Multi-domain Software Defined Networking: Exploring possibilities

Authors and author affiliations

Eduardo Jacob, Alaitz Mendiola
University of the Basque Country UPV/EHU, Spain
{ eduardo.jacob | alaitz.mendiola }@ehu.es

Łukasz Podleski, Radek Krzywania, Miłosz Przywecki, Krzysztof Dombek, Artur Juszczyk
PSNC, Noskowskiego 12/14, 61-704 Poznan, Poland
{ docere | radek.krzywania | mprzyw | kdombek | juszczyk }@man.poznan.pl

José Ignacio Aznar-Baranda, Albert Vico-Oton
The i2CAT Foundation, Barcelona, Spain
{ jose.aznar | albert.vico }@i2cat.net

XavierJeannin
Renater, Paris, France
xavier.jeannin@renater.fr

Kurt Baumann
SWITCH, Zurich, Switzerland
kurt.baumann@switch.ch

Christos Argyropoulos
National Technical University of Athens - NTUA and GRNET, Greece
cargious@netmode.ntua.gr

Paper type

Technical paper

Abstract

Since the appearance of OpenFlow, the Software Defined Networking has become a hot topic inside the networking industry and academia. One of the topics currently under study, is the adaptation of SDNs to the multi-domain environment, as current OpenFlow lacks of mechanisms by itself to accomplish that. In this scenario, the GN3plus JRA2T1 task group is investigating how to integrate SDN/OpenFlow with the Network Services Framework, which has been proven useful for its application in multi-domain environments. The research is focused on two different areas. The first one tackles with the establishment of end-to-end connections across different domains where at least one of the domains is SDN/OpenFlow-enabled. The second one aims to provide isolated slices to experimenters where the resources belong to different domains.

Keywords

Multi-domain, Network Services Interface – Connection Service, Network Services Framework, Software Defined Networking, SDN, OpenFlow.

1. Introduction

Software Defined Networking (SDN) [1] is a novel networking paradigm that has rapidly evolved since its appearance a few years ago. SDN technologies, and the OpenFlow protocol [2] in particular, can be used effectively to provide the environment for clouds, networking test-beds, campus networks and in general to all applications that require fast adaptability (security, traffic engineering, Network-as-a-Service) and programmability. As a consequence, Service Providers in general and NRENs in particular are also experimenting with the integration of SDN/OpenFlow based technologies in the core of both their network and cloud facilities.

One of the SDN/OpenFlow related hot topics currently under investigation is the adoption of SDN/OpenFlow multi-domain solutions, with the objective of supporting the use of aggregated resources for research and end-users with large scale trials on top of which OpenFlow-aware services could be deployed. Nevertheless, current SDN solutions do not naturally support multi-domain environments. Several initiatives have recently started working on it (e.g. FELIX [3] or the FI-PPP XIFI project [4]) trying to provide a smart solution to the SDN/OpenFlow multi-domain panorama. In order to solve this limitation, one of the most promising solution relies on the Network Services Framework (NSF)[5] and the Network Service Interface – Connection Service (NSI-CS)[6] protocol. Being a technology agnostic solution, which has been proven useful for its application in different transport networks, it is the ideal candidate to provide the multi-domain features that SDN/OpenFlow lacks of.

In this scenario, the GN3plus JRA2T1 task group is working on a solution able to provide both, end-to-end federated slices and connectivity services across several domains. Firstly, the connection-oriented solution is focused on providing circuits or connectivity services in a multi-domain environment. It encompasses both cases, when all the domains involved are OpenFlow enabled, and when they are not. Secondly, the slice-oriented multi-domain SDN solution aims to create distributed network slices, composed by network resources that belong to different domains. Such solution should enable the creation of federated test-beds or datacenters, which could be perceived as one logical domain, resulting in an easier control of the resources (e.g. FP7 NOVI project [7]).

As it can be deduced, the slice-oriented solution needs to tackle with the connectivity between the resources that belong to different domains as well. Thus, the connection-oriented solution is going to be considered a transversal solution, which can be used as a basis to achieve slice-oriented multi-domain SDN so as pure connection-oriented services such as end-to-end Bandwidth on Demand.

This paper is organized as follows. Section 2 introduces the motivations, including a use case for the connection-oriented solution. Then, Section 3 describes NSF and NSI-CS and presents the GN3plus JRA2T1 task group approach to integrate OpenFlow with NSI-CS. Section 4 provides a detailed description of the slice-oriented approach and presents the OpenNaaS orchestrator. Finally, Section 5 summarizes the conclusions.

2. Motivations

The extension of SDN/OpenFlow towards a widely accepted multi-domain networking programmable platform is hindered by several barriers that require experimental research. Future Internet end-users (experimenters, cloud application developers, virtualized data-center operators) require flexible and elastic provisioning, management and control of the ICT resources, distributed in multiple administrative domains. To that end, experimental research has to be performed at the provisioning and control planes of independently managed yet interconnected Software Defined Network infrastructures that support control and data plane decoupling through SDN/OpenFlow. Inter-domain flowspace reservations, flow-table updates and orchestration of policy-based schemas into a federated service layer are key challenges to be addressed and validated.

Aforementioned, the GN3plus JRA2T1 task group is precisely concentrated on two approaches: the slice-oriented solution and the connection-oriented solution. The connection-oriented solution can be useful for the provisioning of different services. In fact, it can also be useful to tackle with the multi-domain connectivity required in the slice-oriented case to connect the different elements. For that reason, a use case of the connection-oriented solution is presented.

2.1. Extension of the BoD service to OpenFlow based campus networks

Due to the increasing demand of high-capacity and highly reliable connections, most world leading NRENs as well as Internet2 or GÉANT are offering in their portfolio point-to-point connectivity services. One of the most common services are OSCARS and AutoBAHN, which provide Bandwidth on Demand to users.

When users request this type of services, they are provisioned only inside the domain that offers it. The service does not traverse different domains and it is not provisioned in the campus network the user is connected to.

Furthermore, very similar and compatible services are implemented differently in each domain, where even the transport network and provisioning mechanism can vary. In order to improve this situation and support the provisioning of the service across different NRENs and campus networks, and to support, in general, the establishment of point-to-point circuits that traverse more than one domain, multi-domain solutions are needed.

These multi-domain solutions benefit researchers trying to conduct a global experiment that requires the establishment of a connection between two distant hosts not directly connected through their campus network. For instance, Bandwidth on Demand services are often used by researchers involved in high-particle physics research or genetics.

When one of the domains involved is SDN/OpenFlow enabled, the multi-domain technology must be able to leverage the granularity provided by OpenFlow and dispose of the required mechanism to guarantee that the service is still going to be provided even in those domains that do not have such level of granularity. Figure 1 depicts the case where a user connected through a SDN/OpenFlow enabled campus obtains a connection-oriented service that traverses both, the SDN/OpenFlow based campus network and the NREN network, which may or may not be SDN/OpenFlow based.

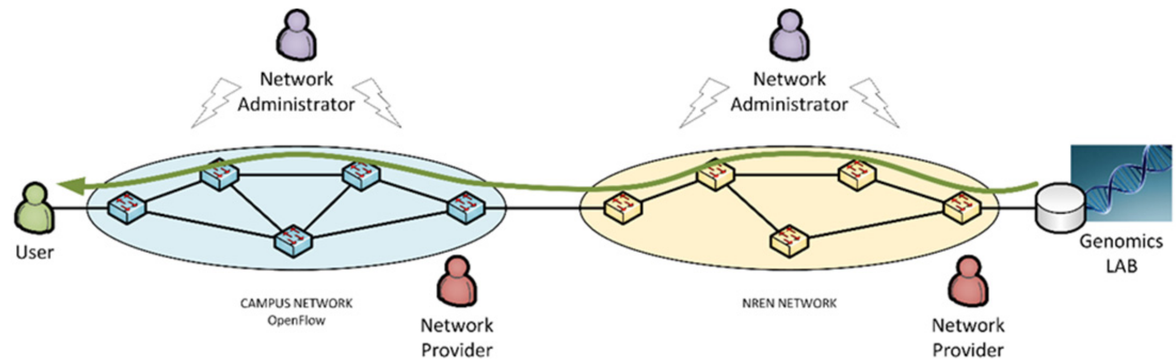

Figure 1. Connection-oriented SDN/OpenFlow multi-domain use case

In such scenario, the network providers involved must agree the type of the service that are going to provide to users. Accordingly, the networks administrators must configure the network to fulfil the specific requirements of the service. From that moment, users are able to request the service to their network provider, which will request it also to the remaining domains involved.

The SDN/OpenFlow multi-domain solution must be able to satisfy the following requirements:

- Users should be able to request a service through its local network provider.
- Users should be able to obtain end-to-end connectivity services with certain QoS guaranteed.
- Network providers should be able to negotiate the service characteristics with the network providers of the remaining domains.
- Network should be automatically configured whenever possible.
- Services should be automatically provided when requested by users
- Services should only be provided after an authentication and authorization process
- The required mechanisms to guarantee that a service has been successfully removed should be provided.
- The solution must be unique and technology agnostic, meaning that it should be valid for any combination of SDN/OpenFlow domains and non-SDN/OpenFlow domains.
- It should make the most out of SDN/OpenFlow granularity and flexibility.

3. Connection-oriented multi-domain SDN

As previously stated, the NSF and the NSI-CS protocol are ideal candidates to add multi-domain capabilities to SDN/OpenFlow technologies. When at least one of the domains involved in the multi-domain is SDN/OpenFlow, the term multi-domain SDN/OpenFlow is used. In this section a technology overview is provided, presenting the most interesting aspects of NSI, so as our proposal for the integration of both technologies.

3.1. Network Services Framework

The NSF is an effort of the Open Grid Forum (OGF) that describes network resources as manageable objects and enables the automated provisioning of federated network services. Within the framework, network services are used by applications to monitor, control, interrogate and support the network resources. For the moment, three key network services have been defined. Firstly, the NSI Connection Service (NSI-CS) provides the means to create connections that traverse different network domains. Secondly, the NSI Topology Service (NSI-TS) is used to share topological information using NSI Topology, a standard ontology and schema based on the Network Markup Language (NML). Finally, the NSI Discovery Service (NSI-DS) is a web service that allows the discovery of available resources.

Furthermore, the NSF defines a set of NSI architectural elements. The Network Service Agent (NSA) is a software agent that implements the NSI protocol. If the NSA request a service, it is called the ultimate Requester Agent (uRA), and if it provides the service, ultimate Provider Agent (uPA). In addition, the NSA can also have the role of an aggregator (AG), which aggregates the responses from child NSAs and acts as a gateway to other providers. These elements, the NSI protocol and the NSAs reside on a plane called NSI Service Plane.

3.2. SDN/OpenFlow integration in NSF

Worth mentioning that the NSF identifies two different topologies: the inter-domain and the intra-domain topologies. The first one is concerned with the interconnection of networks whereas the second one is related to the resources within the network. Being NSI a technology to obtain federated services across networks, only the inter-domain topology is within the scope of the framework. The NSF aims to be a technology agnostic solution, meaning that it is intended to work regardless the underlying transport technology used at the network. To be able to do that, the NSF defines the Network Resource Manager (NRM) to control and manage the local network resources, which is actually outside the notional NSI Service plane.

This feature is what makes NSI an ideal candidate for its integration with SDN/OpenFlow. By using an OpenFlow controller as the NRM of a domain, it is possible not only to obtain multi-domain connectivity services between SDN/OpenFlow enabled domains, but also between SDN/OpenFlow and non-SDN/OpenFlow domains. In Figure 2 the NSI service plane with the corresponding architectural elements is depicted. It also shows how an OpenFlow controller can be integrated within the NSF as an NRM.

One of the most interesting topics which needs to be addressed is how to setup multi-domain connections leveraging the SDN/OpenFlow capabilities in heterogeneous environments. OpenFlow introduced a very flexible traffic differentiation mechanism that allowed packet forwarding to be done using much more parameters than in the traditional forwarding approach. The flexibility provided by OpenFlow enables circuits to be established in numerous forms – from pure Layer 1 connections (port-port), through Layer 2 and Layer 3 (IP address space) to Layer 4 TCP/UDP ports.

However, in the NSF, where the requested service must be supported in all the domains involved, the granularity of OpenFlow imposes some challenges. Since not every domain has to be SDN/OpenFlow-enabled, optional SDN/OpenFlow-specific parameters need to be passed without disabling the possibility to setup a circuit in non-SDN/OpenFlow domains. The most recent specification of this service (v2.0) introduced new data model elements that enable the extension of the base functionalities without changing the core elements of the protocol. One of the most important features, from this considerations point of view, is the existence of the 'ANY'[1] attribute in NSI messages, which semantic meaning depends on a defined namespace instead of on pre-defined assumptions [8].

[1] According to the specification, the 'ANY' attribute "provides a flexible mechanism allowing additional elements to be provided such as the service specific attributes specified by 'serviceType'. Additional use of this element field is beyond the current scope of this NSI specification, but may be used in the future to extend the existing protocol without requiring a schema change" [8].

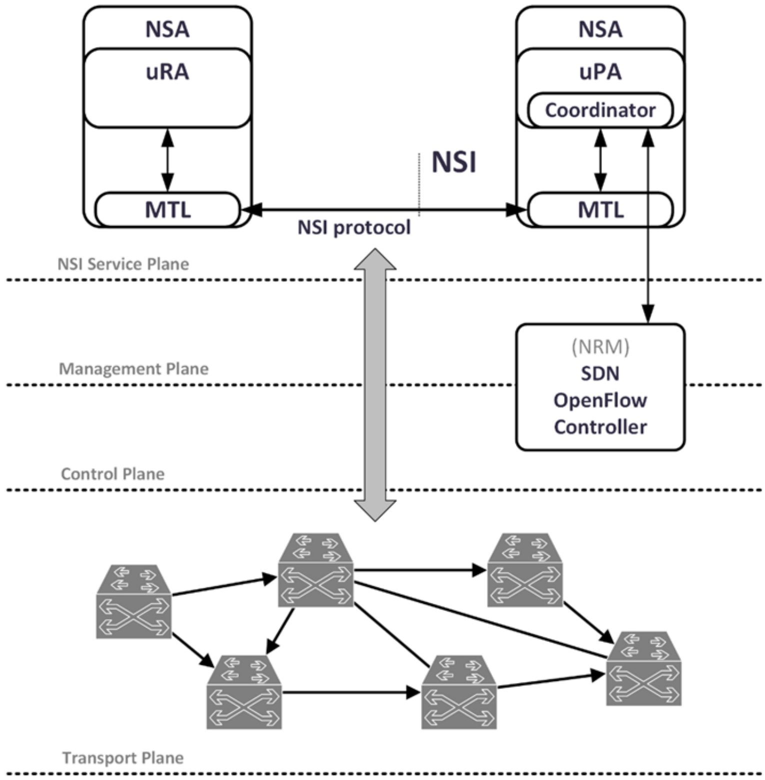

Figure 2. OpenFlow controller integration within NSF

By utilizing the newest Service Termination Point (STP) definition[2], it is possible to code into the 'TypeValueType' string attribute [9] information regarding e.g. a range of VLANs, a L3 IP subnet/range or even the L4 TCP/UDP ports available on the interface. In that way, appropriate STP ports can be chosen to transport network traffic with specific network parameters (characteristics). The schema of 'Reserve' connection request remains untouched and for a connection reservation process basic NSI service type (P2P) can be used, providing compatibility with the non-SDN domains [10].

In the second version of NSI-CS, the definition of the 'Reserve' request message uses the 'serviceType' field (inside the ReservationRequestCriteria object) in order to transport additional (technology- or domain-specific) parameters within the protocol message [11]. SDN/OpenFlow-specific information can be passed in the form of new base service type or by defining a new optional namespace in the request.

- **New service base type:** The protocol defines a base P2P service type that provides a set of properties for multi-domain connections [12]. However, there is a possibility to define additional parameters (placed in the 'ANY' attribute), which could make possible to perform the agreement between the different service domains agents involved. As a consequence, the whole service could be provided to the customers. In order to provide SDN/OpenFlow-specific connections, a new service type can be proposed. It should extend the service base type with layer 3 or/and layer 4 fields to enable the setup of more granular flow-based connections. There is also the opportunity to define a new service with the needed parameters and attributes.
- **New OpenFlow/SDN-specific namespace:** Despite service-specific attributes, 'Reserve' request possess the ReservationRequestCriteria object with a property called 'anyAttribute', which has been added in order to be able to cope with any domain/technology-specific extensions without a need of modification of the protocol core and the addition of new service parameters or the whole service. Custom SDN namespace might enable flexible approach to the management of network resources and

[2] Service Termination Point (STP) represents the logical/virtual ports of the domain. Ingress and egress traffic is transported through these ports and between them NSI connection is setup. Definition of the STP (despite of location, device serial number, physical port etc.) also contains information regarding network resources present on that port. [9]

enable the use of connection-related SDN apps (e.g. custom statistics monitoring, packet inspection, resiliency or load balancing etc.) or suggest a quality of service for the circuit. Domains that do not support this extension(s) should silently ignore them, instead of dropping the whole request. As one of the NSI's basic service types, [12] still can be used in the connection request, circuits via non-SDN domains can be easily established.

4. Slice-oriented multi-domain SDN: A centralized approach

The main objective of the slice-oriented solution is to enable the creation of sets of federated and isolated resources across several domains for different users. Such multi-domain federation entails (i) the SDN domains slices creation and (ii) E2E multi-domain connectivity among such slices to build the multi-domain federation. Several services are involved to accomplish the previously stated two fold step process, namely, a connection management service, an slicing management service (to create and handle the slices) and also a compute management service to manage the storage and computing capabilities assigned to each slice. This proposal aims to leverage most up-to-dated services for the SDN multi-domain framework. For multi-domain connectivity purposes, NSI service has been chosen as described in the previous sections. While provisioning several services across multiple domains, a logical-centralized approach enables to keep an overall updated status of the connection management as well as services' life-cycle. Such centralized approach relies on an orchestrator that acts as management platform and coordinates services while yet delegating the execution of operations locally to each of the domains. This approach smartly fits as a solution to provide with E2E multi-domain SDN resources federation. All in all, such orchestrator can be seen as the entity that solves the gap between users (applications) and the network, making use of several mechanisms (services). OpenNaaS [13] management platform system brings up important features and capabilities which makes it a smart option to such orchestrator.

The slice-oriented multi-domain approach is based on the Slice-Based Federation Architecture (SFA) [14], which is a technical realization of the slice concept: a common API is designed to facilitate a federated slice. Such a slice will consist of resources located in different physical locations and in a variety of technological domains. The SFA specification defines a minimal set of interfaces and data types that are necessary to facilitate the reservation of these resources. From an experimenter's viewpoint, a request for experimental resources will be converted to an SFA-based request, understood by every site within the federation. The SFA specification is designed to simplify the process of federation and thus ease accessibility to different resources for experimenters.

SFA v2.0 defines a control framework architecture to allow a federation of slice-based network substrates to interoperate. In this context, SFA identifies two authority roles for the control and management of a federated system:

- **Management Authority (MA):** responsible for a subset of physical components and ensures the proper behavior of the components (that is, that hose execute the resource allocation accordingly)
- **Slice Authority (SA):** responsible for the registration and control of one or more slices as well as managing the user access to the slices

The main relevant concepts in the SFA are summarized in Table 1.

SFA key concept	Description
Resource	Resources include physical resources (e.g., CPU, memory, disk, bandwidth), logical resources (e.g., file descriptors, port numbers), or synthetic resources (e.g., packet forwarding fast paths). Resources are described through a resource specification (RSpec), typically expressed in XML format following specific schemas. RSpecs are used to list (advertisement RSpec), reserve (request RSpecs), or describe reserved resources (manifest RSpecs) [15], [16]
Component	Components are the primary building blocks of the SFA architecture (e.g., an edge computer, a customizable router, a programmable access-point, etc.). Every component can encapsulate a set of homogeneous or heterogeneous resources, depending on the nature of the component.
Sliver	A sliver can be considered as a resource container, which guarantees the isolation from every other sliver belonging to the same component.

	This requirement can be fulfilled via component virtualization or partitioning the component into distinct resource sets. Either way, the user is granted a sliver of the component.
Slice	A user-defined subset of virtual networking and computing resources, created from the physical resources available in federated domains. A slice has the basic property of being isolated from other slices defined over the same physical resources, and being dynamically extensible across multiple domains. On top of each slice, a specific set of control tools can be instantiated, depending on the specific domains it traverses.

Table 1: SFA main concepts

All the facilities will be controlled programmatically through well-defined interfaces by the Slice-based federation framework, which orchestrates resources in a multi-domain environment. These facilities, extending the slice concept in SFA, are composed of computational and network resources belonging to distributed the different SDN domains in the envisioned multi-domain infrastructure, interconnected via NSI.

A simplified version of the NREN/GEANT multi-domain scenario:

- GEANT (Points of Presence) PoPs connected in a full mesh fashion. Links between PoPs are configured and operated by GEANT NOC
- Several OF islands in NRENs facilities with Service Termination Points (STPs). Each of them constitute an administrative domain.
- Attached to NRENs, Last Mile research institutions, campus, etc. composed of several islands belonging to the same administrative domain (different SDN controllers but not privacy issues).

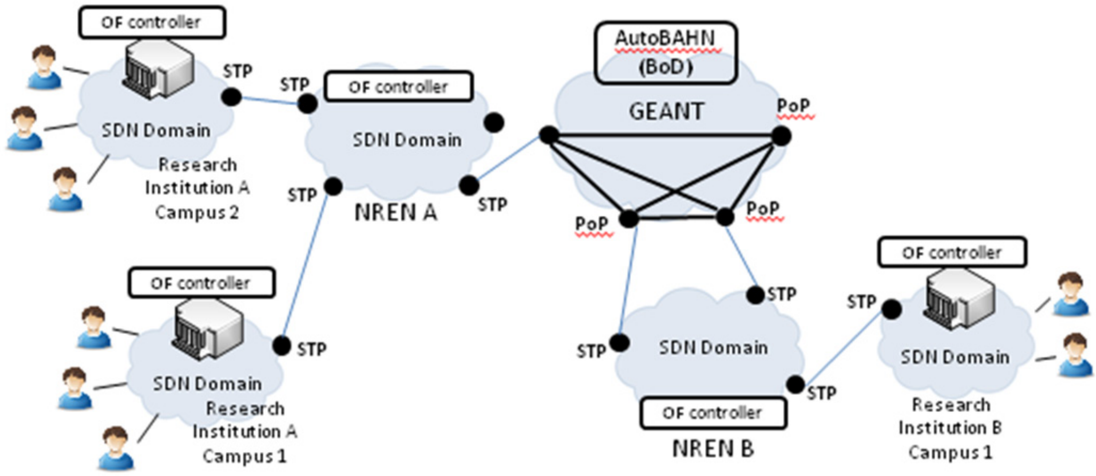

Figure 3. NREN/GEANT multi-domain scenario

OpenNaaS:

OpenNaaS is a smart framework to control and manage network services in general. As discussed, the specific usage that we would be giving to OpenNaaS here is Service orchestrator, Resource manager, and potentially some other extensions. As orchestrator, it can be seen as the entity that solves the gap between users (applications) and network, making use of several mechanisms.

For instance, OpenNaaS may be able to:

- attend to users´ requests for a multi-domain circuit request
- take care of isolating multi-domain provisioned circuits of different users.

On the other hand, a potential function that OpenNaaS may perform over the underlying layers is to (attending to users/application requests) trigger the mechanisms to configure the resources and provide the multi-domain connectivity service.

4.1. Orchestrator + NSI + SDN

In the complete slice-based scenario, the NSI management system manages the NSA. Hence, each domain's network provider must announce to the NSA the STPs that it will manage. Moreover, depending on the slicing mechanism we can, at the moment of the creation of the slice, specify which subset of the STPs belongs to that slice, or just use single STPs and let the slicing mechanism slice the traffic inside the domain.

Independent of the slicing mechanism inside each domain, the AG should aggregate the NSI messages of the PA agents of each slice/domain.

The slicing mechanism should provide the client isolation on the network as well as define the desired virtual topology that the client has requested; this can be done per domain, or per cosmos (whole set of domains). It should provide client isolation such that two clients are not able to interact between them (unless agreed by both). Technologies achieving such objectives are e.g. FlowVisor [17], OpenVirtex [18] or VeRTIGO [19].

The compute management system manages the computational capabilities assigned to each client slice.

Finally, the Orchestrator act as management integrating and coordinating previous management systems as services, in order to provide E2E multi-domain sliced-based services (virtual topologies, client isolation, and computational nodes). Figure 1 shows the integrated view of the proposed multi-domain SDN management framework.

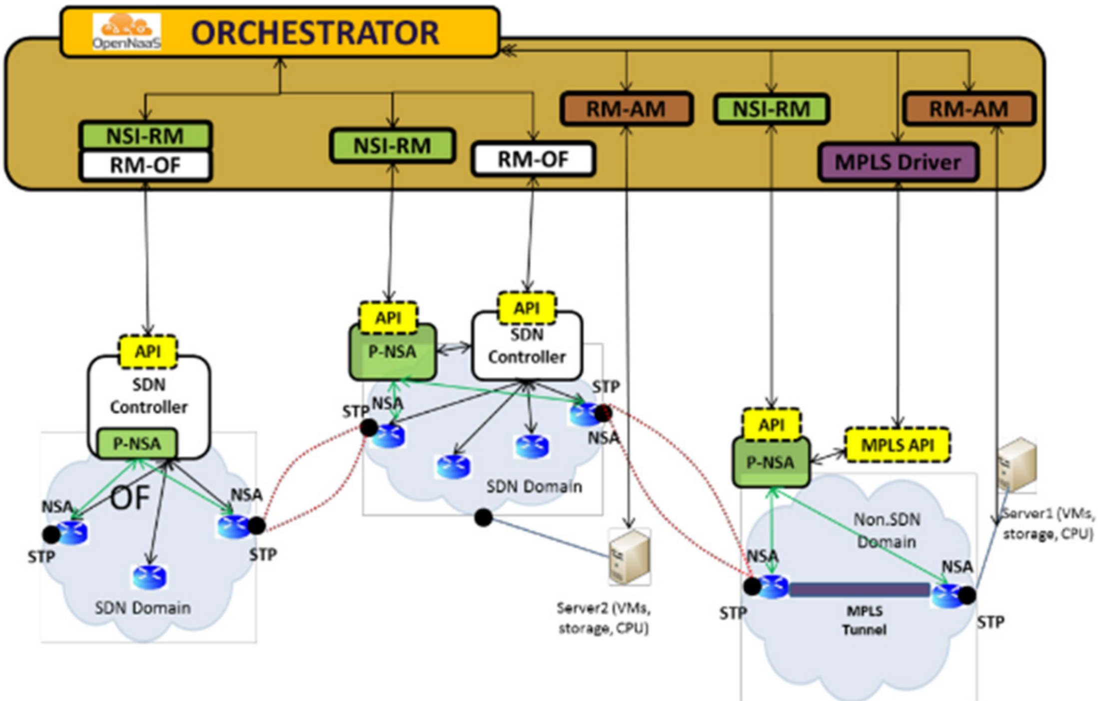

Figure 4. Multi-domain SDN management framework

5. Summary

The solutions proposed by the GN3Plus JRA2T1 task group are meant to fill a gap in the multi-domain network environment where the novel SDN concept is used. Research on multi-domain subject has been divided into two main areas of interest: connection-oriented and slice-oriented. For a connection-oriented case NSI-CS has been chosen as a promising multi-domain protocol. Moreover two possible solutions for the integration of NSI with SDN/OpenFlow has been proposed: the first one based on a new service type and the second one based on a custom namespace, where the latest looks promising as a potential overall solution. For a novel slice-oriented case, enabled by SDN/OpenFlow concept, new architecture has been described. Above studies can be a starting point for a further development process as well as can introduce changes in the multi-domain networking protocol in the form of NSI-CS. GN3plus was started in 2013 and further results are expected till Q2 2014.

Acknowledgements

This work is carried out under GN3plus project JRA2 activity.

References

[1] Open Networking Foundation, 2012. Software-Defined Networking: The New Norm for Networks.
[2] Open Networking Foundation, 2013. OpenFlow Switch Specification 1.4.0
[3] Felix. <http://www.ict-felix.eu/>
[4] XIFI. <https://www.fi-xifi.eu/home.html>
[5] Roberts, G., Kudoh, T., Monga, I., Sobieski, J., MacAuley, J. and Guok, C., 2014. Network Service Framework v2.0 (draft). Open Grid Forum.
[6] Roberts, G., Kudoh, T., Monga, I., Sobieski, J., MacAuley, J. and Guok, C., 2014. NSI Connection Service Protocol v2.0 (draft). Open Grid Forum.
[7] NOVI. http://www.fp7-novi.eu/
[8] OGF. NSI Connection Types v2.0 WSDL schema, https://code.google.com/p/ogf-nsi-project/source/browse/trunk/ConnectionService/ogf_nsi_connection_types_v2_0.xsd
[9] OGF. NSI Service Types v2.0 WSDL schema.
[10] OGF. NSI Connection Services P2P v2.0 WSDL schema. <https://code.google.com/p/ogf-nsi-project/source/browse/trunk/ConnectionService/ogf_nsi_services_p2p_v2_0.xsd>
[12] MacAuley, J., 2013. NSI-CS Service Decoupling. <https://redmine.ogf.org/dmsf_files/13101?download=>
[13] OpenNaaS. <http://www.opennaas.org/>
[14] SFA. <http://opensfa.info/doc/opensfa.html>
[15] GENI. <http://opensfa.info/doc/opensfa.html>
[16] GENI AM API <GENI Am aPI: http://groups.geni.net/geni/wiki/GAPI_AM_API_V3>
[17] Al-Shabibi, A. and Sherwood, S. Flowvisor. <https://github.com/OPENNETWORKINGLAB/flowvisor>
[18] ONLAB. OpenVirteX. <http://tools.onlab.us/ovx.html>
[19] R. D. Corin, R. D., Gerola, M., Riggio, R., Pellegrini, F. and Salvadori, E., 2012. VeRTIGO, Network Virtualization and Beyond. Software Defined Networking (EWSDN), 2012 European Workshop.

Author biographies

Eduardo Jacob (EHU) got in 1991 a BSc in Industrial Engineering and a MSc. in Industrial Communications and Electronics from the University of the Basque Country (UPV/EHU). He spent 2 years in a public R&D in Telecommunications enterprise which is nowadays Tecnalia. Later he spent some years as IT director in the private sector before returning to the Faculty of Engineering of Bilbao getting his PhD in ICT in 2001. He is assistant professor at the same Faculty, where is acting as Head of the Communications Engineering Department. He also leads the I2T research lab. He is the promoter and coordinator of the EHU OpenFlow Enabled Facility. His interests are related to application of the Software Defined Networks to support advanced applications, privacy and security in sensor based end to end communications systems and data privacy He is actually leading UPV/EHU's participation in GN3Plus and FP7 projects ALIEN, SECRET and UNIFY.

Alaitz Mendiola (EHU) received her BSc and MSc degrees in telecommunication engineering in 2012 from the University of the Basque Country (UPV/EHU). She is currently pursuing a second MSc in Information and Communication Systems in Wireless Networks at the UPV/EHU. She joined the I2T Research Group in 2010 and she has participated in several OpenFlow/SDN related projects. Her research interests include Software Defined Networks, Network Virtualization and DOCSIS access networks. She is involved in GN3+ JRA2T1 efforts.

Łukasz Podleski (PSNC) received his M.A. in Musicology in 2012 and B.Sc. degree in Computer Science in 2012 from the Adam Mickiewicz University in Poznan, Poland. He is a Computer Systems Analyst in Poznan Supercomputing and Networking Center. In the ADDONAS project he works on technologies for improving last mile network utilizing OpenFlow 1.0 protocol. Currently he is working within the JRA2T1 activity of the GN3plus on the applying SDN concept onto multi-domain and in the NSI-CONTEST on test suite of the NSI v2.0 protocol framework. His main interests are in advanced networking technologies (especially Software Defined Networking), solutions for multi-domain environments and Software Engineering. He is an ANSI C/C++/Java architect and developer with experience in enterprise-class real-time systems for industry.

Radosław Krzywania (PSNC) received the M.Sc. degree in Computer Science – Software Engineering from the Poznan University of Technology in 2003. He is working in Poznan Supercomputing and Networking Center as a senior network engineer. He participated in several FP6 IST projects: 6NET (IST-2001-32063), PHOSPHORUS (IST034115) and GN2 (IST511082). He also participated in a number of national initiatives funded by Polish Ministry of Science and Higher Education (e.g. Clusterix). Currently he is involved in the national project "Engineering of Future Internet" and FP7 project GN3 (Project no. 238875). The main experience is Bandwidth on Demand services, network control planes, and network management. He is author or co-author of papers in professional journals and conference proceedings.

Miłosz Przywecki (PSNC) received a M.Sc. degree in Electronics and Telecommunications from Poznan University of Technology in 2003 and joined the Network Division of Poznan Supercomputing and Networking Centre as a Networking Systems Analyst in 2005. He participated in a number of European networking projects (Porta Optica Study, GÉANT2/3, PHOSPHORUS, ADDONAS). His main interests are in advanced networking technologies, network protocols and services, particularly focused on SDN technologies. He is GN3plus JRA2T1 "OpenFlowOpenFlow/SDN for Specialised Applications" task leader.

Krzysztof Dombek (PSNC) received the M.Sc. degree in 2008 from the University of Technology and Life Sciences in Bydgoszcz, Poland. He's a Computer System Analyst in Polish NREN – Poznan Supercomputing and Networking Center. He was a participant of many European research projects e.g. PHOSPHORUS, GEANT3 or GEYSERS. He was also involved in national project of the Future Internet Engineering. At present he's a participant of ALIEN project founded by EU in FP7 where he's involved in integration with the OFELIA Control Framework.

Artur Juszczyk (PSNC) received the M.Sc. degree in the Electrical Engineering, majoring in the Microprocessors and Control Systems, at the Poznan University of Technology in 2005. He works in the Poznan Supercomputing and Networking Center since 2007 as a computer systems analyst. He took part in the several European and national projects, PHOSPHORUS, GEYSERS, GEANT and Future Internet Engineering as a programmer, tester, system engineer. He is currently involved in the international project ALIEN. His interests focus on the Unix base systems, virtualization of resources, management of network infrastructure and next-generation networks.

Xavier Jeannin (RENATER) was formerly activity manager of the network activity in the European Grid project EGEE. He is in charge of LHCONE deployment in France, an international L3VPN that currently connect the LHC computing center over the world. He is the task leader of MP-VPN in GN3-plus project (SA3T3) that is in charge of the design, service specification and deployment of Multi-Domain VPN service in Europe. He is member of JRA2T1 (OpenFlow/SDN for Specialised Applications) task within GN3+.

José Ignacio Aznar-Baranda (i2CAT) received the Telecommunication Engineering and Msc degrees in 2008 and 2010, respectively, both from the Superior Politechnique Center at the University of Zaragoza (UZ), Zaragoza, Spain. He joined i2CAT Foundation's in 2013 and he is involved in several GN3+ activities and FI-PPP XIFI projects mainly related to network virtualization and infrastructure as a service (IaaS) topics.

Albert Vico-Oton (i2CAT) holds a Ph. D. in Computer Science by the University Rovira i Virigili. In January 2013 he joined i2CAT Foundation to lead the Future Internet Testbed research line. At present, he works in the OFELIA-FP7 project leading the work package in charge of the development of the control framework for the OFELIA testbeds. He also, joined the Fed4FIRE project where his main objective is to allow i2cat testbed Experimenta to be federated with the rest of testbeds. Furthermore, he works on the FIBRE-BR-EU project working within the package aiming to build the European facility within FIBRE. His current research interests are related to SDN, network virtualization, distributed storage, network coding and future internet testbeds.

Kurt Baumann (SWITCH) received a Master Degree in Mathematics of the University Of Zurich (UZH) in 2001. In 2002 he passed the IBM trainee program. Afterwards he worked in a position of a security officer and customer engineer for IT-Infrastructures projects in the strategy-outsourcing department at IBM. In 2005 he joined SWITCH as a member of the middle ware group in a position of the Project leader of SWITCHconnect. Today he is a member of the Peta Solutions Department at SWITCH with focus of network research support. He is actively participating in R&D projects in Wireless Mesh Networks, FEDERICA, cloud computing (Swiss Academic Cloud) and is task leader of the GN3 JRA2T5, Network Factory.

Christos Argyropoulos (GRNET) received his Diploma in Electrical Engineering and Computer Science from the Polytechnic School of University of Patras. He is a PhD Candidate at the ECE department of the National Technical University of Athens (NTUA) and a member of the Network Management and Optimal Design Laboratory (NETMODE) since 2008. He has been a teaching assistant in the course "Network Management and Intelligent Networks". He has been participating in EFIPSANS, NOVI, GÉANT (GN3 and GN3plus) European FP7 projects. His main research interests lie in the area of computer networks with emphasis on network virtualization and software defined networking.