experiment virtual network topology.

An additional VM is created to host the experiment OpenFlow controller, such as Floodlight<sup>1</sup>, POX<sup>2</sup> or Ryu<sup>3</sup>, acting as a L2 learning switch. A L2 learning switch learns the mapping between MAC addresses and ports by watching packets. If the switch has already seen a particular destination, it can send to exactly one port; otherwise, it must flood the packet out all ports like a hub to find out the corresponding destination.

To configure the data plane network in the instantiated VMs, the student must access each VM through the provided control plane IP via SSH and configure the virtualized network for the experiment according to the configuration shown in Figure 1. For example, Listing 1 contains the steps for configuring network interfaces, VLANs, IPs and routing of the endhost1 VM.

```
root@endhost1:~# vconfig add eth1 100
```

<sup>1</sup> <http://www.projectfloodlight.org/floodlight/>

<sup>2</sup> <http://www.noxrepo.org/pox/about-pox/>

<sup>3</sup> <http://osrg.github.io/ryu/>

```
Added VLAN with VID == 100 to IF -:eth1:-
root@endhost1:~# ifconfig eth1 up
root@endhost1:~# ifconfig eth1.100 192.168.1.10
root@endhost1:~# vconfig add eth1 102
Added VLAN with VID == 102 to IF -:eth1:-
root@endhost1:~# ifconfig eth1.102 192.168.3.10
root@endhost1:~# route add -host 192.168.2.30 gw 192.168.1.20 eth1.100
```

Listing 1. Configuring the endhost 1 for the routing experiment

After configuring the network of all end hosts, the experimenter must enable packet forwarding in the endhost2 VM with the command shown in Listing 2.

```
root@endhost2:~# echo 1 > /proc/sys/net/ipv4/ip_forward
```

Listing 2. Enabling IP packet forwarding in the endhost2 VM.

Finally, the experimenter can verify the correctness of their configuration by using the ping command in the endhost1 VM, as described in Listing 3. Note that the round trip latency of the first packet is greater because of the behaviour of L2 learning switch dynamic port discovery.

```
root@endhost1:~# ping 192.168.2.30
PING 192.168.2.30 (192.168.2.30) 56(84) bytes of data.
64 bytes from 192.168.2.30: icmp_req=1 ttl=62 time=641 ms
64 bytes from 192.168.2.30: icmp_req=2 ttl=62 time=53.0 ms
...
```

Listing 3. Testing the routing experiment.

Through the procedures described in this section, students and educators can use the FIBRE testbed to allocate the dedicated virtual networks and virtual machines necessary for setting up a basic network routing experiment that spans multiple sites.