# Network Virtualization over Heterogeneous Federated Infrastructures: Data Plane Connectivity

Christos Argyropoulos[1], Georgios Androulidakis[1], Dimitris Kalogeras[1], Błazej Pietrzak[2], Bartosz Belter[2], Leonidas Lymberopoulos[1] and Vasilis Maglaris[1]

[1] Network Management & Optimal Design Laboratory (NETMODE)
School of Electrical & Computer Engineering
National Technical University of Athens (NTUA), Greece
{cargious, gandr, dkalo, leonidas, maglaris}@netmode.ece.ntua.gr

[2] Poznan Supercomputing and Networking Center (PSNC)
Poznan, Poland
{blazej.pietrzak, bartosz.belter}@man.poznan.pl

*Abstract*—**Federation of Future Internet (FI) infrastructures is an emerging requirement of the FI research community giving the ability to experimenters to use a variety of virtual appliances spread across different virtualized platforms. Federated architecture implementations, such as the Slice-based Federation Architecture (SFA), are capable of providing a unified way of interaction across domains but do not address the data plane interconnection problem, i.e. one virtual entity in one domain connecting at the data-link (Ethernet) layer with another virtual entity in a remote domain. This paper presents a novel scheme for interconnecting heterogeneous virtualized infrastructures at the data plane that considers concurrence, isolation, elasticity and programmability aspects. We tested and assessed our scheme using a software switch prototype that interconnects resources drawn from PlanetLab and FEDERICA virtualization infrastructures. Performance evaluation measurements indicate that our proposed stitching approach introduces minimal degradation on end-to-end delays and bandwidth between virtual entities, compared to physical (substrate) entities.**

***Keywords--Network virtualization; heterogeneous federated infrastructures; Future Internet; data plane connectivity; PlanetLab; VINI; FEDERICA***

## I. INTRODUCTION

A key factor for the Future Internet (FI) success is the combination of computing and networking virtualized infrastructures into a federated framework. In this spirit, the Future Internet emerges as a complex ecosystem, offering holistic services to users over shared federated interconnected platforms. Towards developing such combined environments, novel architectures, protocols and applications need to be tested and validated in realistic testbeds that have been deployed worldwide for use of experimental FI research.

Specifically, FI infrastructures should cater for dynamic provisioning and interconnection of resources drawn from independent yet loosely coupled heterogeneous infrastructures as elaborated in [1]. Earlier efforts to that direction include VINI – Virtual Network Infrastructure [2], PlanetLab [3], Emulab [4] and ProtoGENI [5] infrastructures in the US (GENI NSF initiative [6]) and FEDERICA [7], PanLab [8] and OneLab [9] in Europe (FIRE EC initiative [10]).

Current federation efforts concentrate on development of tools and APIs to federate virtualized infrastructures for research and experimentation for the Future Internet. The first effort is the Slice-based Federation Architecture (SFA) [11], adopted by both the GENI in the US and the OneLab/OneLab2 EC project and the second one is the Teagle [12] architecture employed within the European PanLab testbed.

In this paper we focus on the network connectivity aspects of heterogeneous federated infrastructures. The existence of unified architectures across federated domains, such as SFA, is capable of providing a homogenized way of interaction between the federated domains but does not solve the problem of the data plane stitching. The use of virtual resources in a networked environment requires techniques that can provide the creation of virtual networks across resources. Virtual networks should provide the typical network services that one can find in a network environment (Layer 2/Layer 3 connectivity plus services like DNS, DHCP) enriched with some additional characteristics such as elasticity (up-scaling/downscaling) and migration capabilities (rearrangement of resources and reconstruction of arbitrary network topologies over different physical infrastructures). The problem of providing virtual networks becomes more challenging with the existence of heterogeneous network resources that need to be managed and configured within the federated environment.

The main objective of our work is the instantiation of virtual networks consisting of virtual interfaces, virtual links, virtual routers and virtual switches, deployed across federated domains. We chose to consider porting of particular open-source software components, such as Open vSwitch [13], rather than developing software from scratch. Our aim is not only to develop software that provides network functionalities but to design and implement the techniques that can make these functionalities interoperable and adoptable in different virtualized environments.

This work is motivated by ongoing research in the FIRE FP7 STREP project NOVI – Network Innovations over Virtualized Infrastructures [1] [14]. NOVI aspires to develop a framework (information model, management plane tools and algorithms) that will empower FI users to discover, view,

978-3-901882-50-0

monitor, control and provision virtualized networking resources within a federated networking substrate, thus complementing their distributed storage and computing service baskets with interconnection-specific resources. Our proof of concept validation is based on the realization of the data plane stitching of a private PlanetLab instance with resources interconnected over the Internet and FEDERICA, an infrastructure of virtual resources interconnected via dedicated networking facilities of European National Research & Education Networks and GÉANT [15].

NOVI experimental research was tested on these two heterogeneous platforms since they represent basic architectural profiles: FEDERICA is based on controlled connectivity and virtualization tools commonly employed in private data centers (VMWare) while PlanetLab can be seen as a simplistic model of public cloud facilities. A combined slice would be anticipated by data center users complementing their slices on demand using public cloud facilities.

The remainder of the paper is organized as follows. In Section II we present an overview of past and on-going research activities which deal with network virtualization aspects of FI platforms. In Section III we explain the motivation and contribution of our data plane stitching approach of federated FI infrastructures. Section IV describes in detail the network aspects of PlanetLab/VINI and FEDERICA platforms (the NOVI case), while Section V analyzes our proposed mechanism for data plane stitching of federated heterogeneous virtualized infrastructures. In Section VI we present performance evaluation measurements of the proposed stitching mechanism and we conclude in Section VII, trying to define future directions of network virtualization inside FI ecosystem.

## II. RELATED WORK

One of the first large-scale collaborative environments that was deployed in 2002 is PlanetLab [16]. Its initial scope was to allow experimentation and testing of multiple services concurrently and continuously, each deployed in its own isolated slice [17]. The slice is defined as a horizontal cut of global PlanetLab virtual resources (slivers). The PlanetLab platform, through the VNET module [18], ensured slice isolation via a restricted form of raw IP and raw packet sockets. VNET does not enable the experimenter/user to create specific overlay topologies, as it relies on legacy internet protocols (e.g. BGP) that cannot be overridden by the user.

The ability to create controlled network profiles within large-scale collaborative environments came with VINI [2]. Network capabilities of PlanetLab (PL) were extended in PL-VINI that used: (i) a modified version of Linux TUN/TAP module, (ii) the Click modular software router [19], (iii) XORP routing protocol suite [20], (iv) OpenVPN [21], and (v) UDP tunnels for creating point-to-point links in the overlay network. The aforementioned components allowed conducting controlled experiments that evaluate the existing IP routing protocols and forwarding mechanisms under realistic conditions, providing a virtual network infrastructure [22].

After the PL-VINI implementation on the VINI infrastructure a set of extensions to the PlanetLab kernel and tools, Trellis, was implemented and deployed to the same experimental environment [2] [23]. Trellis was structured on two container-based virtualization technologies: (i) VServer [24], (ii) NetNS [25] and (iii) an Ethernet over GRE [26] mechanism, trying to provide high-speed virtual network capabilities to the VINI platform.

The Emulab [4] platform supports network experiments in an emulated environment providing researchers with a large-scale testbed experimentation using formerly PlanetLab facility and lately ProtoGENI (see below) [5]. It also includes IEEE 802.11 wireless nodes and software defined radio devices. Emulab consists of a set of nodes, the operating system and software running on those nodes, links (802.1Q VLANs are used for traffic isolation between different experiments) connecting the nodes, traffic shaping attributes of those links, network routing information, and runtime dynamics such as traffic generation [28]. Ethernet frame forwarding is handled from high-end commodity switches [29].

ProtoGENI [5] is a large-scale integration effort of existing and under-construction systems controlled by GENI. The integration consists of a high-speed backbone on Internet2's wave infrastructure and a set of sliceable network resources embedded in this backbone. An enhanced version of the Emulab management software, plus additional software from PlanetLab and VINI are used to manage the backbone and to provide a common point of integration between them. Researchers will be able to select the topology of VLANs to run on top of the infrastructure.

A different approach to network virtualization was taken by the FEDERICA platform [7] which focused on experiments in a controlled environment: Instead of using legacy Internet protocols, it relies on L2/VLAN technology and Juniper Logical Routers, connected via SDH/SONET 1Gbps circuits. Notably, it uses programmable high-end routers (Juniper MX-480) [27] and switches (Juniper EX3200) for the network resources and VMware ESXi 3.5 for the computing resources. FEDERICA's management plane is capable of partitioning the different devices that compose the physical substrate, using the virtualization capabilities supported by the individual resources, by means of their corresponding APIs.

None of the above platforms, however, cover data plane interconnection between heterogeneous virtual infrastructures. Moreover, there is no management plane mechanism that can handle heterogeneous network technologies.

## III. MOTIVATION AND DESIGN

Virtual networks attempt to efficiently utilize networking infrastructures by reusing individual physical or logical networking resources for multiple concurrent network instances, or to aggregate multiple such resources to obtain increased capabilities [30]. The key properties/ capabilities of virtual networks that were taken into account in our effort are summarized in the following points:

*a)* ***Concurrence****:* Network elements should be capable of being used concurrently by multiple virtual network instances

*b)* ***Isolation****:* Clear separation (independency without interference) between virtual networks must exist *c)* ***Abstraction****:* Virtual resources (hosts, links, routers, switches, etc.) may not directly correspond to physical or logical components resources (aggregation / resource sharing)

*d)* ***Elasticity:*** Flexible reorganization of resources (up-scaling/ downscaling)

*e)* ***Migration:*** Rearrangement of resource usage and reconstruction of arbitrary network topologies over physical network topologies

*f)* ***Programmability:*** Ability to develop procedures and tools in order to handle different formats, network protocols, etc.

Our motivation was to provide a novel scheme for interconnecting virtualized infrastructures at the data plane. The key points of our design are:

*a)* ***Federation of heterogeneous virtual infrastructures at the data plane:*** Applicable to existing heterogeneous infrastructures in order to connect their data planes (our experimental approach includes the federation of platforms, such as PlanetLab, VINI and FEDERICA, with different virtualization techniques, implemented in a large-scale environment).

*b)* ***Network virtualization layer agnostic mechanism:*** Connection of L3-L2 (i.e. PlanetLab - FEDERICA), L2-L2 (i.e. VINI/TRELLIS - FEDERICA) or L3-L3 platforms is applicable.

*c)* ***Configuration of inter-domain multi-point connection points pertaining to platform and slice view points:*** Ability to connect the federated data planes to multiple points creating complex topologies and providing a proper substrate for network engineering (load balancing, fail-over, etc.).

*d)* ***Use of Generic Routing Encapsulation tunnels (GRE over the Internet) and 802.1Q VLANs:*** Combination of GRE protocol capability to encapsulate IP packets or Ethernet frames over IP networks and 802.1Q VLAN wide support to network infrastructures, plus the inherent mechanisms they provide for data flow separation consist a convenient solution.

*e)* ***Use of software programmable switch (Open vSwitch):*** Open vSwitch capability to bind with physical interfaces (like Ethernet interfaces) and logical interfaces like TUN/TAP, GRE over IP, GRE over IPSEC enables a dynamic configurable L2 forwarding plane.

## IV. STITCHING FI INFRASTRUCTURES – THE NOVI CASE

NOVI established an experimental testbed that required data-plane connectivity of FEDERICA and PlanetLab/VINI platforms. This leads towards an environment consisting with holistic network virtualization properties, based on stitching technologies implemented in each of the aforementioned platforms. Provisioning workflow and management mechanisms of heterogeneous federated infrastructures are implemented on the NOVI management plane by a specific component, namely the NSwitch Service. In this paper, however, we focus on data plane interconnection challenges.

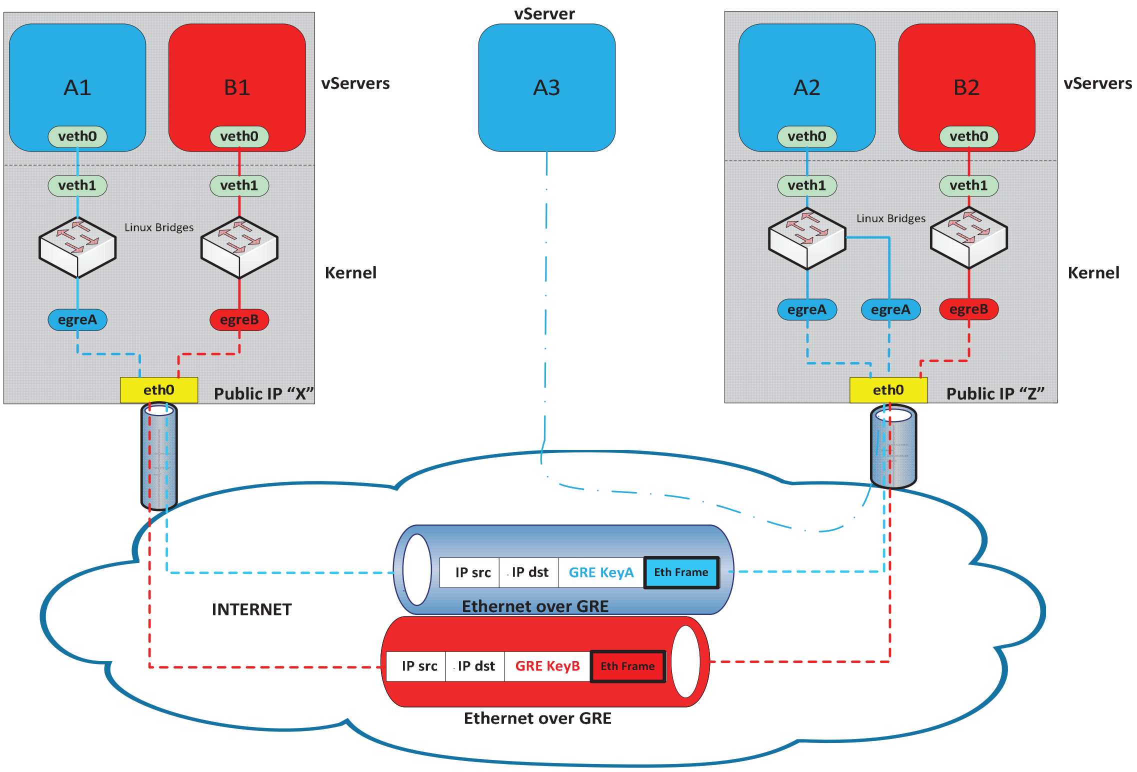

**Figure 1**. L2 connectivity among container-based virtualized hosts over the Internet, using Ethernet over GRE tunnels and Linux bridges as S/W switches (VINI/Trellis architecture)

### A. *PlanetLab/VINI*

In the PlanetLab/VINI domain GRE tunnels are used over the Internet to create virtual links between virtual hosts (slivers). Ethernet framing over GRE tunneling was chosen as it has a small, fixed encapsulation overhead and allows multiple layer network virtualization. A specific field of GRE header, named *key field* [32], makes GRE protocol a proper solution for point-to-point multiple virtual links. The *key field* contains a four octet number which is inserted by the encapsulator (incoming end-point of Ethernet frames). The *key field* provides the context for multiple traffic flow separation between encapsulator and decapsulator (outgoing end-point of Ethernet frames). Packets belonging to a traffic flow are encapsulated using the same *key value* and the decapsulating tunnel endpoint identifies packets belonging to a traffic flow based on the *key field* value. Suppose we have two nodes that host multiple virtual nodes and a single public IP address bound per node (PlanetLab case), the creation of multiple virtual links between them is based on traffic flow separation. Every logical traffic flow, equivalent to virtual link, is defined by the tuple {source public IP, destination public IP, key field}.

An example is given in Figure 1. Each of the two physical nodes (grey square) that runs a Linux-based operating system hosts two virtual nodes. The ability of Linux bridge to act as a MAC learning switch with multiple interfaces permits to a third virtual node (A3), to be connected to the same broadcast domain with the other two virtual nodes (A1, A2) that participate to the same slice (slice A consists of A1,A2,A3 virtual nodes). At the center of the Figure 1, we can see an abstraction of two IP packets that belong to different traffic flows. Despite their identical endpoints they can be delivered to the appropriate Ethernet over GRE logical interface inside the Linux kernel according to their different GRE keys (*keyA* for slice A, *keyB* for slice B). Therefore the assignment of a unique GRE key per slice provides the mechanism for traffic flow separation on the root context of physical nodes. In this way, conformance to concurrence, isolation and abstraction requirement can be accomplished.

### B. *FEDERICA*

FEDERICA [33] network infrastructure is geographically dispersed to 13 physical sites. Four of these sites are equipped with Juniper MX-480 high-end routers and linked in a full-mesh. The 9 non-core nodes are equipped with Juniper EX-3200 switches. The 4 core nodes are equipped with 2 servers running VMware ESXi 3.5 and the non-core with 1 server. VMware ESXi 3.5 bare-metal OS is a full virtualization platform. Therefore, each virtual host can run its own independent OS ignoring that the resources it handles are virtual. The Juniper MX480 is an Ethernet-optimized edge router that provides both switching and carrier-class Ethernet routing. Juniper's Operating System (JUNOS ®) that runs on MX-480 provides logical router capabilities. These capabilities allow partitioning of a single physical router into multiple logical instances, where each logical instance performs independent routing tasks. The data plane connectivity between VMware virtual nodes and Juniper routers and switches outside the server boxes is provided by the VMware vSwitch (VMware's software switch implementation) [34]. VLANs are used to connect slice resources and create the required topology. Figure 2 shows two separate slices (a blue & a red slice). Each one includes one logical router and two virtual machines which reside into different ESXi hypervisors. Separate VLANs are used for traffic isolation between slices.

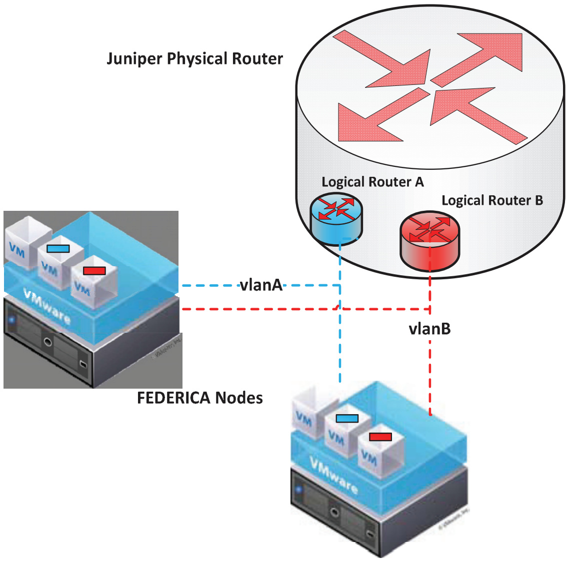

**Figure 2.** L2 connectivity among VMware fully virtualized hosts and a Juniper MX-480 logical router (part of FEDERICA architecture)

## V. DATA PLANE STITCHING

In Figure 3 we depict NOVI's L2 stitching approach towards combining slivers from both platforms; i.e. combining the blue slice in Figure 1 and the blue slice in Figure 2 into a consistent network topology, without sacrificing slice isolation.

Our stitching approach is based on introducing the NSwitch component, a software programmable switch, based on Open vSwitch [13]. Performance-wise the Open vSwitch choice is similar to the Ethernet bridge implementation in the Linux kernel [35], nonetheless it was chosen due to its ease of deployment and its built-in capabilities required for our purposes, e.g. 802.1Q VLAN, and support of multiple tunneling protocols (Ethernet over GRE, IPsec, GRE over IPsec).

Open vSwitch limits disruption by using existing hooks into the kernel, therefore it can be deployed as a module without requiring any modification to the kernel. An additional feature compared to the Linux kernel Ethernet bridge that was mandatory for our NSwitch implementation was the ability to include a GRE logical interface to the same broadcast domain with VLAN logical interfaces. By adding a VLAN ID (VID) to GRE logical interfaces we create a unified L2 broadcast domain that includes network resources from two or more discrete heterogeneous federated infrastructures.

To create a multi-tenant environment over the unified heterogeneous network resources of the federated testbeds, we need to create a mapping mechanism between network resources of testbeds A and B. PlanetLab, VINI/Trellis and FEDERICA platforms correlate each slice of resources with at least one user.

In PlanetLab-based platforms we decided to use the GRE key as the network resource id in order to create a multi-tenant environment. As we described in Section IV, the GRE key may be used for data flow separation. The range of values that can hold (key field is a 32-bit integer) is wide enough to cover the network partitioning needs of large-scale environments. Recently, PlanetLab platform adopted and integrated GRE tunnel capabilities that were developed in VINI/Trellis. We make use of these capabilities in order to construct the required network control plane mechanism that combines the network resources of the federated testbeds.

In the FEDERICA case, where the VID is used as network resource id the maximum value of 4095 is enough to cover the number of experimenters' slices, but in a data center that hosts cloud computing facilities (e.g. IaaS) one might need thousands of VLANs to partition the traffic according to the specific group that the VM may belong to. The current VLAN limit is inadequate in such situations.

We decided to define the network resource id based on the range of the GRE key values. Values from 1 to 4095 can be used in order to define compound slices that include network resources from both VLAN-based and GRE-based testbed infrastructures. Values above 4095 can be used in order to create network slices only from GRE-based testbed infrastructures. Hence we permit data plane stitching of federated heterogeneous infrastructures without introducing a possible limitation of the number of slices that one testbed may have to the whole federation.

For the deployment of the compound broadcast domains one Open vSwitch bridge instance can be used as we exploit the VID built-in concept for broadcast domains logical separation. In case that VIDs inside the federation are exhausted and we want to make use of NSwitch component as a switch for LAN segments of GRE-enabled testbed infrastructures one Open vSwitch instance per slice/GRE key should be used.

To the best of our knowledge L2 network devices may use the VID as broadcast domain logical separator but they do not support the same functionality with use of the GRE key. That is the case for the Juniper MX480 routers that belong to the FEDERICA testbed. GRE tunnels created on these routers are represented as tunnel interfaces. However, these interfaces do not have the full functionality of their physical counterparts. For instance it is not possible to tag the tunnel interface with VLAN ID. This limitation prevents from creating a unified L2 broadcast domain containing resources from federated infrastructures using Juniper MX480 features and opens the need for the NSwitch, a software programmable switch.

In Figure 3 the transformation functionality between a GRE and a VLAN interface is presented inside the box (left corner). Two compound slices (blue & red) have been configured. egreA/B and vlanA/B values belong to the 1-4095 range that is used for such slices. Two different logical routers have been created in FEDERICA side. Different virtualized hosts that belong to the two slices have L2 connectivity, through the NSwitch component, with the Logical Routers. Depending on the Logical Router configuration (L2 or L3 forwarding function) virtual hosts from the two testbeds may belong to the same Local Area Network (LAN) or to two separate LANs.

## VI. PERFORMANCE EVALUATION

In this section we study and evaluate the performance of the

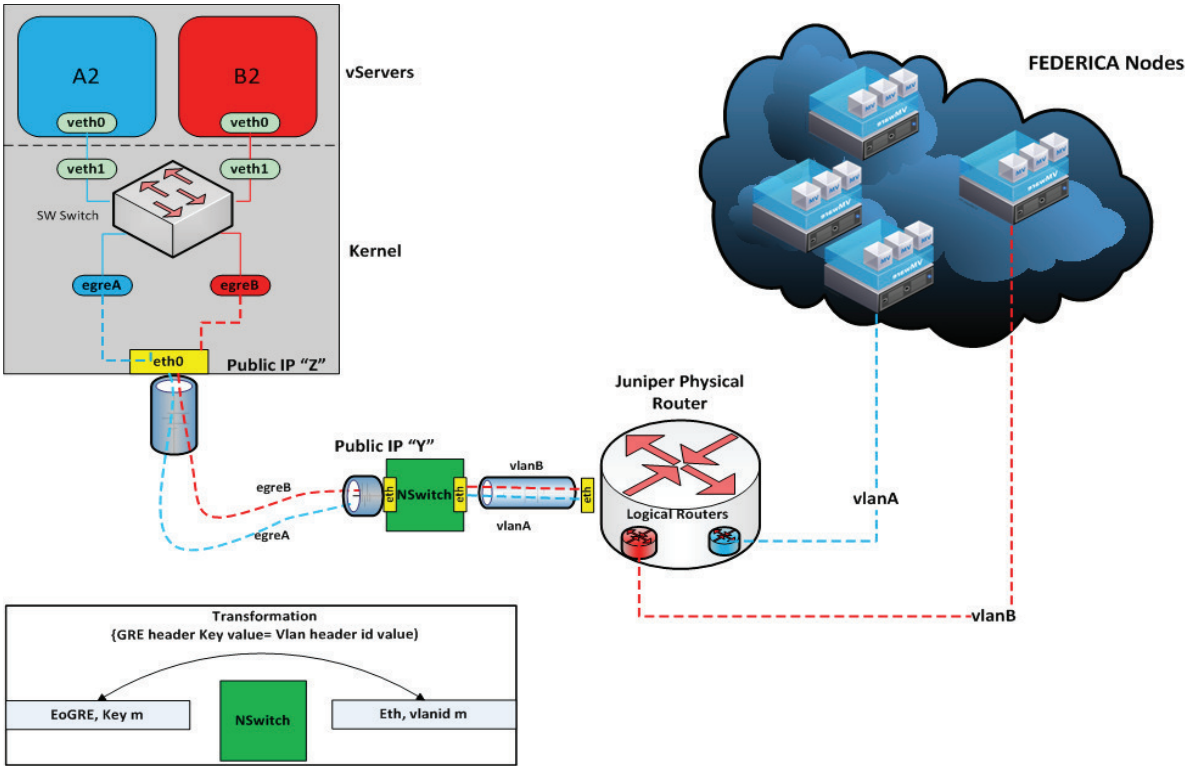

**Figure 3.** L2 network federation example of heterogeneous virtual infrastructures (GRE & 802.1q protocols are used for L2 topology creation)

proposed data plane stitching mechanism. The presented results and their corresponding observations are based on real measurements conducted in NOVI's testbed environment. Our experimentation environment includes three Point of Presence (PoPs), namely the National Technical University of Athens (NTUA) PoP in Greece, the Poznań Supercomputing and Networking Center (PSNC) PoP in Poland and the German National Research and Education Network (DFN) PoP in Germany. More specifically, we created an experimental slice that includes both PlanetLab and FEDERICA resources: (i) a PlanetLab sliver instantiated within a PlanetLab host at NTUA; (ii) a FEDERICA Logical Router at PSNC; (iii) a VMWare Virtual Machine (VM) located at PSNC; (iv) a Logical Router at DFN and (v) a Virtual machine (VM) at DFN. Logical Routers are instantiated within FEDERICA Juniper MX480 and VMs are instantiated inside VMware ESXi servers. Their interconnection is depicted in Figure 4.

In our experiments, to gain some insight of the performance of our proposed stitching mechanism, we compare the mean RTT (Round-Trip Time) and the maximum bandwidth between virtual resources (Sliver, Logical Router, VM), compared to equivalent values between the physical resources embedding them (PlanetLab host, FEDERICA Physical Router, VMware ESXi Server).

In the first row of Table I, we present the mean RTT values between physical resources i.e the PlanetLab Host <NTUA PL Host> and (i) FEDERICA PSNC Physical Router <PSNC PR>, (ii) FEDERICA PSNC VMware ESXi Server <PSNC HV>, (iii) FEDERICA DFN Physical Router <DFN PR>; and, (iv) FEDERICA PSNC VMware ESXi Server <DFN HV>. In the second row of Table I, we provide the corresponding values for the virtual resources (Sliver, Logical Routers, VMs). As we can observe, there is minor performance degradation in the RTT values between the physical and virtual resources. For example, the mean RTT value between <NTUA PL Sliver> and <PSNC LR> increases from 83.4 (physical resources) to 83.6 msec, thus introducing 0.2% performance degradation with respect to the Round-Trip Time. The same pattern is observed for all RTT values between the virtual resources with the increase in RTT values ranging from 0.1% - 0.2%.

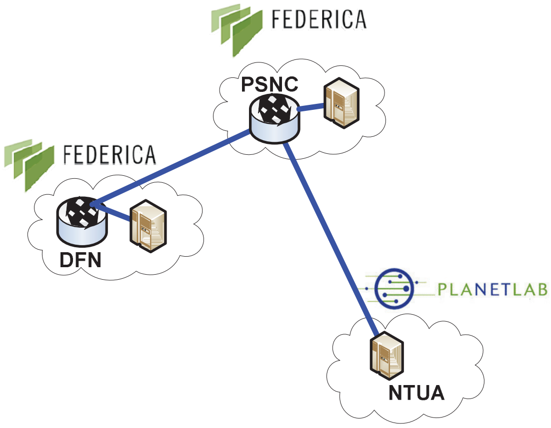

**Figure 4.** Slice topology used for performance evaluation measurements

TABLE I. ROUND-TRIP TIME BETWEEN PHYSICAL AND VIRTUAL RESOURCES

| RTT (ms) | PSNC PR | PSNC LR | PSNC HV | PSNC VM | DFN PR | DFN LR | DFN HV | DFN VM |
|---|---|---|---|---|---|---|---|---|
| **NTUA PL Host** | 83.4 | - | 83.8 | - | 116.7 | - | 117.1 | - |
| **NTUA PL Sliver** | - | 83.6 (-0.2%) | - | 84.0 (-0.2%) | - | 116.8 (-0.1%) | - | 117.3 (-0.2%) |

TABLE II. MAXIMUM BANDWIDTH BETWEEN PHYSICAL AND VIRTUAL RESOURCES

| Bandwidth (Mbps) | NSwitch (PSNC) | | PSNC VM | | DFN VM | |
|---|---|---|---|---|---|---|
| | TCP | UDP | TCP | UDP | TCP | UDP |
| **NTUA PL Host** | 81.5 | 95.1 | - | - | - | - |
| **NTUA PL Sliver** | - | - | 76.8 (-5.8%) | 89.8 (-5.6%) | 72.4 (-11.2%) | 84.9 (-10.7%) |

In Table II we present the maximum bandwidth values between the physical and virtual resources with respect to TCP and UDP traffic. Due to the fact that we cannot measure the bandwidth inside the VMware ESXi Hypervisor, we use the NSwitch component at PSNC to compare the maximum bandwidth values within the VMs residing at PSNC and DFN. As we can observe from Table II, the maximum bandwidth has decreased by 5.8% and 5.6% for the VM at PSNC in terms of TCP and UDP traffic respectively. However, taking into account that the maximum bandwidth decreases due to the fact that the measurements involve virtual machines (and not physical nodes), the actual decrease in bandwidth, with respect to network stitching, is less than the ones in Table II. Finally, the further deterioration of the bandwidth of the VM at DFN is attributed to the fact that it is compared towards the NSwitch which is located at PSNC.

## VII. CONCLUSIONS & FUTURE WORK

This paper presented a data plane stitching mechanism for federated heterogeneous virtual infrastructures. Our mechanism is network virtualization layer agnostic and may function in multi-tenant environments which demand isolation, concurrence, programmability and elasticity. In terms of scalability the heterogeneity of our mechanism does not introduce limitations as far as the total number of concurrent network slices is concerned, and does not introduce any significant performance degradation on end-to-end delays and bandwidth between virtual entities, compared to physical (substrate) entities. In addition, the use of well known and widely supported network protocols (GRE tunnels and IEEE 802.1Q VLANs) ensures the compatibility with a large number of FI infrastructures.

The work reported in this paper motivates further extensions and research on network virtualization issues in federated environments. We plan to create the required control mechanisms for Virtual Machine network profiling, including state information, i.e. switch port, L2 learning table entries, L3 forwarding state, ACLs, QoS policy and monitoring configuration.

Another future direction of our work is the possible exploitation of protocols described in Internet drafts such as VXLAN [36] and NVGRE [37]. These protocols aim to engineer multi-tenancy network substrate in a combined private/public cloud environment and solve scalability issues to L2 broadcast domain (L2 traffic partitioning problem described in Section V and [36]). Last but not least, our stitching scheme based on Open vSwitch can easily extend to an OpenFlow [38] environment empowering experimenters with flow-based programmability within slices spanning across heterogeneous virtualization platforms.

## ACKNOWLEDGMENT

This work was partially supported by the European Commission, 7th Framework Programme for Research and Technological Development, Future Internet Research & Experimentation (FIRE), Grant No. 257867 - NOVI.

## REFERENCES

[1] Lymberopoulos, L., Grosso, P., Papagianni, C., Kalogeras, D., Androulidakis, G., van der Ham, J., de Laat, C., Maglaris, V., "Managing federations of virtualized infrastructures: A semantic-aware policy based approach", in Proc. of Integrated Network Management Symposium , pp.1235-1242, 2011.

[2] VINI – A VIrtual Network Infrastructure, http://www.vini-veritas.net

[3] PlanetLab, http://www.planet-lab.org

[4] Emulab, http://www.emulab.net

[5] ProtoGENI - Prototype implementation and deployment of Global Environment for Network Innovations, http://www.protogeni.net

[6] GENI - Global Environment for Network Innovations, http://www.geni.net

[7] FEDERICA – Federated E-infrastructure Dedicated to European Researchers Innovating in Computing network Architectures, http://www.fp7-federica.eu

[8] PanLab, http://www.panlab.net

[9] OneLab, http://www.onelab.eu

[10] FIRE – Future Internet Research & Experimentation, http://cordis.europa.eu/fp7/ict/fire

[11] Slice Federation Architecture (SFA), http://groups.geni.net/geni/wiki/SliceFedArch

[12] Teagle, http://www.fire-teagle.org

[13] Open vSwitch, http://www.openvswitch.org

[14] NOVI – Network Innovation over Virtualized Infrastructures FP7 STREP Project, http://www.fp7-novi.eu

[15] GÉANT, http://www.geant.net/pages/home.aspx

[16] PlanetLab history, http://www.planet-lab.org/history

[17] Chun, B., Culler, D., Roscoe, Bavier, A., Peterson, L., Wawrzoniak, M., Bowman M., "PlanetLab: an overlay testbed for broad-coverage services", SIGCOMM Comput. Commun. Rev. 33, 3, 2003.

[18] Huang, M., "VNET: PlanetLab Virtualized Network Access" , Tech. Rep. PDN–05–029, PlanetLab Consortium, 2005.

[19] Kohler, E., Morris, R., Chen, B., Jannotti, J., Kaashoek M. F., "The Click modular router", ACM Transactions on Computer Systems, vol. 18, pp. 263–297, 2000.

[20] Handley, M., Kohler, E., Ghosh, A., Hodson, O., Radoslavov, P., "Designing extensible IP router software" , Proc. Networked Systems Design and Implementation , 2005.

[21] OpenVPN, https://www.openvpn.net

[22] Bavier, A., Feamster, N., Huang, M. , Peterson, L. , Rexford J., "In VINI veritas: realistic and controlled network experimentation", in Proc. of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications, September, Pisa, Italy, 2006.

[23] Bhatia, S., Motiwala, M., Muhlbauer, W., Mundada, Y., Valancius, V., Bavier, A., Feamster, N., Peterson, L., Rexford, J., "Trellis: a platform for building flexible, fast virtual networks on commodity hardware", in Proceedings of the ACM CoNEXT, 2008 Conference (CoNEXT '08). ACM, New York, NY, USA, Article 72.

[24] Linux-VServer, http://linux-vserver.org

[25] LXC - The userspace control package for Linux Containers, http://lxc.sourceforge.net

[26] Farinacci, D., Li, T., Hanks, S., Meyer, D., Traina P., "Generic routing encapsulation (GRE)", IETF RFC 2784, March 2000, http://tools.ietf.org/rfc/rfc2784.txt

[27] JUNOS Logical Systems, http://www.juniper.net/techpubs/software/junos/junos93/feature-guide/id-11139212.html

[28] Hibler, M., Ricci, R., Stoller, L., Duerig, J., Guruprasad, S., Stack, T., Webb, K., Lepreau, J., "Large-scale virtualization in the Emulab network testbed. In USENIX 2008 Annual Technical Conference on Annual Technical Conference (ATC'08)", USENIX Association, Berkeley, CA, USA, 113-128.

[29] Emulab hardware overview, https://users.emulab.net/trac/emulab/wiki/UtahHardware

[30] Virtual Networks Research Group (VNRG) of Internet Research Task Force (IRTF), http://irtf.org/vnrg

[31] Bhatia, S., Stasi, G., Haddow T., Bavier, A., Muir, S., Peterson L., “Vsys: a programmable sudo”, in Proceedings of the USENIX conference on USENIX annual technical, 2011 conference (USENIXATC'11). USENIX Association, Berkeley, CA, USA, 22-22.

[32] Dommety, G., “Key and Sequence Number Extensions to GRE”, IETF RFC 2890, 2000, http://tools.ietf.org/html/rfc2890

[33] Campanella, M., “The FEDERICA Project: creating cloud infrastructures”, http://www.fp7-federica.eu/documents/CloudComp09-FEDERICA-v7.pdf

[34] VMware vSwitch, http://www.vmware.com/pdf/vi3_35 /esx_3/r35u2/vi3_35_25_u2_3_server_config.pdf

[35] Pfaff, B., Pettit, J., Amidon, K., Casado, M., Koponen, T., Shenker, S., “Extending Networking into the Virtualization Layer”, in*HotNets*, 2009.

[36] VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks, Internet Draft – Experimental, http://tools.ietf.org/html/draft-mahalingam-dutt-dcops-vxlan-00

[37] NVGRE: Network Virtualization using Generic Routing Encapsulation, Internet Draft –Experimental, http://tools.ietf.org/pdf/draft-sridharan-virtualization-nvgre-00.pdf

[38] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., Turner, J. , “OpenFlow: enabling innovation in campus networks”, *SIGCOMM Comput. Commun. Rev.* 38, 2, 69-74, 2008.