Demonstrating inter-testbed network virtualization in OFELIA SDN experimental facility

M. Gerola, R. Doriguzzi Corin, R. Riggio
F. De Pellegrini, E. Salvadori
CREATE-NET
via Cascata 56/D
38123 Trento (I)

H. Woesner, T. Rothe
EICT GmbH
Ernst-Reuter-Platz 7
10587 Berlin (D)

M. Suñé, L. Bergesio
i2CAT Foundation
Gran Capita, 2-4
08034 Barcelona (E)

***Abstract*—OFELIA is an experimental network designed to offer a diverse OpenFlow-enabled infrastructure to allow Software Defined Networking (SDN) experimentation. OFELIA is currently composed of ten sub–testbeds (called islands), most of them in Europe and one in Brazil. An experimenter get access to a so-called slice; a subset of the testbed resources like nodes and links, including the Openflow programmable switches to carry on an experiment. A new network virtualization tool called VeRTIGO has been recently presented to extend the way isolation is achieved between slices (slicing), allowing each experimenter to instantiate an arbitrary virtual network topology on top of a physical testbed. In this paper we present preliminary results obtained by deploying and using VeRTIGO in an experiment running across several OFELIA islands, which has proven to increase flexibility to experimenters willing to play with novel SDN concepts at large scale.**

***Index Terms*—Testbed, Network Virtualization, OpenFlow, Software Defined Network**

I. Introduction

OFELIA (OpenFlow in Europe – Linking Infrastructure and Applications) is an experimental network designed to offer a diverse OpenFlow-enabled infrastructure that allows Software Defined Networking (SDN) experimentation on a programmable, multi-layer and multi-technology network.

OFELIA is currently composed of ten sub-testbeds (called islands), most of them in Europe and one in Brazil. The basic common infrastructure of all sub-testbeds is the shared L2 Ethernet OpenFlow network infrastructure, along with virtualized end-hosts and other resources, made available in a Infrastructure as a Service (IaaS) like fashion. With this infrastructure a complete network experiment setup can be deployed and run, which includes end-hosts to establish a communication over (e.g. a web-server and a client) the Openflow network and to deploy the experiment controller(s) to program its forwarding plane.

As many experimenters run different experiments over the same testbed substrate, isolation among resources associated with each experiment is needed by properly slicing them. A "slice" consists of a set of resources which are explicitly reserved and configured to be used by the (network) experiment. In the current deployment of the OFELIA testbed, slicing is performed with the OpenFlow network virtualization tool called FlowVisor. As highlighted by the authors in [1], one of its major limitations is the inability to establish virtual topologies not restricted by the substrate physical topology.

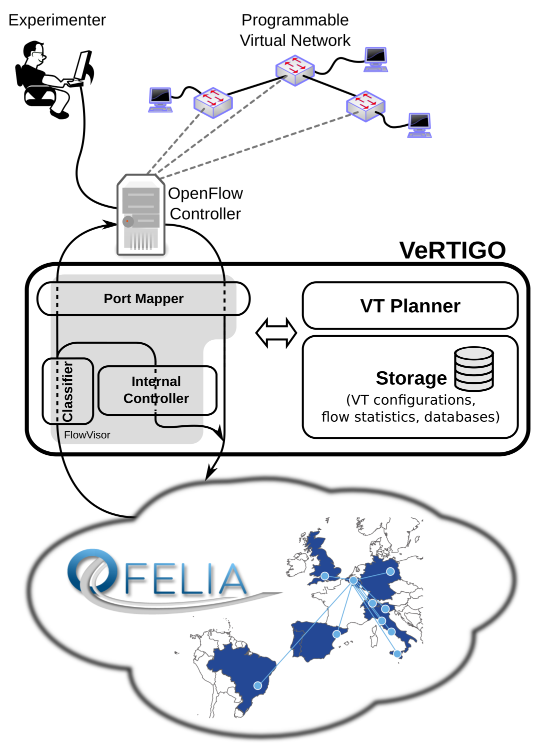

Fig. 1. VeRTIGO architectural building blocks.

A new component named VeRTIGO (ViRtual TopologIes Generalization in OpenFlow networks) is currently under development within the framework of OFELIA project [2]. VeRTIGO has the objective to overcome current FlowVisors limitations by extending the concept of virtual networks to any kind of arbitrary topology and any number of network nodes. The work in [2] presents the basic architecture and some preliminary results obtained out of a single testbed.

In this paper we present the results of an early deployment of VeRTIGO, showing how it can run on several testbeds and thus extending the network virtualization layer capabilities

∗Work partially supported by the EU FP7 project OFELIA http://www.fp7-ofelia.eu

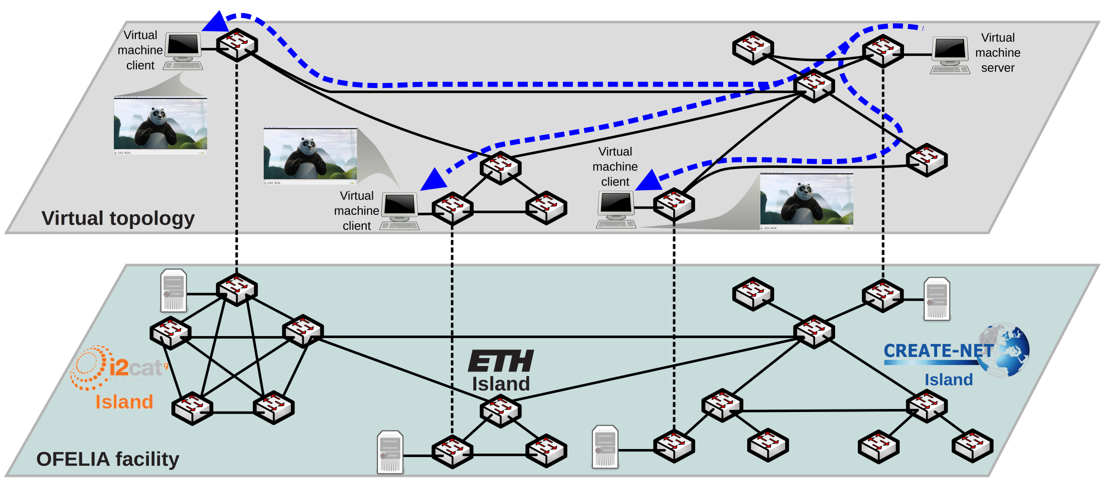

Fig. 2. Pilot setup.

across the whole OFELIA experimental facility.

II. Architecture

As shown in Figure 1, VeRTIGO architecture has been designed to allow requests of virtual network configurations from users through the OFELIA Control Framework[3]. In particular, an experimenter is able to build a request of a virtual network via a Web-GUI and then to perform an experiment on it by deploying and customizing his/her own Openflow controller.

VeRTIGO extends FlowVisor to allow the instantiation of generalized virtual network topologies through the implementation of virtual links, as an aggregation of multiple physical links on OpenFlow-enabled switches. VeRTIGO also allows to further improve the level of abstraction of the network virtualization layer by slicing network nodes (e.g. network switches) into several virtual nodes, thus giving the user/experimenter the capability of designing fully virtual networks, even if they consist of more nodes than the ones available down into the physical infrastructure.

Figure 1 shows the software architecture of VeRTIGO's software, and the functional interaction among the different modules during the communication between an OpenFlow switch and a controller (see [2] for more details). In particular, the VT Planner submodule contains a set of smart algorithms that can efficiently map experimenter's resource requests down to the available infrastructure, according to the status of the physical network and hosts, resource availability and other criterias. VeRTIGO allows the experimenter willing setup his/her own Openflow experiment to define a certain (virtual) network, and then either (i) manually instantiate virtual links and virtual nodes on the available physical infrastructure or (ii) sketch the desired topology via a WebGUI a network, and then leave the VT Planner to select (and accommodate) the best resources to use to set up the experimental network, according to the criterias and constraints to be minimized/maximized. In both cases the result will be a programmable network that VeRTIGO defines it through a set of tuples included in the configuration file, specifying each component of the instantiated experimental network.

III. Pilot deployment

The objective of the pilot deployment within OFELIA is to show how VeRTIGO can be applied (i) to instantiate a virtual network with arbitrary topology across several islands and (ii) to demonstrate its effectiveness by performing some experimental activities on top of it.

The pilot setup involves three islands, one in CREATE-NET, Trento (Italy), one in ETH, Zürich (Switzerland) and one in i2CAT, Barcelona (Spain). As described in the previous section, a certain virtual network is then "carved out" from these three physical testbeds by selecting which links, switches and hosts include in each island . As highlighted in Figure 2, a certain virtual network across the three testbeds is then instantiated, together with four virtual hosts.

On top of the instantiated network, a multicast traffic management software, based on a SDN architectural framework was used. The results of this experiment have shown that, compared to a fully distributed mechanism, with SDN is possible to better manage link failures affecting multicast traffic since the controller can compute the best redundancy subtrees offline, and then apply them when such event happens. During this experiment, we were able to demonstrate that a video stream flowing from a server host in Trento to three clients in different islands is not perturbed by the failure of a physical link corresponding to a link in the virtual topology used by the initial multicast subtree. As soon as this subtree fails, a new (precomputed) subtree is "installed" in the involved switches, thus minimizing traffic disruptions.

References

[1] R. Sherwood et al., "Can the production network be the testbed?" in *Proc. of USENIX OSDI*, Canada, 4-6 Oct. 2010.

[2] R. Doriguzzi Corin et al., "VeRTIGO: Network virtualization and beyond," in *Proc. of EWSDN*, Germany, October 2012.

[3] "Ofelia control framework (ocf)," http://fp7-ofelia.github.com/ocf.