Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 98 (2016) 560 – 565

Procedia Computer Science

The 3rd International Symposium on Emerging Information, Communication and Networks (EINC 2016)

Video broadcasting services over SDN-NFV enabled networks: a prototype

Gabriele Baldoni[a], Marcello Melita[b], Sergio Micalizzi[a], Corrado Rametta[a]*, Giovanni Schembra[a], Alessandro Vassallo[a]

[a]DIEEI – University of Catania, V.le Doria 6, Catania 95125, Italy
[b]Strategy & Innovation, Joint Open Lab - WAVE, Telecom Italia, V.le Doria 6, Catania 95125, Italy

Abstract

Software Defined Networks (SDN) and Network Functions Virtualization (NFV) have been introduced in the telecommunications engineering as a new way to design, deploy and manage networking services. Working together, they are able to consolidate and deliver the networking components using standard IT virtualization technologies making, in such a way, carriers infrastructures more flexible and adaptive in respect to the needs of both end-users and service providers.
In this context, this paper presents the architecture and the related prototype of a video broadcasting service tailored for SDN-NVF enabled networks, allowing network and application functions deployment simplification and management cost reduction. In such a way, the proposed solution enables small/medium and unusual content providers to share events with a restricted number of interested users without the need of adopting a dedicated and expensive data delivery infrastructure.

Peer-review under responsibility of the Program Chairs

Keywords: Software Defined Networking; Network Functions Virtualization; Network Orchestration; On-demand services, Multimedia Applications.

* Corresponding author. Corrado Rametta, DIEEI, Univ. of Catania.
E-mail address: crametta@dieei.unict.it

Peer-review under responsibility of the Program Chairs
doi:10.1016/j.procs.2016.09.086

1. Introduction

The new paradigms of Software Defined Networks (SDN)[1,2] and Network Functions Virtualization (NFV)[3,4,5] have recently redefined the vision of the communications networking, providing network managers with a complete and programmatic control of a dynamic view of the network. 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 technology of NFV introduces an important change in the network service provisioning approach, leveraging standard IT virtualization technology to consolidate many network equipment facilities onto standard servers that could be located in data centers, network nodes or even in the end user premises[5,6,7]. Moreover, with the NFV paradigm, network functions become software applications that can easily be migrated according to specific policies aimed at optimizing energy efficiency, costs and performance.

The combined application of both SDN and NFV is strongly stimulating the interest of Service Providers and Network Operators to make the innovation cycles of networks and services faster and easier, reducing both OPEX (OPerating EXpenditure) and CAPEX (CAPital EXpenditure), thanks to the enormous possibilities of making operation processes (e.g. configuration of network devices) automatic, and network functions and services more flexible and cheaper. In fact, decoupling network functions, e.g. middle-box functionalities from dedicated hardware devices, and putting them into Virtual Machines (VMs)[8], NFV makes functions and services deployment over the network easier, faster and cheaper. Moreover, using SDN, traffic control and management functions, like routing, can be moved out of the network nodes and placed on a centralized controller software. Coupling both the approaches, a centralized entity, called Orchestrator, has a complete view of the network to manage the Control Plane and deploy Virtual Network Functions (VNFs)[9,10,11]. This will represent an enabling factor for a rapid evolution of the dynamic service chain provisioning. Many deployment scenarios of SDN and NFV could be envisioned[12,13], depending on the network segments (e.g., core or edge) and, consequently, on the exploitation time horizon (e.g., medium-long term or short term).

According to the general approach of NFV, and thanks to both the growing potential of virtualization frameworks (nowadays more flexible, effective and efficient) and the capability to dynamically and adaptively connect physical and virtual devices according to the SDN paradigm, any service of a Telco network can be virtualized. The approach of a partial or even total virtualization, intended as a virtualization of both network functions and service applications (let us clarify that with this assertion we mean all those devices that work over the layer 3 and up to the application level, and that may be located either in the Telco network or at the user side) seems to begin widely accepted by Telco Providers that show a growing interest in investigating this newborn research field. The reason is to not only reduce costs of network infrastructure deployment and management, but also promptly migrate to a dynamic and flexible network architecture that is able to quickly adapt to market and technology needs.

Leveraging on these technologies, one of the main applications that is stimulating the interest of both Telco Providers and over-the-top (OTT) providers is point-to-multipoint live video transmission[14,15,16]. In this context, this work proposes a softwarized video broadcasting service (see Fig. 1), and the related prototype deployed over a geographical SDN-NFV compliant network. The scope is to enable small/medium and unusual content providers to share events with a restricted number of interested users without the need of adopting a dedicated and expensive data delivery infrastructure and/or subscribing expensive contracts with carriers.

More in detail, Section 2 will give a brief description of the proposed architecture; in Section 3 we provide a detailed description of the realized prototype; finally, in Section 4 conclusions and future works will be illustrated.

2. Architecture overview

The platform is characterized by the following elements:

1) *Clients*: they are the end-users of the system; they connect to the TLC network and require the service to enjoy multimedia content made available by providers exploiting the platform. As it will be clarified in the next section, clients will be able to perform some actions such as *select provider*, *select content*, *play or record live contents*, and *view recorded content*. Telco operator provides them a client software (i.e. Mobile App, Web Portal or similar) thanks to which it is possible to receive multimedia content transmitted by content providers.

Based on the number of users connected and requiring contents, the Telco operator, by using a softwarized approach, sets up and manages the number of nodes and the amount of resources involved in the service.

2) *Content Providers*: with this term we refer to small, medium or unusual content providers that do not have the economic capacity - or it is out of their core business scope - to deploy their own delivery network, or subscribe dedicated *content delivery services* with TLC operators. Examples of this category are universities, conference and workshop organizing committees, small municipalities, sports clubs, training institutions and so on willing to share events with a restricted number of interested users.
3) *Application Specific Server*: it represents the front end service of the platform; more in detail, clients and content providers contact the application server whenever they require a platform service. The application server acts as a middleware between end-users and SDN-NFV network. Thus, it is a high-level management application, independent from the SDN-NFV underlying architecture, which manages:
   - Authentication, authorization and accounting (AAA) of each client accessing the platform;
   - Each request coming from clients;
   - Communications exchange with the Orchestrator to notify each request coming from clients and content providers.
4) *Orchestrator*: it runs on a dedicated server and communicates with all the platform nodes through the Telco IP network. Its goal is to allocate, migrate and terminate VMs running platform functions, and to control the traffic paths according to the run-time evolution of the network. More specifically, for that concerns the main functionalities carried out by the Orchestrator, they could be summarized as:
   a. *Software Defined Network Controller*: it is the core application of a softwarized network; it acts as a sort of operating system for the network, sending rules to the nodes about port forwarding, and choosing optimal paths for application data delivery;
   b. *Network Function Virtualization (NFV) Coordinator*: it refers to the capability of managing the lifecycle of virtual machines running streaming, forwarding and/or storage functionalities inside the Telco network. It is based on the information coming from both the Orchestration Engine and the SDN Controller. The NFV coordinator is in charge of creating, migrating and destroying VMs, communicating with the virtual machines hypervisor residing on each PE/PC (Provider Edge / Provider Core) node, and to send it the instructions regarding the execution of the VMs;
   c. *Orchestration Engine:* it is the component that gathers information about the network (topology, number of connected clients, required content, availability of network devices implementing NFV, etc.). According to specific algorithms, it decides how to manage the resources of the network devices. The implemented policy can refer to a wide range of aspects: energy consumption, end-to-end delay between source and destination of a data flow, number of hops of the path, overall number of virtual machines active at the same time, quality of service, SLA with customers, etc.
5) *SDN-NFV enabled nodes*: they are the physical elements hosting the platform and allowing software defined networking and network/service functions virtualization functionalities. For that concerns this aspect, we introduced two virtual functions: the *Acquirer* function, usually located as much close as possible to the content

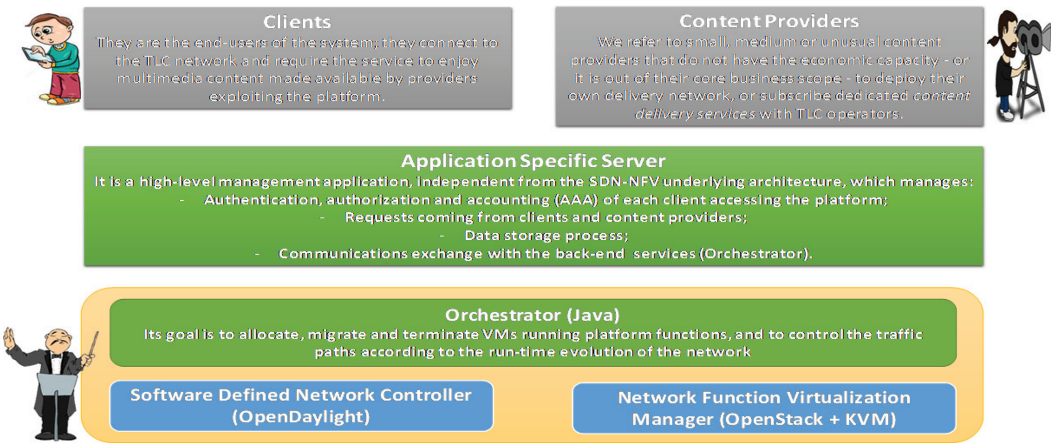

Figure 1. An overview of the proposed architecture.

provider and whose main tasks are recognizing the media streams coming from the content providers and receiving/executing transmission instructions from the Orchestrator; the *Edge Streamer* function, serving clients and exposing functionalities such as transcoding of high quality video to a lower quality video format to fit both end users and network requirement.

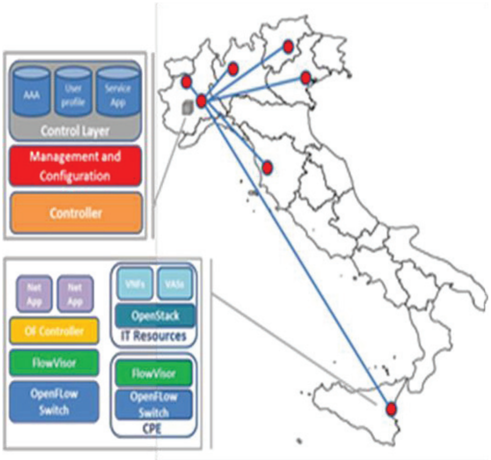

Figure 2. The JOLNet.

3. The prototype

In this section we present an implementation of the above described architecture within the SDN-NFV network deployed by TIM and realized through the deployment of a dedicated infrastructure between different research centers Joint Open Lab, co-located within the following Italian University campuses: 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 (who mentions it as a use-case of success on its website[17]) and allows the creation of stable and logically separated environments for the development and in-network deployment of different proof-of-concepts. 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 by a set of seven Point of Presence (PoP) infrastructure nodes, allowing end users to access the whole network infrastructure.

Each PoP node is constituted by two OpenFlow-enabled switches[18], one working towards the local LAN as Customer Premise Equipment (CPE), and another towards 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 the orchestration of the network and IT assets, such as the OpenStack framework[19]. This last framework allows the instantiation of appropriate applications hosted on virtual machines 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 SDN network. This meshing is logically realized through appropriate tunneling technologies.

As shown in Fig. 2, the JOLNet is deployed as a star topology with its core hub in TI-Labs, Turin. 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 own experiments in each network tenant as in a dedicated infrastructure. It is also responsible for the control layer of the entire infrastructure, performing AAA (authentication, Authorization and Accounting) and User Profile managing.

In order to run the proposed platform on the JOLNet, the following actions have been performed.

First, an SDN Controller, realized as a new Java bundle of OpenDaylight[20], has been installed on the PoP node of Catania to control the network slice reserved for the experiment.

The *Application server* is an http server written in Java language, aiming at handling the RESTful requests coming from clients and content providers and communicating them to the Orchestrator process.

The *Orchestrator* process is the main element of the platform and it is called every time a new action is required by clients or content providers to the Application server; it is a Java routine communicating with:

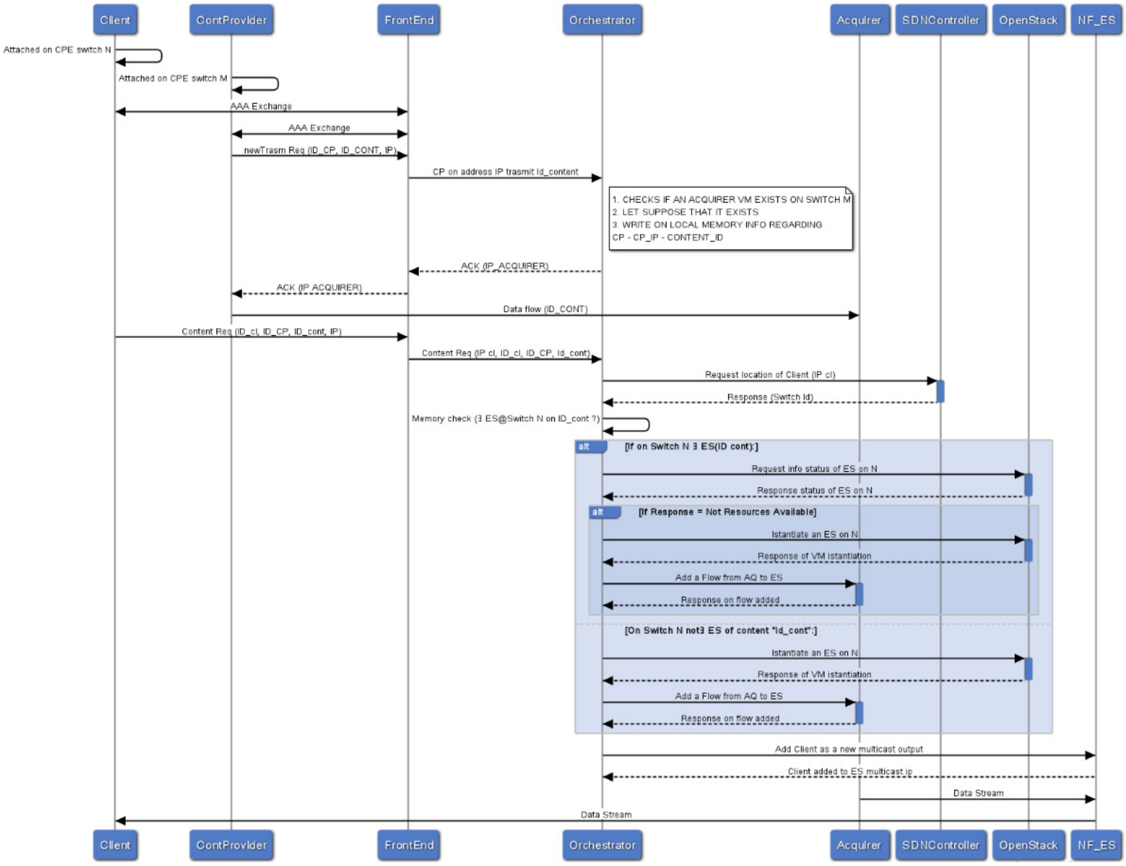

Figure 3. Messages exchange when end-users connect to the platform.

1. the SDN controller in order to establish the positioning of platform users within the network and to evaluate the links quality;
2. the hypervisor, i.e. OpenStack, with the aim of create/destroy/migrate/assign virtual functions such as the Acquirer and the Edge Streamer processes;
3. the running virtual machines in order to manage them according to the established delivery scheme.

Finally, in our testbed we developed a mobile Android client (app) with the purpose of testing the platform functionalities. The mobile app has been developed ensuring frontward compatibility with Android 4.4. It allows users to play live video contents transmitted by content providers, where video transmission is based on RTP protocol. On the other hand, in order to allow Content Providers to expose their own contents, a web application has been released. The main functionalities of this application are: providing a schedule for the future events to be broadcasted with a relative description; streaming a content to a destination *Acquirer node*, according to the rules set by the Orchestrator, towards the couple IP address-port number advertised by the front-end process and dedicated to each streaming flow. The Application server (front-end) and the Orchestrator have been instantiated on a server connected to Milan Node; templates of Acquirer and Edge streamer functions have been created and uploaded in OpenStack in order to make them available when requested by the Orchestrator.

Figure 3 shows the messages exchange among the platform elements when two end-users, a content provider and a client, access the platform to respectively transmit and receive a data stream.

4. Conclusions and future works

This paper presents a softwarized live streaming platform designed to allow small/medium and unusual content providers to benefit of a cheap, smart and easily deployable delivery infrastructure, without the need of expensive service subscription with the Telco operator. The basic idea is to apply an SDN-NFV approach to realize this smart and flexible architecture that each Telco operator can exploit to offer content delivery services to its customers that, in such a perspective, are both end users and content providers.

A proof-of-concept has been realized by exploiting a full SDN geographical network deployed by TIM SPA. Open source and free software such as OpenStack and KVM have been exploited as hypervisor and virtual machine lifecycle manager, OpenDaylight as SDN controller and proprietary Java software to realize the live streaming application server (front-end), the network orchestration (back-end server), the client app for Android smartphones, and two simple web app to provide services to content providers and PC clients.

Future works will focus on the evaluation of the application-related key performance indicators (KPIs) and the implementation of an orchestration engine based on an analytical model.

Acknowledgements

This work has been performed in collaboration with Telecom Italia Joint Open Lab WAVE, and partially supported by the INPUT (In-Network Programmability for next-generation personal cloUd service supporT) project[21] funded by the European Commission under the Horizon 2020 Programme (Call H2020-ICT-2014-1, Grant no. 644672).

References

1. White paper on “Software-Defined Networking: The New Norm for Networks”, available at *https://www.opennetworking.org/*.
2. Kreutz, D.; Ramos, F.M.V.; Esteves Verissimo, P.; Esteve Rothenberg, C.; Azodolmolky, S.; Uhlig, S., "Software-Defined Networking: A Comprehensive Survey," in Proceedings of the IEEE , vol.103, no.1, pp.14-76, Jan. 2015.
3. White paper on “Network Functions Virtualisation”, available at *http://portal.etsi.org/NFV/NFV_White_Paper.pdf*.
4. A. Manzalini et al., “Software-Defined Networks for Future Networks and Services,” White Paper based on the IEEE Workshop SDN4FNS.
5. Network Functions Virtualisation – White Paper #3, “Network Operator Perspectives on Industry Progress,” available at *https://portal.etsi.org/ Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf*.
6. A. Lombardo, A. Manzalini V. Riccobene, G. Schembra, “An Open Architecture for Software Defined Services at the Edge,” EuCNC2014, Bologna, Italy, June 23/26, 2014.
7. G. Faraci and G. Schembra, “An analytical model to design and manage a green SDN/NFV CPE node,” IEEE Transactions on Network and Service Management, vol. 12, no. 3, pp. 435–450, Sept 2015.
8. G. Faraci, A. Lombardo, G. Schembra, “A Processor-Sharing Scheduling Strategy for NFV Nodes,” Journal of Electrical and Computer Engineering, vol. 2016, 2016, pp. 1-10.
9. A. Lombardo, A. Manzalini, G. Schembra, G. Faraci, C. Rametta, V. Riccobene, "An open framework to enable NetFATE (Network Functions at the edge)", in Proc. of 1st IEEE Conference on Network Softwarization (NetSoft), 2015, 13-17 April 2015.
10. ETSI GS NFV 002, “Network Functions Virtualization (NFV): Architectural Framework”, 10-2013, available at *http://www.etsi.org/ deliver/etsi_gs/nfv/001_099/002/01.01.01_60/gs_nfv002v010101p.pdf*.
11. P. Costa, M. Migliavacca, P. Pietzuch and A. L. Wolf, “NaaS: Network-as-a-Service in the Cloud,” in Proc. of Hot-ICE’12, April 24 2012, San Jose, CA.
12. ETSI GS NFV 001, “Network Functions Virtualization (NFV): Use Cases”, 10-2013, available at *http://www.etsi.org/deliver/etsi_gs/ nfv/001_099/001/01.01.01_60/gs_nfv001v010101p.pdf*.
13. ETSI – PoC, “NFV Proofs of Concept,” available at *http://www.etsi.org/ technologies-clusters/technologies/nfv/nfv-poc*.
14. Julius Rückert , Jeremias Blendin, David Hausheer, “Software-Defined Multicast for Over-the-Top and Overlay-based Live Streaming in ISP Networks,” Journal of Network and Systems Management, April 2015, Volume 23, Issue 2, pp 280-308.
15. V. D’Amico, A. Lombardo, M. Melita, C. Rametta, G. Schembra, “An SDN/NFV Telco Operator Platform for Video Broadcasting,” in IEEE Communications Magazine, Special Issue on “SDN Use Cases for Service Provider Networks”, October 2016.
16. H. Koumaras, M. A. Kourtis, C. Sakkas, G. Xilouris, S. Kolometsos, "In-service Video Quality assessment based on SDN/NFV techniques," in Proc. of IEEE ICT2016, Thessaloniki, Greece, 16-18 May 2016.
17. https://www.cisco.com/web/IT/assets/pdf/telecom_italia_v3cs_final.pdf
18. A. Lara, A. Kolasani, and B. Ramamurthy, “Network innovation using OpenFlow: A survey,” Communications Surveys Tutorials, IEEE, vol. 16, no. 1, pp. 493–512, First 2014.
19. http://www.openstack.org/
20. OpenDaylight, “OpenDaylight: A Linux Foundation Collaborative Project,” 2013. [Online]. Available: http://www.opendaylight.org
21. INPUT Project website available at http://www.input-project.eu/