An SDN/NFV Telco Operator Platform for Video Broadcasting

Valeria D'Amico, Alfio Lombardo, Marcello Melita, Corrado Rametta, and Giovanni Schembra

The authors present an SDN/NFV-based platform for telco operators who intend to enter the market of video broadcasting service providers. The proposed platform allows telco operators to achieve deployment simplification and management cost reduction to support big/small live streaming content providers, and even unusual video broadcasters, in transmitting video flows to a number of interested users that can be either fixed or mobile, without the need to adopt a dedicated and expensive data delivery infrastructure.

Abstract

In the last few years, SDN and NFV have introduced a new way to design, deploy, and manage networking and application services in telecommunications networks. In this context, this article presents an SDN/NFV-based platform for telco operators who intend to enter the market of video broadcasting service providers. The proposed platform allows telco operators to achieve deployment simplification and management cost reduction to support big/small live streaming content providers, and even unusual video broadcasters, in transmitting video flows to a number of interested users that can be either fixed or mobile, without the need to adopt a dedicated and expensive data delivery infrastructure. The design of the architecture proposed in this article is the result of a cooperative work realized by Telecom Italia, the University of Catania, and Wave Joint Open Lab (JOL), representing three of the most important roles in the evolution process of telecommunications networks toward network softwarization, that is, telco operator, academia, and research lab. Besides an operational and functional description of the proposed architecture, the article discusses implementation aspects of the platform, by introducing a proof of concept that has been realized on the JOLNet platform, a nationwide Italian SDN experimental network developed by Telecom Italia to interconnect its JOL labs with a large-scale geographical facility, with the aim of carrying on research and experimentations on SDN technologies and applications.

Introduction

In the last decade, video delivery on the Internet has registered a tremendous increase. As recently reported in [1], online video streaming viewers are growing at a staggering rate, whereas the number of TV watchers are declining, and according to a Cisco study [2], 75 percent of mobile traffic will be video by 2020. For this reason, the problem of optimizing video transmission on the Internet has been the object of research in the past and continues to be challenging today. Specifically, the authors in [3] presented an analysis which demonstrates that users have much more interest in broadcast video than on-demand content, with applications ranging from video transmitted by big broadcasting stations to video of occasional sources for short-lived events, such as sport transmissions, university lectures, and even personal parties and journeys.

Unfortunately, common solutions for video on demand (VoD), like those used by YouTube, Hulu, and Netflix, which cache at the edge of the network to keep content near users to reduce backhaul traffic and content access delay, cannot be used for live video broadcasting where video has to be received when it is produced.

Another approach that has been attempted in the past is IP multicast. However, as known, it exposes telco operators to a number of issues that discourage them from its application: first, it is no longer affordable to use static multicast to stream all TV channels, as some telco providers do at present; second, using dynamic IP multicast can lead to scalability problems, which can translate to increased cost for the service providers, especially when the number of users viewing a multicast channel is low.

This trend has stimulated the interest of a huge number of over-the-top (OTT) providers, which have sustained the diffusion of these applications through client-server or peer-to-peer (P2P) approaches [4]. The P2P approach, followed, for example, by the PPLive and PPStream platforms, is an OTT approach that, besides problems of high latencies due to the long path a video flow has to follow, suffers possible instabilities due to the volatile behavior of the clients that play the role of peers. Moreover, the OTT approach reduces telco operators to the limited position of only providing telecommunications infrastructure or, at most, contributing to on-demand video applications with a content delivery network infrastructure, and hence not entering the more appealing arena of live multimedia service provisioning.

In the last few years, the introduction of software defined networking (SDN) [5] and network functions virtualization (NFV) [6] is revolutionizing telecommunications networks, giving new perspectives to telco operators. The power of SDN is based on its characteristic of decoupling control and data planes, moving the network intelligence to a centralized controller. On the other hand, the emerging NFV paradigm introduces an important change in the network service provisioning approach, leveraging on standard IT virtualization technologies to consolidate many

Valeria D'Amico and Marcello Melita are with Telecom Italia; Alfio Lombardo, Corrado Rametta, and Giovanni Schembra are with the University of Catania.

0163-6804/16/$25.00 © 2016 IEEE

network equipment facilities onto standard servers that could be located either in data centers, in core network nodes, or at the edge of the network [7, 8].

Now, thanks to the joint application of SDN and NFV, telco operators have the opportunity to strongly contribute to the flexible, agile, and efficient deployment of multimedia services. In this context, the challenge for telco operators is to plan a gradual but rapid evolution of their networks according to the above SDN/NFV paradigms in such a way to as to follow the market, and specifically to support broadcasting companies and occasional video broadcasting sources to provide users with their services.

With this in mind, this article describes the current result of a cooperative work realized within the European Project In-Network Programmability for Next-Generation Personal Cloud Service Support (INPUT) [9] by Telecom Italia, the University of Catania, and TI Wave Joint Open Lab (JOL), as representatives of the main three roles in this network softwarization process, that is, a telco operator, academia, and a research lab.

The target of this article is to introduce an SDN/NFV-based platform for telco operators that intend to enter the market of live video broadcasting services. The proposed platform allows telco operators to achieve deployment simplification and management cost reduction, in terms of both capital expenditure (CAPEX) and operating expenditure (OPEX), to enable big/small broadcasting providers and even unusual content sources to transmit live video flows to a number of interested users that can be either fixed or mobile, without the need to adopt a dedicated and expensive data delivery infrastructure. The platform introduced in this article presents the following characteristics:

1. It is backward compatible, since it allows not only the co-presence of legacy and SDN switches, but also interworking between users entering the network through SDN/NFV nodes and users using legacy access nodes.
2. It is able to minimize end-to-end latency thanks to the possibility of creating and updating the video streaming trees according to the position of the content providers and their clients.
3. It maintains its efficiency even in the presence of users who change either their access point during the service due to their mobility or the connection interface.

In addition, the proposed approach, thanks to its "in-network" peculiarity that is intrinsic to the SDN/NFV paradigm, presents many advantages compared to the widely used OTT approach. In fact, by applying the SDN/NFV paradigm, the video content distribution platform runs only in the core network, which is typically over-dimensioned. Moreover, core network nodes that realize flow replication and transcoding are decided and can be varied by the orchestrator at runtime according to the current state of the network, thus making the service platform more robust to possible congestion of the underlying network.

Instead, if it were used, a centralized OTT approach (e.g., Cisco WebEx) that is realized

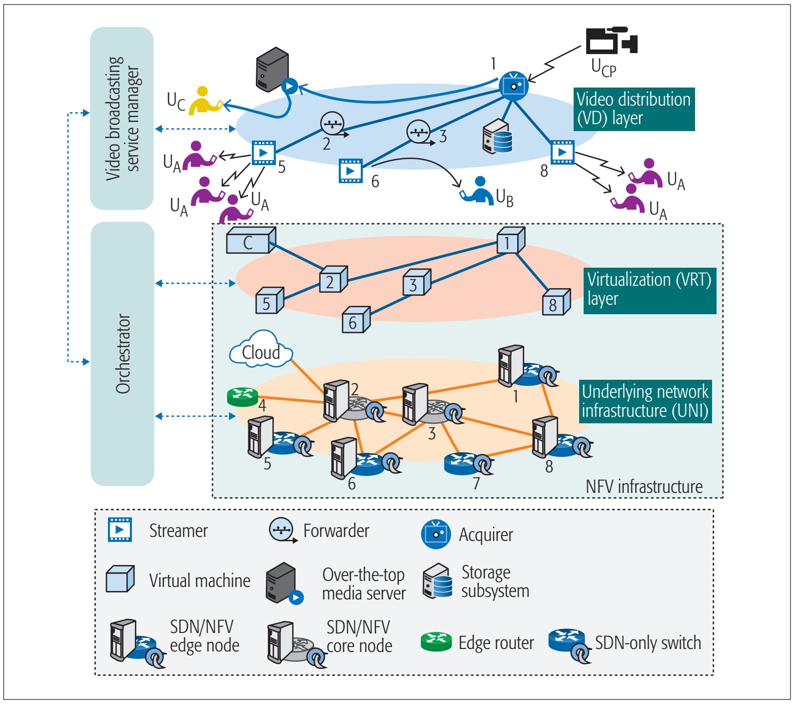

Figure 1. Video broadcasting platform architecture.

by some remote servers running outside the network (e.g., in a cloud) to replicate video flows with one replica for each receiver, the quality of experience (QoE) would be compromised by bad performance in the access nodes of the content providers or the OTT platform servers, which would represent a bottleneck for the whole system.

On the other hand, if a distributed OTT approach were used, such as the P2P live streaming used by PPLive and PPStream, network stability should not be guaranteed due to the volatile behavior of the clients that play the role of peers, and the behavior at the overlay network level should be agnostic regarding the state of the underlying network.

After an operational and functional description of the architecture, the article discusses implementation aspects of the platform, by introducing a proof of concept (POC) that has been realized on the JOLNet platform, an SDN nationwide Italian experimental network, developed by Telecom Italia to interconnect a number of research labs with a large-scale geographical facility to carry on research and experimentation on SDN technologies and applications.

Architecture Overview

The platform proposed in this article, which is compliant with the SDN/NFV paradigm, has the target of providing users with a video broadcasting service. As shown in Fig. 1, it is constituted by three different layers:

1. The underlying network infrastructure (UNI)
2. The virtualization (VRT) layer
3. The video distribution (VD) layer

The two lowest layers constitute the NFV infrastructure (NFVI), according to the NFV terminology [10], while the highest layer is specific to realize the considered service.

The orchestrator manages and orchestrates the two layers of the NFVI, while the VBSM controls the activities at the highest layer. They are separated entities that run on dedicated servers, and communicate with each other and with all the platform nodes through the telco operator IP network.

The *UNI layer* provides the computation infrastructure, and realizes the hardware connectivity among all the elements that constitute the platform. According to the SDN/NFV approach, it is managed by a telco operator, and made up of a set of core and edge nodes connected to each other. It has backward compatibility in such a way that not all the nodes have to be SDN/NFV compliant. For example, referring to Fig. 1, the edge nodes labeled 1, 5, 6, and 8, and core nodes labeled 2 and 3 are SDN/NFV compliant, node 7 is an SDN-only switch (with no NFV capabilities), while node 4 is a legacy access node. All SDN-only nodes are realized by switches that support the SDN protocol; therefore, they are controlled by a remote SDN controller, which resides in the orchestrator. SDN/NFV nodes, on the other hand, are compliant with both the SDN and NFV paradigms [11]. Therefore, they consist of a server, where a hypervisor is installed to virtualize both network and application functions, and an SDN switch, which can be either a software tool installed on the same server (e.g., an Open vSwitch) or separate hardware. Another element that belongs to the UNI, but resides outside the telco operator network, is the cloud, where some software packages run to manage users not accessing the network through an SDN edge node, as described below.

The *VRT layer* is the medium layer of the platform. It is present on the SDN/NFV nodes only. According to the NFV paradigm, it comprises the virtual machines (VMs) that the VNF managers residing in the orchestrator launch on each server of the SDN/NFV nodes. To improve the readability of Fig. 1, the same number identifies a node at the UNI layer and the set of VMs running on it at the VRT layer. The above VMs are able to execute virtualized network and application functions according to the decisions assumed by the orchestrator. More specifically, following the requests coming from the upper layer, the orchestrator launches, stops, halts, migrates, and interconnects VMs in order to create the virtual infrastructure to host the video broadcasting platform.

The highest layer of the proposed platform is the *VD layer*. It realizes the video streaming trees (VSTs), the structures that are responsible for receiving media streams from multimedia content providers (CPs) and forwarding them to a set of interested end users, in the following referred to as clients.

Each CP, indicated as U_{CP} in Fig. 1, is either a stable or an unusual user that produces live videos. Let us notice that each CP can send different video streams simultaneously, each representing a different transmission channel.

Clients are the end users of the system. They connect to the platform through a fixed or mobile terminal, and require the service of receiving videos on different channels from different CPs. As shown in Fig. 1, we have three kinds of clients, according to their access to the network: clients labeled as U_A access the network through an SDN/NFV edge node, where a streamer instance can run, as illustrated below; the access node of U_B clients is an SDN-only node that is able to automatically redirect the client flows to the closer Streamer instance; finally, clients labeled as U_C access the network through legacy edge nodes, and therefore can receive the platform service through a P2P connection with the OTT media server (OTMS), which runs on the cloud.

The actions that can be performed by the clients are: *select provider*, *select channel*, *watch and/or record live channels*, and *view recorded contents*. Of course, one VST is associated with each channel. Once the VST has been created, the CP is attached to the tree root (the acquirer), while the relative clients are attached to the leaves (the streamers).

The main requirement in the definition of this platform is that the VD layer should be seen by the clients of each channel as a tree of legacy hardware nodes that have been specifically deployed to provide a video broadcast service [12]. Accordingly, the VST is made up of virtual instances of the acquirer, the forwarders, and the streamers, which are the root, the internal nodes, and the leaves of the tree, respectively. All the above nodes, the architecture of which is described later in more detail, are virtualized within the VMs that run at the VRT layer.

Two more important elements belong to the VD layer. The first element is the storage subsystem (SS), which is in charge of the storage of multimedia contents according to the requests coming from the clients of the whole platform. As described below, it saves only one copy of each content if at least one recording request has arrived at the system. In order to pursue the best trade-off between the occupied storage space and the latency incurred by the clients to access the recorded contents, it can use some techniques, such as forward error correction (FEC) or random linear network coding (RLNC), to split the recorded content and distribute the relative parts on different nodes in such a way that the original content can be rebuilt from the closest nodes to the current position of the clients requesting that content.

The last important element of the VD layer is the video broadcasting service manager (VBSM), which, cooperating with the orchestrator, has the task of coordinating the platform at the VD layer, as illustrated in the following section.

Platform Coordination

As introduced so far, the two entities aimed at coordinating the whole platform are the orchestrator and the VBSM. The orchestrator manages and orchestrates the two layers of the NFVI, while the VBSM controls the activities at the highest layer. They are separate entities that run on dedicated servers, and communicate with each other and with all the platform nodes through the telco operator IP network.

The *VBSM* works as a front-end entity for the interaction with all the platform actors, that is, the CPs and the clients. For this purpose, the VBSM manages a database containing the lists of CPs and clients, together with their accounting and billing information and quality of service (QoS) requirements. Moreover, from the CP side, the VBSM periodically receives an updated list of channels from each CP and, for each channel, the timetable of the upcoming video contents, in order to provide this information to interested clients. Instead, on the client side, for each upcoming video content, the VBSM collects

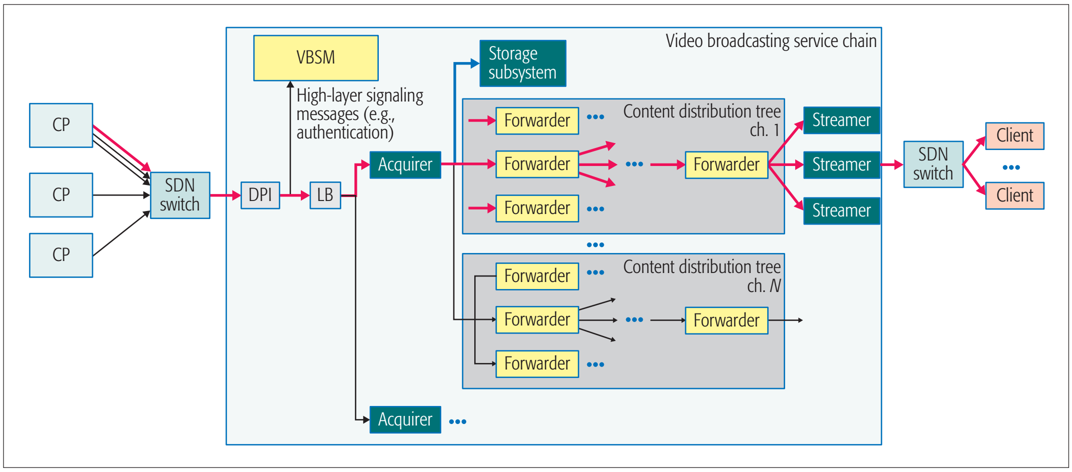

Figure 2. Video broadcasting platform architecture.

> The main hallmark of JOLNet could be summarized as "being a physical geographic network that allows the development of services thereby evaluating potentiality and impacts of a concept as close as possible to a real production network environment"

requests from clients to watch that content, and consequently it issues requests to the orchestrator to organize the underlying NFVI in order to instantiate the relative VST at the scheduled start time. In addition, if it has received some recording request for a given content, it asks the orchestrator to connect the SS to the acquirer of the relative VST.

In the case in which a client accesses the platform through an SDN-only switch (client U_B in Fig. 1), the VBSM provides its IP address to the OTMS in order to be able to send the content directly to that client.

Another service the platform provides to its clients is the playout service of previously recorded contents. Content recording may have been determined by either a request from some client or the same CP when this believes that a content it is transmitting will be appealing to clients in the future. When a client decides to receive this service, it issues a request to the VBSM specifying a content chosen from a list received by it. Consequently, the VBSM contacts the SS to send all the parts needed to rebuild the requested content to the streamer where the client is connected.

The other coordination entity of the platform is the *orchestrator*, which works as a back-end entity that receives high-level requests from the VBSM, and translates them into operative actions on the NFVI layers, implementing specific policies for orchestration and resource management. Its goal is to allocate, migrate, and terminate VMs running both virtual network functions (VNFs) and virtual application services (VASs), also chaining them, managing hardware and virtual resources of the NFVI, and controlling the traffic paths according to the runtime evolution of the underlying network and the platform users. In order to be NFV-compliant, its architecture follows the management and orchestration (MANO) guidelines specified in [13].

NFV Architecture

This section presents a detailed description of the NFV architecture that has been defined to realize the video broadcasting platform presented so far. Figure 2 shows the overall diagram describing the data flows during the video broadcasting service. In this figure we focus our attention on a set of CPs connected to the platform through the same edge node. Video flows emitted by the CPs are intercepted by the SDN switch in the edge node, which identifies that they are carrying video broadcasting channels, and sends them to the *video broadcasting service chain*.

According to the information received by the VBSM regarding the access nodes of the CPs and the clients, the data rate of the video channel flows, and the QoS parameters specified by each client, immediately before the beginning of the transmission the orchestrator instantiates the needed VNFs and VASs, launching the supporting VMs in some nodes of the NFVI and allocating an opportune amount of hardware resources for them. Then the network service orchestrator (NSO) entity residing within the orchestrator [13] creates a VNF forwarding graph connecting the above instances in such a way that the video flows can reach the interested clients as described below.

At the entrance of the video broadcasting service chain, all the flows coming from the CPs first arrive at a deep packet inspector (DPI), which separates the signaling flows, which are forwarded to the VBSM, from the data flows, which are sent to a load balancer (LB) that distributes them to the available acquirers instantiated in the node by the orchestrator. The structure of the generic *acquirer* is shown in Fig. 3. It is constituted by a DPI, which identifies the different channels emitted by the CPs, and sends each of them to a specific *flow replicator*, each representing the root of the VST instantiated to transmit one channel. To this purpose, a flow replicator creates as many copies of the channel flow as the number of forwarder nodes at the first level of the VST. An additional copy is sent to the SS to create a stored copy of the channel, if requested by at least one client. Hence, each flow is sent along the different levels of the VST, crossing one forwarder for each level, up to the last level, constituted by the streamers.

The forwarder structure, shown in Fig. 3, is very easy. In fact, it is constituted by a flow replicator that sends one copy of the flow to each node of the lower layer of the same VST. If the input load is heavy, more than one instance of flow replicators can work in parallel, and the input LB distributes the arriving load to them.

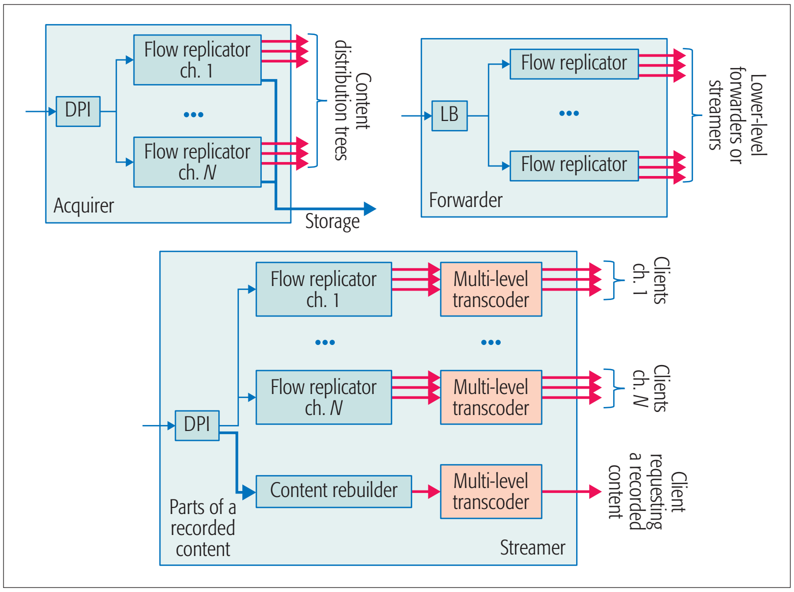

Figure 3. NFV graphs of the main platform elements.

The streamers are the leaves of the VSTs, and are in charge of forwarding the received flows to the clients connected to them. Moreover, when a client requests a recorded content, the streamers are in charge of rebuilding the content by elaborating the parts received from some nodes of the SS, for example, by applying an inverse FEC if the SS has split the content using an FEC encoder. The streamer structure is shown in Fig. 3. A DPI is at the ingress of the chain to distinguish between live streaming flows, which are sent to the replicators according to the channel to which they belong, and the parts of a recorded content, which are sent to the cntent rebuilder to provide a requested content to a client. The output of both flow replicators and content rebuilder are then sent to a battery of transcoders to adapt the video flow quality to the attached clients.

The Provided Service

This section presents the service provided by the proposed platform, describing what happens when CPs and clients connect to the network to receive the platform service through two applications, the *CP_App* and the *Client_App*.

When a CP intends to transmit some contents, it joins the network using the *CP_App*. Once authenticated and authorized by the VBSM, the CP sends it the list of available channels and the related content schedule. Afterward, each client willing to enjoy the video broadcasting service by using the *Client_App* consults the list of channels/contents, and performs an action among *select provider*, *select channel*, *watch and/or record live channels*, and *view recorded contents*.

At the scheduled time of a given content, taking into account the current position of both the CP and the interested clients, the VBSM issues a request to the orchestrator to create the VST. Then the orchestrator sets up the acquirer, the forwarders, and the streamer nodes in such a way as to satisfy the service level agreements (SLAs) with all the involved users. Finally, all the elements are chained as described in the previous section, attaching the CP to the acquirer and the clients to the streamers, and managing the SDN switches to interconnect the acquirer, the forwarders, and the streamers. Thus, the transmission starts, and the VBSM sends a notification to all the clients, which will view a pop-up message on their application. As illustrated in the previous section, video flows can be transcoded by the streamers according to the quality required by the clients and the characteristics of their terminal and their connections.

When at least one client needs to record an event, or the same CP decides to store a content since it is expected that this will be of interest for some clients in the future, the SS is also connected to the acquirer. The SS is in charge of the storage of the multimedia stream, which is split in different blocks according to the employed encoding policy (e.g., FEC or RLNC). Finally, the obtained blocks are sent to the storage elements composing the SS in such a way that it will be possible to rebuild the multimedia content upon a playout request from a client. This is done by using a subset of blocks quickly chosen to minimize the latency toward the requesting client. These blocks are sent to the streamer to which the client is attached, and the streamer is in charge of both the rebuild process and the transmission of the content to the client.

A Proof of Concept on the JOLNET

The JOLNet project is an SDN testbed network, realized through the deployment of a dedicated infrastructure between different JOL research centers, co-located within the following Italian university campuses: the Politecnico of Turin (TO), Politecnico of Milan (MI), Scuola Superiore Sant'Anna of Pisa (PI), University of Catania (CT), University of Trento (TN), and the research center TI-lab in Turin. The JOLNet network connects the different JOL locations in a full mesh topology, realized over a physical star network of national transmission high-speed connections.

JOLNet has been realized in collaboration with Cisco (which mentions it as a use case of success on its website [14]), and allows the creation of stable and logically separated environments for the development and in-network deployment of different proofs of concept.

The main hallmark of JOLNet could be summarized as "being a physical geographic network that allows the development of services thereby evaluating potentiality and impacts of a concept as close as possible to a real production network environment."

The entire JOLNet architecture is composed of a set of seven point of presence (PoP) infrastructure nodes, allowing end users to access the whole network infrastructure [15]. Each PoP node is constituted by two OpenFlow-enabled switches, one working toward the local LAN as customer premises equipment (CPE) and another toward the core of the JOLNet. The architecture also provides the functionality of network virtualization by means of the Flowvisor, a network virtualization layer capable of partitioning the experimental infrastructure resources, creating a suitable number of independent contexts called *slices*.

Furthermore, the nodes are equipped with tools allowing the configuration and orchestration of the network and IT assets, such as the OpenStack framework. This last framework allows the instantiation of appropriate applications hosted on VMs within the IT resources (compute, storage). Usually, those applications perform functions, which typically involve treatment of the L4–L7 traffic needed to support the offer of network and application services.

The PoP nodes are interconnected through a mesh of dedicated links to form the backbone of the SDN network. This meshing is logically realized through appropriate tunneling technologies.

The JOLNet is deployed as a star topology with its core hub in TI-Labs. This hub hosts a Cisco Extensible Network Controller (XNC), which is the "Slice 0" controller, in charge of execution of the JOLNet resource partition, so that each user can run his/her own experiments in each network tenant as in a dedicated infrastructure. It is also responsible for the control layer of the entire infrastructure, performing authentication, authorization, and accounting (AAA) and user profile management.

In order to run the proposed platform on the JOLNet, the following actions have been performed. First, an SDN controller, realized with PoX, has been installed on the PoP node of Catania to control the network slice reserved for the experiment. Then we have connected two CPs to the same node of Catania (CP1 and CP2), each generating one video flow, two clients for the CP1 content (both connected to Venice), and three clients for the CP2 content (located at Turin, Venice, and Pisa). Then an acquirer process has been instantiated in the node of Catania, a streamer process has been instantiated on each node of the JOLNet involved in Turin, Venice, and Pisa, and a forwarder in TI-lab and Trento. The resulting platform PoC is shown in Fig. 4.

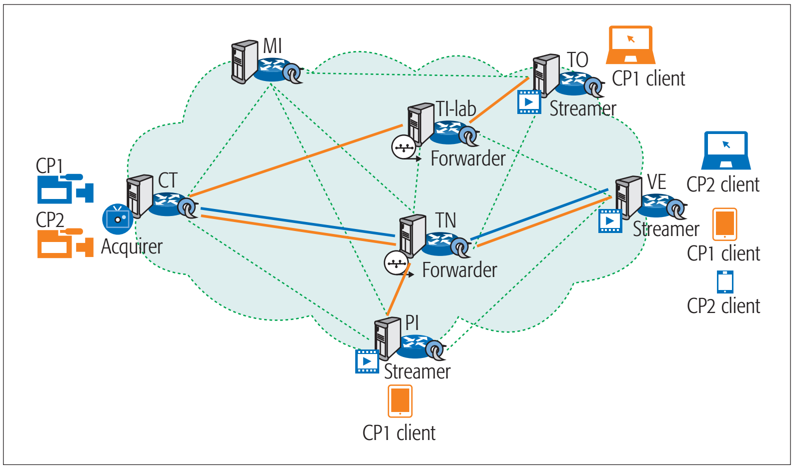

Figure 4. Proof-of-concept deployment.

Conclusions and Future Work

This article describes a use case of an SDN/NFV network, designed to allow small/medium and unusual content providers to benefit from a flexible and easy-to-deploy delivery infrastructure of live video broadcasting.

The proposed framework presents concrete advantages for the three main stakeholders of a multipoint live streaming system (i.e., content providers, telco operator, and clients). In fact, content providers do not need a specific and expensive infrastructure to transmit their video flows, but can leverage on the platform services made available by the telco operator, not only for streaming of live flows, but also to provide video flows with different levels of quality and bit rate, and even store some of them. A telco operator, on the other hand, is advantaged in terms of both CAPEX and OPEX, thanks to deployment simplification and management cost reduction. Finally, clients also gain from the proposed framework since they receive flows with lower latency and with a specified quality wherever they access the Internet, given that flows arrive directly to them, without the need to traverse remote OTT servers for replication and transcoding.

Besides an architectural description of the proposed platform, the article presents the NFV implementation of the platform, and introduces a POC that has been realized on the JOLNet network, an SDN experimental network developed by Telecom Italia to interconnect seven research labs installed within different Italian universities.

The described use case can be considered as the starting point of different research activities. As potential future work, we aim at analyzing the impact of the underlying network performance on the QoE at the application level. Another future research activity needed to optimize the performance of the proposed framework is the definition of algorithms for resource allocation, and the implementation of orchestration policies to automate the control of the whole platform.

Acknowledgments

This work has been performed in collaboration with Telecom Italia Wave Joint Open Lab, and partially supported by the In-Network Programmability for Next-Generation Personal Cloud Service Support (INPUT) project funded by the European Commission under the Horizon 2020 Programme (Call H2020-ICT-2014-1, Grant no. 644672).

References

[1] Nielsen, The Total Audience Report, Dec. 2014; http://www.nielsen.com/content/dam/ corporate/us/en/reports-downloads/2014 percent20Reports/ total-audience-report-december-2014.pdf, accessed June 2016.

[2] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015–2020 white paper, Feb. 3 2016.

[3] Ericsson ConsumerLab, "TV and Video – An Analysis of Evolving Consumer Habits," Ericsson Consumer Insight Summary Report, Aug. 2012; https://www.ericsson.com/res/docs/2012/consumerlab/tv_video_consumerlab_report.pdf, last access June 2016.

[4] D. Ciullo *et al.*, "Network Awareness of P2P Live Streaming Applications: A Measurement Study," *IEEE Trans. Multimedia*, vol. 12, no. 1, 2009, pp. 54–63.

[5] "Software-Defined Networking: The New Norm for Networks," white paper; https://www.opennetworking.org/, last accessed June 2016.

[6] "Network Functions Virtualisation," white paper; http://portal.etsi.org/NFV/NFV_White_Paper.pdf, last accessed June 2016.

[7] A. Manzalini and R. Saracco, "Software Networks at the Edge: A Shift of Paradigm," *Proc. IEEE SDN4FNS*, Trento, Italy, Nov. 11–13, 2013, pp. 1–6.

[8] G. Faraci and G. Schembra, "An Analytical Model to Design and Manage a Green SDN/NFV CPE Node," *IEEE Trans. Network and Service Management*, vol. 12, issue 3, Sept. 2015, pp. 435–50.

[9] European Project INPUT, www.input-project.eu, last accessed June 2016.

[10] ETSI NVF GS, Network Functions Virtualization (NFV) Infrastructure Overview, NFV-INF 001 v1.1.1, Jan 2015.

[11] G. Faraci, A. Lombardo, and G. Schembra, "A Processor-Sharing Scheduling Strategy for NFV Nodes," *J. Electrical and Computer Eng.*, vol. 2016, 2016, pp. 1–10.

[12] D. Wu *et al.*, "Streaming Video Over the Internet: Approaches and Directions," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, no. 3, Mar. 2001, pp. 282–300.
[13] ETSI NFV GS, "Network Function Virtualization (NFV) Management and Orchestration," NFV-MAN 001 v0.8.1, Nov. 2014; http://www.etsi.org/deliver/ etsi_gs/NFV-MAN/001_099/001/01.01.01_60/gs_nfv-man001v010101p.pdf, last access on June 2016..
[14] Cisco Report, "Software-Defined Networking Promises Competitive Advantage," 2014; https://www.cisco.com/ web/IT/assets/pdf/telecom_italia_v3cs_final.pdf, last access on June 2016.
[15] L. Grossi *et al.*, "SDN e NFV: Quali Sinergie?," *Notiziario Tecnico Telecom Italia*, no. 2, 2014, pp. 48–65.

Biographies

VALERIA D'AMICO (valeria.damico@telecomitalia.it) received her M.S. degree in electronic engineering from the University of Catania, Italy, and her Master's in business administration from SDA Bocconi, Milano. Currently she is the head of the TIM Joint Open Lab Department, applying an Open Innovation paradigm to corporate business development. As a senior project manager she has gained a solid track record of over 15 years in innovation project start-up, execution, and coordination, in both domestic and international environments, in the broad field of mobile communications, also contributing to the definition and assessment of the Long Term Evolution (LTE) fourth generation mobile standard through active participation to EU projects such as WINNER+ and FP7 Artist4G.

ALFIO LOMBARDO (alfio.lombardo@dieei.unict.it) received his degree in electrical engineering from the University of Catania in 1983. Until 1987, he acted as a consultant at CREI, the center of the Politecnico di Milano for research on computer networks, where he was involved as technical leader in the SEDOS and CTS-WAN/FTAM eCOST 11 TER – FDT/ABM CEC projects. In 1988 he joined the University of Catania where he is a full professor. He has been the leader of the University of Catania team in many European projects like DOLMEN, VESPER, and TREND. From 2007 to 2011 he was leader of the project Piano ICT per l'eccellenza nel settore Hi-Tech nel territorio catanese (ICT-E1) supported by the Sicilian regional government; in that context he was the scientific coordinator of the OpenLab laboratory. His research interests include distributed applications, network modeling and analysis, multimedia, green networking, microfluidic networks, software defined networks and network functions virtualization.

MARCELLO MELITA (marcello.melita@telecomitalia.it) received his Master's degree in telecommunications engineering at the University of Catania. In 2013 he joined the Joint Open Lab WAVE, TIM, where he works as a research and prototyping engineer on smart devices, ultrabroadband networks, and software defined networking. His current development and research interests are mainly in the area of software defined networks and networks functions/resources virtualization.

CORRADO RAMETTA (corrado.rametta name.surname@dieei.unict.it) received his Laurea degree in electronic engineering and Ph.D. degree in computer science and telecommunications engineering from the University of Catania in 2008 and 2012, respectively. From 2009 to 2012, he was a research engineer in the field of wireless communications with the Consorzio Nazionale Interuniversitario per le Telecomunicazioni (CNIT). Since 2012 he has worked as a post-doc researcher at the University of Catania. His research interests include wireless mesh networks, ad hoc and sensor networks, modeling and simulation of communications protocols, software defined networking, network functions virtualization, and embedded systems for multimedia applications.

GIOVANNI SCHEMBRA (giovanni.schembra@dieei.unict.it) is an associate professor at the University of Catania. From September 1991 to August 1992 he was with the Telecommunications Research Group of the Cefriel of Milan, working on traffic modeling and performance evaluation in broadband networks. He was involved in several national and EU projects. In particular, for the University of Catania he has been acting as WP leader in the NoE Newcom, and as a CNIT member he is currently working in the INPUT Horizon 2020 EU project. His research interests are mainly concerned with software defined networks, network functions virtualization, traffic modeling, cloud computing and data center management, and mobile cloud networks.