Empowering GÉANT deployments with ONOS brigades

Authors

Luca Prete (*), Alaitz Mendiola (**), Kostas Stamos (***), Pier Luigi Ventre (*), Andreas Papazois (****), Carolina Fernandez (*****), Jordi Ortiz (*******), Stefano Salsano (******), Afrodite Sevasti (****), David Boswell (*). (* Open Networking Lab, ** University of the Basque Country, *** CTI, **** GRNET S.A., ***** i2CAT, ****** CNIT/University of Rome Tor Vergata, ******* University of Murcia)

Detailed affiliations

Luca Prete 1000 El Camino Real STE 100, Menlo Park, 94025, USA, Phone: +1 415 350 0899
Alaitz Mendiola Alameda Urquijo S/N, 48013, Bilbao, Spain, Phone +34 946018250
Kostas Stamos Building B, University of Patras, Rion 26504, Greece, Phone: +30-2610996954
Pier Luigi Ventre 1000 El Camino Real STE 100, Menlo Park, 94025, USA, Phone: +1 650 322 2720
Andreas Papazois Building B, University of Patras, Rion 26504, Greece, Phone: +30-2610996954
Carolina Fernandez Fundació i2CAT, c/ Gran Capità 2-4 Edifici Nexus I, 08034 Barcelona, Spain, Phone: +34 935 532 510
Jordi Ortiz Facultad de Informática. Campus de Espinardo S/N. 30100, Murcia Spain, Phone: +34 868 888532
Stefano Salsano
Dip. Ing. Elettronica,University of Rome Tor Vergata, Via del Politecnico, 1, 00133, Rome, Italy, Phone: +39 06 72597770
Afrodite Sevasti, Greek Research and Technology Network, 7, Kifisias Av., 11523, Athens, Greece, Phone: +302107474276
David Boswell 1000 El Camino Real STE 100, Menlo Park, 94025, USA, Phone: +1 650.322.2720

Vitae

Luca Prete (luca@onlab.us) Luca Prete is an SDN enthusiast, currently leading IT and deployment activities at Open Networking Laboratory (ON.Lab). He received Bachelor's Degree in Computer Science from University Milano, Italy and the master's degree in Internet technologies from the University of Pisa, Italy. He has been involved in computer science consulting since 2005. Before joining ON.Lab, he collaborated with the Italian NREN in the R&D Department, focusing his attention on Software-Defined Networks.

Alaitz Mendiola (alaitz.mendiola@ehu.eus) received her BSc and MSc degrees in telecommunication engineering in 2012 from the University of the Basque Country (UPV/EHU) and a MSc in Information and Communication Systems in Wireless Networks at the UPV/EHU in 2014. She joined the I2T research lab in 2010 and she has participated in several OpenFlow/SDN related projects. Her research interests include Software-Defined Networking, Network Virtualization and DOCSIS access networks. She has been involved in GN3X, GN4-1 and in the DynPaC projects.

Kostas Stamos (kostas.stamos@gmail.com) received his PhD from the Computer Engineering and Informatics Department, University of Patras. He works as a research engineer for GRNET, and has also worked for Computer Technology Institute and Press “Diophantus”, the Technological Educational Institute of Patras and the Hellenic Open University, where he had the chance to work on multiple EU research projects. His research interests include Software-Defined Networking, Quality of Service and Bandwidth on Demand.

Pier Luigi Ventre (pier.luigi.ventre@uniroma2.it) received his Master’s degree in Computer Engineering from University of Rome Tor Vergata in 2014, with a thesis on Information-Centric Networking and Software-Defined Networking. From 2013 to 2015, he was one of the beneficiary of the scholarship “Orio Carlini” granted by the Italian NREN GARR. His main research interests focus on Computer Networks, Software-Defined Networking, Virtualization and Information-Centric Networking. He worked in several EU research projects and currently he is a PhD student in Electronic Engineering at University of Rome Tor Vergata.

Andreas Papazois (papazois@ceid.upatras.gr) is currently a Network Applications Engineer at GRNET S.A. and Post-Doctoral Researcher at Computer Engineering and Informatics Department, University of Patras. In the past, he has worked in Intracom Telecom S.A. and in Computer Technology Institute and Press “Diophantus”. His research interests include future network services, mobile networks and Software-Defined Networking.

Carolina Fernandez (carolina.fernandez@i2cat.net) received her BSc and MSc degrees on Computer Science from the Autonomous University of Barcelona. She currently works in Software Networks-based applications and infrastructures, their management and federation; as applied in several European R&D projects. Her research interests include Software-Defined Networking, Network Function Virtualization, automation and security.

Jordi Ortiz (jordi.ortiz@um.es) received his B. Sc (2008), M.S. (2009) degrees from the University of Murcia, Spain and is currently a PhD student in the same university. He works as a researcher for the Department of Information and Communications Engineering (DIIC) since 2007. Mr. Ortiz has been involved in multiple research projects in diverse fields such as AAA/Identity environments (DAIDALOS2, STORK2, GN4 SA5T1, GN4-2 JRA3T3), Scalable Video Transmission (SCALNET), Experimentation Infrastructures and deployments (OpenLab, Smartfire) or SDN related solutions (GN3+ JRA2T2, GN4 JRA2T2, GN4-2 JRA1-T2).

Stefano Salsano (stefano.salsano@uniroma2.it) is an Associate Professor in the Electronic Engineering Dept. of University of Rome Tor Vergata. He has participated in 15 research projects funded by the EU, being technical coordinator in two of them (SIMPLICITY, SCISSOR) and project coordinator in one (SUPERFLUIDITY). His current research interests include Network Function Virtualization, Software-Defined Networking, Cybersecurity, Information-Centric Networking, Mobile and Pervasive Computing, Seamless Mobility.

Afrodite Sevasti (sevasti@grnet.gr) is leading Business Development activities at GRNET. At the same time, she leads the GÉANT activities on Software-Defined Networking/Network Function Virtualization evolution and interoperation with e-Infrastructures and Service Providers. Her experience is in advanced network technologies and services as well as the design and delivery of next generation OSS/BSS and IT service management solutions for non-commercial environments. Afrodite holds a PhD from University of Patras, in the field of QoS enhancements on IP-based networks as well as an MSc degree in Information Networking from Carnegie Mellon University.

David Boswell (david@onlab.us) is a community expert who has over 15 years experience with open-source communities and has worked with Mozilla, NASA, Sun Microsystems, Nokia and more on their open-source efforts. David has a Masters of Public Administration degree from Columbia University and was at the University of Texas as an undergrad.

Keywords

SDN, deployment, ONOS, OpenFlow, community, brigade

Introduction

GÉANT has recently joined ONOS core members and other affiliated organizations in a working group called “deployments brigade”. The goal of the brigade is to create a SDN open-source software stack based on ONOS that can be deployed by Research Education Networks (RENs) and Operators. After only a few months of work the brigade released non-trivial software components, already part of field trials within GÉANT. This document explains a) GÉANT requirements and motivations to join the project; b) the concept of ONOS brigades and the goals of the deployment working group; c) the infrastructure and the software stack under development and how GÉANT and its associates intend to test and deploy it; d) the goals achieved and the tasks still in progress.

Status of art: GÉANT network today

GÉANT [GN4] (Figure 1) is the Pan-European REN interconnecting Europe’s NRENs, serving over 50 million users. As demand for more bandwidth, infrastructure and services grows, it becomes more challenging to manage the complex supporting underlays, while keeping costs low.

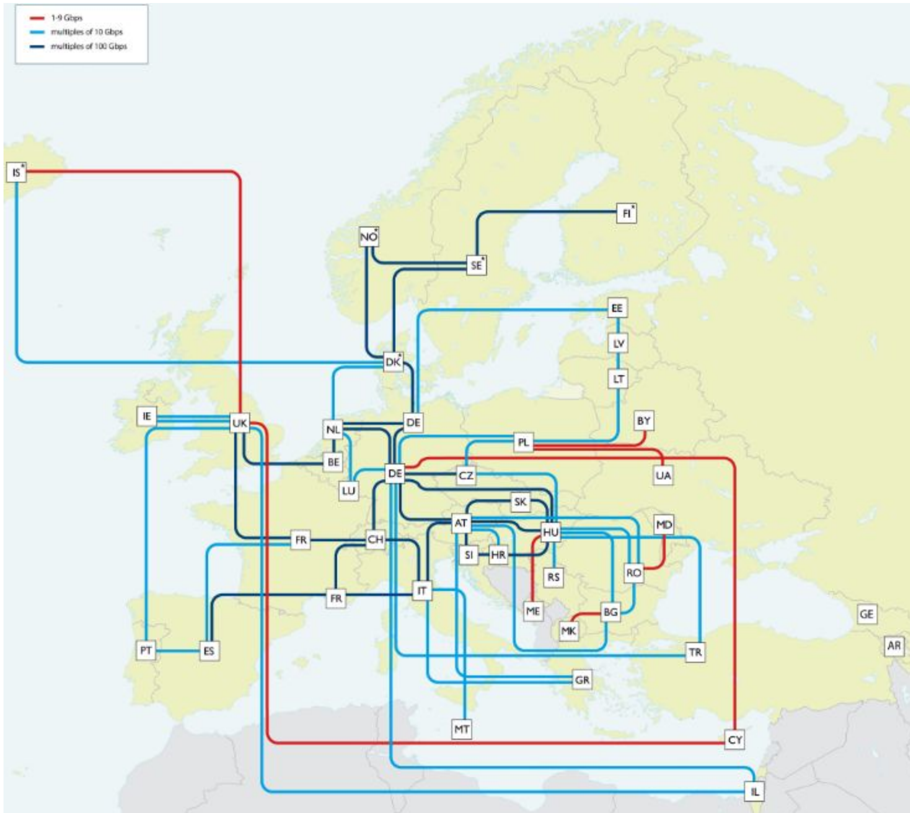

Figure 1 - GÉANT topology, January 2017 (www.geant.org)

Currently, GÉANT is transitioning from doing traditional service provisioning to more dynamic and costless workflows, leveraging multi-point network fabrics and providing services such as programmable traffic processing, monitoring and traffic engineering. Examples include the operation of GÉANT Open [GOpen]: a neutral, policy-free interconnection facility between different Research and Education Networks, users and commercial operators, where GÉANT aims to automate connections and ease IXPs expansion; the GÉANT Ethernet-based connectivity services, where focus is on resiliency and bringing more flexible traffic engineering capabilities; and the interconnection (IP trunks) of GÉANT routers, expected to be managed more efficiently in terms of routing interface usage and handling of elephant flows.

In the case of GÉANT Open, connecting two endpoints (either tagged or untagged) requires the manual intervention of a network operator that needs to set up the service after a user request. Furthermore, a request between users of interconnected exchanges requires the involvement of multiple network operators, their coordination and manual execution of the required actions. There is therefore a strong incentive to automate and speed up the process, while making the expansion easier.

GÉANT connectivity services provide multi-domain, layer 2 circuits for a specified duration to the users. The inter-domain coordination is achieved through a standard interface used by all the parties: the Network Service Interface (NSI) Connection Service [NSI-CS]. Within the single

domains, operating such service is still expensive, complex and time consuming: proprietary devices and platforms are used to provision the requested circuits; operators need to keep track of network resources (for example, bandwidth, endpoints and VLANs) through a static database; the synchronization between the services and the infrastructure is occasionally dealigned, requiring a manual intervention to reinstate the proper state; when the services need to be migrated to new or different equipment, a custom software component, aware of the new hardware details, needs to be developed, making the transition long and painful; support for resilience and failure recovery options is limited.
In summary, the infrastructure can offer innovative services today, but there are still limits in terms of scalability and functionality, and it is also difficult to maintain the services.
Software-Defined Networking (SDN) seems a key solution to address these challenges. SDN promises to reduce CAPEX and OPEX by lowering costs of hardware and speeding up management, delivering network services in a more standard, configurable, and programmable way.

ONOS

The Open Networking Operating System (ONOS) [ONOS] is an open-source SDN control plane platform, meeting Service Provider requirements, released in 2014 as an open-source project by ON.Lab [ON.Lab]. ONOS aims at transforming Service Provider networks through the adoption of open-source SDN software. The architecture is based on three key tenets: i) high availability, scalability and performance; ii) powerful abstractions; iii) separation of concerns and modularity. The overall system has been conceived as a distributed system in the form of a cluster, composed of multiple instances, all functionally identical.
The architecture (Figure 2) can be structured as three tiers: a protocol-aware southbound layer, a protocol-agnostic distributed core layer, and an application layer. Each tier is a collection of pluggable modules/subsystems realizing specific functionalities that make up the ONOS platform. An API is exposed at each tier, providing isolation and modularity.

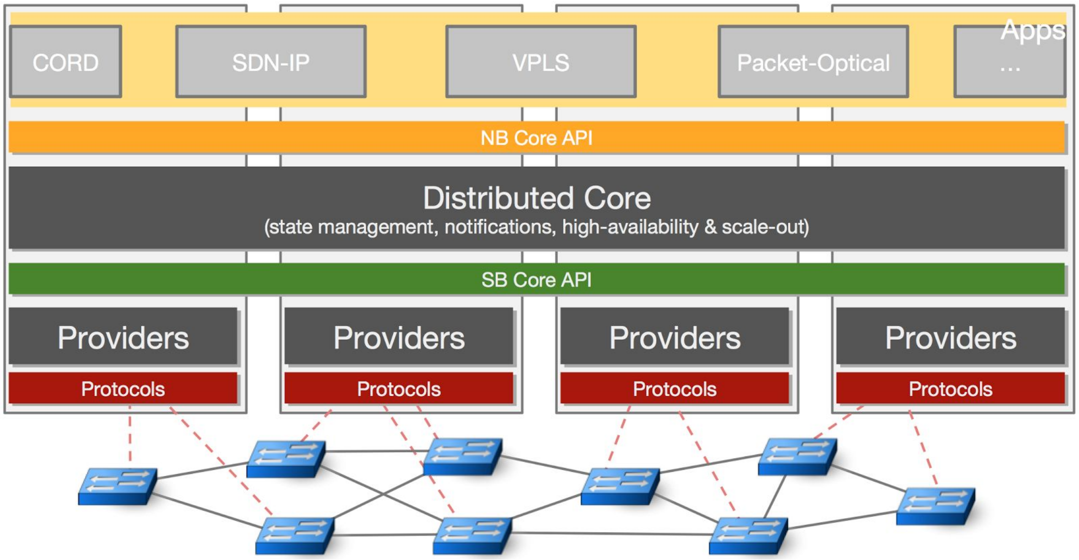

Figure 2 - ONOS high level architecture

The distributed core is responsible for synchronizing and coordinating between the instances in the cluster, building a global network view based on the information learned from the providers and offering services to the application layer. In order to achieve scalability and provide resiliency, various distribution mechanisms are available through a set of primitives. Each core subsystem uses these primitives in different ways, according to the consistency requirements of the state it is managing. On top of the distributed state, a logically centralized network view is constructed and presented to applications. In addition, the workload is partitioned amongst the instances in the cluster. For example, each instance is elected to be responsible for managing a subset of the devices in the network, while the other instances are ready to step in if the primary instance fails.

The southbound layer consists of a collection of software modules called ‘providers’, which interact with data plane devices using different southbound protocols. Providers are designed to be stateless. Their responsibility is to collect information about network state and pass it to the distributed core, as well as to receive instructions from the core and program the devices appropriately. ONOS has a well-defined southbound provider API, which ensures that the system is modular and can support different southbound protocols through the use of different providers.

On the northbound side, ONOS presents powerful abstractions to the applications, including Network Topology, Flow Objectives and Intents. Intents provide applications with a network-centric programming abstraction that allows developers to program the network through the usage of high-level policies that capture what needs to be done, rather than how to do it. The Intent framework determines how to implement an intent and abstracts low-level details of this implementation. Intents make network policies configuration easier, speed up management procedures and tend to reduce the occurrence of configuration errors. Intents are backed by a dedicated subsystem that: i) translates intents into device instructions; ii) coordinates and ensures the installation of the generated instructions; iii) reacts to network changes and modifies paths accordingly; iv) permits optimization across intents translations. Flow Objectives

provide flow-like abstractions and are used to manage the diversity introduced by the multiple table framework and groups defined by the OpenFlow protocol. In other words, they represent the internal ONOS pipeline through which Applications can describe their directives without knowing the implementation details of the device. Currently, three Flow Objectives have been implemented in ONOS: i) Filtering Objective to identify what incoming traffic should be allowed or blocked; ii) Forwarding Objective to define how to forward the traffic in the switch's pipeline; iii) Next Objective to specify the actions performed on the packets at the egress of the pipeline (for example, MAC address rewriting).
The ONOS project is supported by a large open-source community which has improved the platform and has introduced new compelling applications.

ONOS brigades

The ONOS project started in December 2014 and has been steadily gaining contributors since then. As of the time of this writing there are over 60 organizations as well as many individual contributors who are in the community.
As the ONOS community continues to grow, it becomes more challenging to coordinate a large group and make sure everyone is working toward a shared goal. Without a way to coordinate community members we have found that contributions have often not been aligned around top priorities, as identified on the ONOS roadmap.
One way to address this is to clearly communicate the ONOS vision and rally people around creating teams to work together on completing specific parts of that vision. This is where the brigade model comes in. Brigades are teams focused on developing specific features to ship in upcoming versions of ONOS. Once formed, anyone else in the community with interest in such features can join the group and contribute to it.
The concept for brigades has been borrowed from Code for America [Code-for-America], a community of people helping government to deliver better services to the public by using the tools and practices of the digital age. They have successfully organized their efforts using brigades and have been able to scale their activities to over 100 brigades using this model.

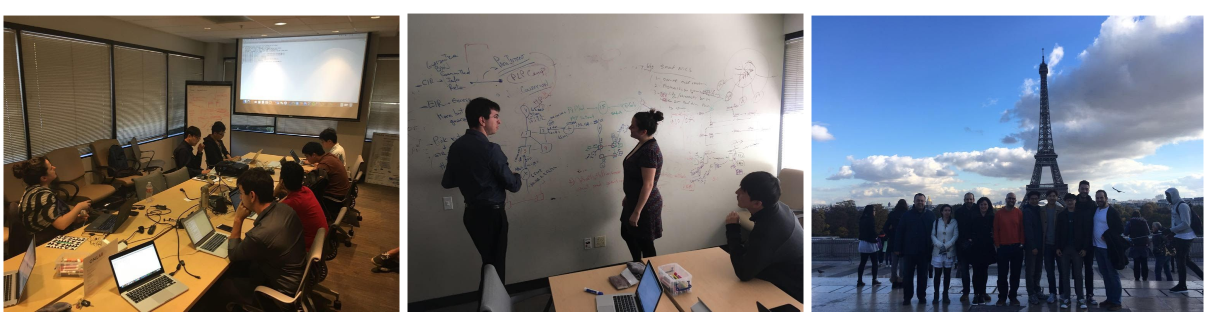

Figure 3 - Pictures from the first meetings of the deployment brigade. On the left at ON.Lab premises in Menlo Park; on the right in Paris, for the ONOS Build conference.

The Deployment Brigade (Figure 3) was the first attempt to use the model in the ONOS community and it turned out to be an effective way to recruit community members and ship new features. GÉANT and other RENs joined the brigade as founding members.

Today the brigade is at its sixth month of activity and is composed of 18 active members from academia, research organizations and industry, and has already achieved some successes. The next section gives more details about the brigade's activities working on these features.

The brigade has succeeded in making progress on its goals and because of this success we have expanded the number of ONOS brigades in 2017 -- there are now 10 active brigades and more being planned. From our experience we would encourage other open-source communities to consider this model as a way to focus community participation and rally people to contribute to an area where more participation is wanted.

The deployment brigade

ONOS has been extensively deployed in research and commercial networks around the world, for over three years. Deployment activities in the past have involved engineers from different organizations that either ended up deploying the software stack proposed by the existing community or developed their own.

While this approach helped many organizations to develop useful skills and useful in-house software, it created a large ecosystem of applications that were performing essentially the same functions but implemented in different ways. Another critical disadvantage of having organizations working in silos was that there wasn't enough coordination between those developing and testing the software with those operating the networks where the software will be deployed.

The deployment brigade was formed to address these concerns and once these research networks and operators came together in the deployment brigade it became clear that they shared the same goals and could benefit from working together in a more coordinated way.

The goal of the brigade is to build a software stack running on top of ONOS, able to facilitate the adoption of the platform in field trials and production deployments by RENs and operators. Basic requirements include providing layer 2 and layer 3 functionalities to networks' users, convergence of packet and optical resources, and compatibility with major standards.

Since the first meetings, the brigade members agreed to work toward converging on a shared stack (Figure 4) of software while still allowing for minor deployment-specific customizations. Using the same applications allows the whole group to benefit from better test coverage, get more significant feedback from a larger community and receive better support from the core developers.

While converging on shared software was something desirable but not easily achievable in the short term, some engineers from each organization have kept working on their existing code, while others focused on the development and the enhancement of the shared one. The goal for those developers was to show the first proof of concepts, while bringing back valuable feedback to the community to improve the shared application stack.

The stack that the group decided to share and improve includes applications able to help operators managing different network layers. Specifically, VPLS [VPLS] is used to create on-demand layer 2 broadcast networks; Carrier Ethernet [CE] supports MEF standards for the creation of layer 2 services like E-Lines (point to point circuits) and E-LANs (multi-point broadcast networks); SDN-IP [SDN-IP] transforms SDN networks into IP transit networks, translating BGP messages exchanged with external ASs to OpenFlow entries on the switches inside the domain; packet-optical [PO] is able to convergently manage the packet and the optical layers, both during normal operations and after failures.

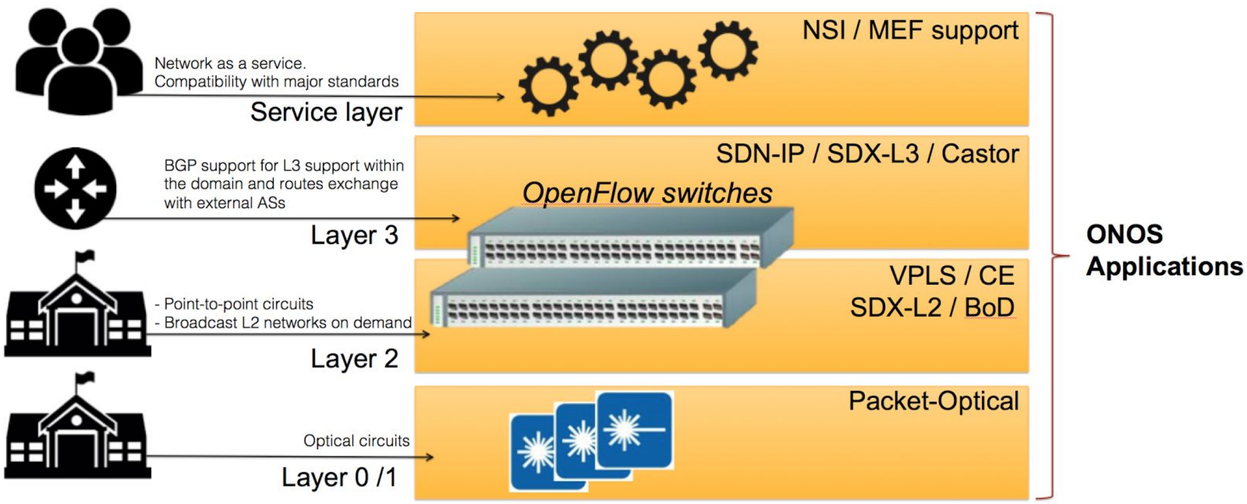

Figure 4 - The software stack developed by the deployment brigade

The applications developed within the RENs can control most of the network layers as well, and answer similar needs. For example, GÉANT and AARNET [AARNET] have created SDX-L2/L3 [SDX-GN4] and CaSToR [CaSToR] to provide layer 2 and layer 3 services to their users; GÉANT also developed Bandwidth on Demand (BoD) [BoD-GN4], an application able to interpret NSI requests and program the data plane accordingly.

The activities of the deployment brigade include the design, the development of new software components, and their integration; Quality Assurance and deployments activities. Members usually take part in different sub-projects and activities, depending on their background and interest.

Design, development and integration activities

Design and development activities are oriented to improve existing applications and create new features.

Integration tasks are fundamental, not only because code from different organizations has to be merged together, but also because even applications created by the same entities had never worked together in a unique stack before.

Creating new features requires the members of the brigade to follow a certain pattern. The group first finds an agreement on the goals and the overall design of the software; then, after a more accurate design phase conducted by a smaller team, the component is developed and finally integrated with the rest of the solution. If the component meets certain quality criteria, it usually gets merged into the ONOS code base.

During the design stage, requirements are provided and analyzed by different brigade members, evaluating the appropriateness and analyzing the consequences of such additions to the code base. Later, the development activities are oriented to improve existing applications and create new features.

Integration tasks are fundamental to merge sources from different organizations while keeping a consistent and readable style; and to make sure that applications, even those created by the same entities, work harmoniously in the same stack.

The brigade initially decided to focus on layer 2 applications, as they are easier to develop, operate, and have fewer performance requirements.

Both SDX-L2 and VPLS have been evaluated as possible candidates to be part of the final software stack. While SDX-L2 [NETSOFT16SDXL2] was the first application deployed in a field trial, VPLS was selected as the potential application to be put in production at a later stage. Both of them realize similar functionalities. The main difference is the way users define the connectivity services: the former provides EVC (Ethernet Virtual Circuit), allowing the creation of Ethernet pipes, both through VLAN tags and MPLS labels, and offers a GUI for easier use. VPLS allows the creation of broadcast layer 2 overlay networks between multiple endpoints, supports dynamic reconfiguration of the overlays and provides resiliency after failures happen. The brigade is looking at the most valuable features offered from both applications, to integrate them in a unique solution to be deployed in the field. For example, SDX-L2 development and testing has produced valuable feedback that has benefited both VPLS and ONOS through new code contributions. The ONOS Intent framework has been properly extended to provide encapsulation support for all intent types, including the ones used by VPLS. The SDX-L2 team delivered several bugfixes which added unique new features to VPLS and contributed to improve the stability and the performance of ONOS. In just a few months, VPLS has been almost entirely re-written: it now supports tagged, as well as untagged interfaces, encapsulation, and it has a new CLI. At the same time, VPLS produced valuable inputs to the SDX-L2 community, which decided to expand its functionalities providing the missing features as part of the final deliverable. As an example of the latter additions, SDX-L2 borrowed from VPLS some CLI commands to further manage the EVCs.

SDN-BoD [BoD-GN4] is another example of a service that GÉANT has recently started evaluating. GÉANT already provides a BoD point-to-point provisioning service, in collaboration with a set of adjacent RENs [GBoD]. The service allows users to provision in a multi-domain environment VLAN-based layer 2 network slices for a specific period of time, using an intuitive UI.

SDN-BoD can be considered an evolution of the existing service. It aims to leverage SDN, and specifically ONOS, to provision the layer 2 circuits in such environment, as a potential alternative to the legacy technologies used today. SDN-BoD is able to receive standard NSI requests and program the SDN data plane accordingly. Features of SDN-BoD include the ability to identify clients based on a VLAN at the edge of the network, doing VLAN translation, bandwidth provisioning, calendaring and enforcement. To allow these functionalities, the DynPaC framework [NETSOFT16BoD] has been integrated as part of a new ONOS application, to provide NSI parsing functionalities. The application communicates with the GÉANT's AutoBAHN [AUTOBAHN] (Automated Bandwidth Allocation across Heterogeneous Networks) framework which is able to receive user's requests and abstract the differences between the specific domain implementations (for example, legacy vs SDN devices). ONOS itself is used to program the underlying infrastructure, according to the requests received by the application.

In order to enforce the bandwidth for a specific circuit, the SDN-BoD makes use of OpenFlow 1.3 meters. As part of the work done, the BoD team has integrated into ONOS all the primitives needed to be able to use OpenFlow meters. The brigade is now introducing meters support in the intent framework, allowing users of the platform to ask the system through high level policies to provision circuits between multiple end-nodes enforcing a certain bandwidth.

While the AutoBAHN GUI is used to manage resources across multiple domains, the BoD app also takes advantage of the ONOS GUI, as an alternative to the more classic CLI, to let operators manage their internal network resources.

The brigade is now evaluating how the intent framework can be improved in order to migrate the applications on top of it and let it leverage it in the future.

More members of the brigade started to bring together in the same stack the layer 2 applications mentioned above with layer 3 applications, such as SDN-IP and SDX-L3. Although these have been all successful proof-of-concepts, they have never been run concurrently in the same deployment. At this stage, the brigade has already demonstrated that such applications can work together, meaning that a given SDN device can now act at the same time as an advanced layer 2 switch, as well as a router.

A next step will consist in integrating the features provided by SDN-IP and SDX-L3 into a unique solution, just as it has been done for SDX-L2 and VPLS.

The last part of the development activities consisted in writing specific device drivers, that have been now included as part of the official ONOS code base. This will allow the deployment of the aforementioned applications on top of the devices that GÉANT aims to use in production.

Testing and deployment activities

Quality Assurance and deployment activities have been fundamental since the beginning to keep a high quality of code and a frequent engagement with the operation departments.

SDX-L2 and SDN-BoD have been moved to a "pilot phase", consisting of both testing and deployment activities.

The applications have both been extensively tested. Tests include functionality tests to validate the correctness of the data plane, of the control plane, and their integration with the underlying hardware.

For example, data plane tests verify that the layer 2 circuits are correctly set up using any combination of tagged and untagged endpoints; control plane tests aim to check the performance of the controller and its applications, and also to evaluate its resiliency and failure recovery characteristics after failures happen. Hardware integration tests check that the operating systems works as expected together with the underlying devices, both during normal operations and in case of failures.

Part of the purpose of the pilot is also to evaluate the effectiveness and usability of the application in a production context. As a result of this evaluation, new requirements and constraints have been identified, leading to new implementations to fully support OpenFlow 1.3.
After a first testing period, the software has been deployed in the GÉANT backbone network for further tests and field trials. The deployment takes advantage of programmable switches providing hardware virtualization functionalities. This allows GÉANT to concurrently run isolated layer 2 network slices, and the team to run disruptive experiments on the production network, without interfering with the rest of the traffic.
Different slices have been activated in the GÉANT PoPs in Milan and Paris. Each slice runs a different application, thus realizing different use-cases that provide services to one or more users. An ONOS cluster made of three instances has been deployed in a third PoP in Madrid. The cluster controls different slices and collects useful debug information.
At this time, both SDX-L2 and SDN-BoD have been extensively tested by the development team and are now ready to be tested in a production slice by the GÉANT NOC.

Conclusions and next steps

The applications described above are part of the use cases that are to be deployed on the GÉANT network by Q2-2017, first as pre-production solutions, later as part of the official services offered by GÉANT to its users. Before reaching the pilot stage, applications are subject to a detailed list of tests that address aspects such as application and network performances, the integration of the ONOS platform with the underlying hardware, as well as the usability of the overall solution.
The deployment brigade has produced a twofold contribution: on one hand, it worked with GÉANT to evaluate and take advantage of new SDN solutions, thereby allowing a wider community to be part of this work and improve its chances for success. On the other, it contributed back to the ONOS community with significant artifacts (including new applications, features, improvements and bug-fixes), made available in the ONOS repository and now usable by other network operators.
The brigade initiative has been so successful that ONOS project has expanded the number of active brigades in 2017.
The brigades initiative also created the opportunity for internships of GÉANT team members at ON.Lab, enhancing the communication between the team members and shortening the learning curve of the ONOS architecture. This way of working is penetrating throughout the GÉANT teams and they are producing outcomes directly to the open-source code base and making the project visible to a part of the industry that is starting to see SDN as something more and more present and real. Being involved in the daily life of the open-source project as part of the brigades is producing an “unexpected” benefit to the team in the form of advice and patches.
In the spirit of open source, all the applications developed have been made completely available to the broader community. The work of the brigade and the results of the tests have been showcased and demonstrated at different international meetings and conferences.
The brigade is now extending the ONOS intent framework to introduce the notions of queues, meters and bandwidth, and is evaluating how the actual applications can run over it once all

these functionalities will be in place. As opposed to working as stand-alone applications, both BoD and CE will be integrated with the stack, bringing compatibility with the NSI and MEF standards while reusing the primitives offered by SDX-L2 and VPLS. SDX-L3 and SDN-IP will be first integrated and then merged with the rest of the stack, tested and deployed. Finally, more and more tests will be conducted before moving the platform to production, in order to ensure the robustness of the solution before it will be offered -as an official service- to the broader community.

References

[AARNET] https://www.aarnet.edu.au

[AUTOBAHN] https://geant3.archive.geant.org/service/autobahn/pages/home.aspx

[BoD-GN4] https://wiki.onosproject.org/display/ONOS/BoD+DynPac+application

[CaSToR] http://www2.ee.unsw.edu.au/~vijay/pubs/conf/16ewsdncastor.pdf

[Code-for-America] http://brigade.codeforamerica.org/brigade/about/

[CE] https://wiki.onosproject.org/display/ONOS/Carrier+Ethernet+Application

[GBoD] https://www.geant.net/Services/ConnectivityServices/Pages/Bandwidth_on_Demand.aspx

[GN4] https://www.geant.org

[GOpen] https://www.geant.org/Services/Connectivity_and_network/Pages/GEANT_Open.aspx

[NETSOFT16BoD] Multi-domain bandwidth on demand service provisioning using SDN http://ieeexplore.ieee.org/document/7502407/

[NSI-CS] G. Roberts et al. "NSI Connection Service v2.0", GFD-R-P.212, Open Grid Forum, 2014

[ON.Lab] http://www.onlab.us

[ONOS] http://www.onosproject.org

[PO] https://wiki.onosproject.org/display/ONOS/Packet+Optical+Convergence

[SDN-IP] https://wiki.onosproject.org/display/ONOS/SDN-IP

[SDX-GN4] https://wiki.onosproject.org/pages/viewpage.action?pageId=4753071

[NETSOFT16SDXL2] GÉANT SDX - SDN based Open eXchange Points http://ieeexplore.ieee.org/abstract/document/7502403/

[VPLS] https://wiki.onosproject.org/display/ONOS/Virtual+Private+LAN+Service+-+VPLS