A service platform architecture enabling programmable edge-to-cloud virtualization for the 5G Media industry

Stamatia Rizou
Singular Logic S.A.
Athens, Greece
srizou@singularlogic.eu

Panagiotis Athanasoulis
Singular Logic S.A.
Athens, Greece
pathanasoulis@ep.singularlogic.eu

Pasquale Andriani
Engineering Ingegneria Informatica S.p.A.
Rome, Italy
pasquale.andriani@eng.it

Francesco Iadanza
Engineering Ingegneria Informatica S.p.A.
Rome Italy
francesco.iadanza@eng.it

Gino Carrozzo
Nextworks
Pisa, Italy
g.carrozzo@nextworks.it

David Breitgand
IBM Research --
Haifa, Israel
davidbr@il.ibm.com

Avi Weit
IBM Research --
Haifa, Israel
weit@il.ibm.com

David Griffin
University College London
London, United Kingdom
d.griffin@ucl.ac.uk

David Jimenez
Universidad Politecnica De Madrid
Madrid, Spain
djb@gatv.ssr.upm.es

Ugur Acar
Netas Telekomunikasyon Anonim Sirketi
Istanbul, Turkey
uacar@netas.com.tr

Oscar Prieto Gordo
RTVE
Madrid, Spain
oscar.prieto@rtve.es

***Abstract*— Media applications are amongst the most demanding services in terms of resources, requiring huge network capacity for high bandwidth audio-visual and other mobile sensory streams. The 5G-MEDIA project aims at innovating media-related applications by investigating how these applications and the underlying 5G network should be coupled and interwork to the benefit of both. The 5G-MEDIA approach aims at delivering an integrated programmable service platform for the development, design and operations of media applications in 5G networks by providing mechanisms to flexibly adapt service operations to dynamic conditions and react upon events (e.g. to transparently accommodate auto-scaling of resources, VNF re-placement, etc.). In this paper we present the 5G-MEDIA service platform architecture, which has been specifically designed to enable the development and operation of services for the nascent 5G media industry. Our approach delivers an integrated programmable service platform for the development, design and operations of media applications in 5G networks.**

Keywords*— *Network Function Virtualization, edge-cloud, management and operation framework, 5G networks for media applications

I. Introduction

Typically, media applications have strict Quality of Service (QoS) demands and impose hard challenges on the management of the underlying computing and network resources. Recent technological advances in the 5G (5th generation wireless networks) domain promise to unlock the potential of the media industry by offering high quality media services through dynamic efficient resource allocation. However, existing solutions for efficient resource management and programming of 5G applications fail to provide a truly integrated approach when applied to the 5G media industry. In fact, most of the capabilities of the current Software Defined Networks (SDN) and Network Function Virtualization (NFV) enabling tools for 5G networks cover the runtime lifecycle management of generic network functions and the orchestration of resources at the network rather than application layer, focussing on radio access, fronthaul/backhaul, edge and core network segments.

In this paper we present the 5G-MEDIA service platform architecture, which has been specifically designed in the context of the 5G PPP 5G-MEDIA project to enable the development and operation of services for the nascent 5G media industry. Our approach delivers an integrated programmable service platform for the development, design and operations of media applications in 5G networks. Each media service comprises a chain of media-specific and network-specific atomic services, all interconnected to deliver an expected output to the end user (media consumer). The resulting graph (i.e., a graph where nodes refer to computing tasks and edges refer to network communication links) is referred as Media Service Forwarding Graph (MSFG). During the design, development and test phase, the platform provides appropriate programming tools to abstract the details of the underlying 5G infrastructure and allow developers to focus on the functionality of the services. Once the media service is deployed in the virtualized infrastructure, the 5G-MEDIA platform provides mechanisms to flexibly adapt service operations to dynamic conditions and react upon events (e.g. to transparently accommodate auto-scaling of resources, Virtual Network Function (VNF) re-placement, etc.).

The rest of the paper is organized as follows: In Section II, we briefly discuss background technologies and main innovation aspects of the 5G-MEDIA architecture. In Section III, we present the overall 5G-MEDIA architecture, comprising of the Software Development Kit (SDK), presented in Section IV and the Service Virtualization Platform, presented in Section

V. In Section VI, we conclude this paper and present future work.

II. Background & Innovation Aspects

Existing ETSI NFV Management and Orchestration (MANO) frameworks (e.g. ETSI Open Source MANO (OSM) [8], OpenBaton [10], OPNFV [11], SONATA [12]) provide baseline functionalities for the management and orchestration of generic VNFs. In the context of 5G-MEDIA, we will rely on ETSI OSM for the realization of MANO functionalities for media services. The main innovation aspects of the platform lie in: i) the coupling of Function-as-a-Service (FaaS) with VNF management to support media applications; ii) the Media Service Monitoring, Analysis, Planning and Execution (MAPE) component to adapt media applications in dynamic network conditions iii) the design of a novel 5G Application and Service Catalogue, which is NFV MANO platform-agnostic in terms of formats and syntax for Network Service (NS) descriptors and VNF Package information model. In the next paragraphs, we briefly discuss each of the innovation aspects:

A. FaaS Integration

Since its introduction by Amazon in 2014 [9], FaaS has generated a tremendous amount of interest both in industry and academia. To the best of our knowledge, prior to 5G-MEDIA, FaaS has not been applied to NFV. Motivated by media use cases, such as short lived media intensive gaming sessions, mobile journalism and content distribution, we observe that FaaS is of great benefit to these scenarios, because it allows event-driven on-demand VNF instantiation and execution and their seamless elasticity in contrast to a traditional Virtual Machine (VM) oriented approach, in which virtual appliances are continuously running leading to low utilization. The 5G-MEDIA platform is pioneering the application of FaaS to VNF management, complementing traditional VM based VNFs with FaaS based media specific functions, aiming at dramatically reducing development cycles and slashing operational costs to 5G-MEDIA users. For the realization of the FaaS integration, Apache OpenWhisk [6] will be used as background technology.

B. Media Service MAPE

The MAPE components will provide dynamic adaptation of MSFGs to different conditions. The Cognitive Network Optimization (CNO) Engine, which constitutes the main intelligence component for optimization tasks will execute machine learning techniques to dynamically establish and update the MSFG. Depending on the input data from the monitoring component and optimization goals, the optimization mechanism will make decision on which algorithm should be used. For instance, Integer Linear Programming can be used for small input datasets in where optimal solutions can be obtained in an acceptable time. Otherwise, heuristic algorithms are used to find close-to-optimal solutions for larger input datasets. In addition, the optimization mechanism will take into account policies (e.g. latency, maximum resource utilization, budget cost, etc.) defined by the service providers. This component will be extend the results of the COGNET project [13] with new Machine Learning algorithms specifically adapted to meet the needs of media applications.

C. Application and Service Catalogue

To date, the high fragmentation in the way NS descriptors and VNF packages are represented across the different MANO frameworks is impacting the ability of NFV developers and service providers to produce a portable offering of their virtual applications and services. In fact, different formats (e.g. JavaScript Object Notation (JSON) vs YAML Ain't Markup Language (YAML)) exist to describe Network Services in NFV, different packetized contents (e.g. monitoring parameters, software images, etc.) and different procedures are used for specifying and configuring service components (e.g. JuJu charms, cloud-init, other Day1/Day2 configuration recipes, etc.), and each ETSI NFV MANO framework available in the state of the art (e.g. ETSI OSM, OpenBaton, Tacker, Rift.Ware, etc.) adopts a proprietary (though ETSI standards inspired) approach. In fact, there is a proliferation of Network Function Virtualization Orchestrator (NFVO) specific descriptors and packages for the various platforms used by service providers, with per-framework solutions to encode variables in descriptors (e.g. for NVF configuration, monitoring, generation of outputs), to encode auto-scaling rules, to support Network Function Virtualization Infrastructure (NFVI) specific extensions (e.g. SR-IOV, DPDK, containers instead of VMs, etc.).

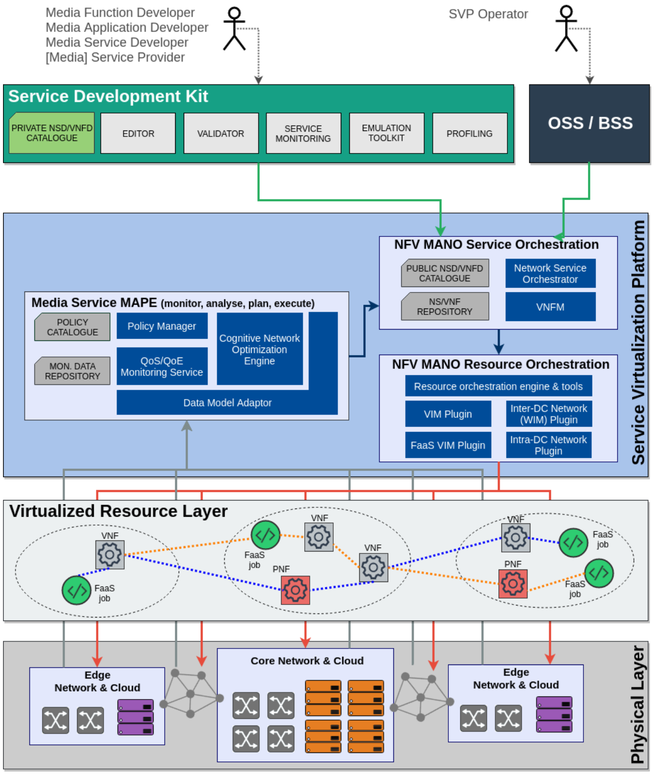
The current status remains for the time being acceptable because the maturity of NFV MANO solutions is still in early stages and most of the deployments are in a context of single MANO operational domains. However, the advent of end-to-end 5G networks composed of multiple virtualized infrastructures split across different administrative domains calls for:

- a new way to compose services from different providers,
- mechanisms for NS orchestration and delivery not limited by the characteristics and capabilities of the provider's NFVI (e.g. its geographical coverage, the quantity of resources available at the edge and at the core of the network)
- the possibility to use Virtual Functions from federated MANO systems (e.g. to complement a domain's catalogue of NSes and VNFs with items made available by other federated domains)
- the possibility to enable the vertical application developer (i.e. the customer of the NFV Network Operator) to onboard new virtual applications for their specific services.

The design of the 5G-MEDIA Catalogue will respond to these challenges, by providing a NFV-MANO platform agnostic solution.

III. 5G-MEDIA Service Platform Architecture

As shown in Figure 1, the main building blocks comprising the 5G-MEDIA architecture include an Application/Service Development Kit that enables access to media applications development tools and a Service Virtualization Platform (SVP) that hosts the components related to the ETSI MANO framework, the Virtual Network Functions and the Media Application Repository as well as the generic components that can be used across many applications (such as the monitoring, optimization and FaaS tools) and enable integration with different NFVIs that provide the cloud resources from different operators to host media-specific VNFs.



Media Function Developer
Media Application Developer
Media Service Developer
[Media] Service Provider
SVP Operator
Service Development Kit
PRIVATE NSD/VNFD CATALOGUE
EDITOR
VALIDATOR
SERVICE MONITORING
EMULATION TOOLKIT
PROFILING
OSS / BSS
Service Virtualization Platform
NFV MANO Service Orchestration
PUBLIC NSD/VNFD CATALOGUE
Network Service Orchestrator
NS/VNF REPOSITORY
VNFM
Media Service MAPE (monitor, analyse, plan, execute)
POLICY CATALOGUE
Policy Manager
Cognitive Network Optimization Engine
MON. DATA REPOSITORY
QoS/QoE Monitoring Service
Data Model Adaptor
NFV MANO Resource Orchestration
Resource orchestration engine & tools
VIM Plugin
Inter-DC Network (WIM) Plugin
FaaS VIM Plugin
Intra-DC Network Plugin
Virtualized Resource Layer
VNF
FaaS job
PNF
Physical Layer
Edge Network & Cloud
Core Network & Cloud
Edge Network & Cloud

Figure 1: 5G-MEDIA High Level Architecture

The SDK provides a set of open source tools to support the rapid development of media applications using the DevOps approach. In particular, these tools allow to define MSFGs (also using already existing VNFs, stored in the VNF Repository), to proof and package the various functions as well as to emulate behaviours of the virtualized infrastructure, to accelerate application development and provide a testing environment to be utilized prior to service deployment in the runtime Service Virtualization Platform. The SDK tools enable also the use of the innovative concept of the FaaS approach, where developers need not care about the low-level details related to the virtual computing and storage infrastructure (e.g. virtual server profiling in terms of CPU, RAM, etc.), thus drastically contributing to reduce the service creation time cycle and maintenance effort. In this line, the service developers will be able to create the so-called FaaS VNFs, i.e., VNFs that are instantiated upon the detection of specific events. The combination of the FaaS approach with the VNF packaging and the enablement of inserting FaaS VNFs in a typical MSFG is one of the main innovation aspects of the proposed 5G-MEDIA approach.

The SDK interacts with the SVP, which hosts the components related to the ETSI MANO framework (NFV Orchestrator, VNF Manager(s), Infrastructure Manager(s) and Virtualization & Abstraction Layer), the VNF and NetApp Repository as well as generic components that can be used by many applications (e.g. monitoring and optimization components). The monitoring components provide aggregated monitoring values for metrics specified by the developer per VNF. These components are part of the generic monitoring system, which includes probes/taps to monitor the performance of the infrastructures (NFVIs) and the applications themselves. In this respect, the monitoring system and, specifically, the monitoring probes must be compatible with several NFVIs (OpenStack, VMware, etc.). The measured performance metrics are directly used by the Service/VNF orchestrator and the VNF Manager to react, following the orchestration rules of each network service and/or triggering the deployment of FaaS VNFs. In this line, a specific innovation of the 5G-MEDIA project is the integration of the Cognitive Network Optimizer (CNO) within the SVP. The CNO comprises mechanisms that take advantage of machine learning techniques and optimization policies management and will trigger the dynamic instantiation of MSFGs on the different NFVIs. The CNO is able to respond to dynamic changes of the environment (e.g., location change of end users, varying QoS demands) and to adapt the deployment of MSFGs seamlessly to continuously meet expected QoS requirements. Finally, in terms of the supporting physical infrastructure, the proposed architecture considers that several cloud-based NFVIs will be connected to the SVP, allowing for the instantiation of network applications closer to the user (edge computing paradigm). Again, as the VNFs form a MSFG, part of a network application may consist of several components deployed in more than one Virtual Infrastructure Manager (VIMs)/NFVIs also connecting the end-users to the 5G-MEDIA ecosystem through mobile devices, tablets and other resource-constrained devices.

In the next paragraphs, we present the details of the main building blocks of the 5G-MEDIA architecture, i.e., the 5G-MEDIA Service Development Kit and the Service Virtualization Platform.

IV. Service Development Kit

The 5G-MEDIA SDK is a set of tools that allows for the creation of applications, network services or functions and supports developers in implementing, packaging, deploying, monitoring or analyzing the software. The SDK provides a programming model for application developers by providing several functionalities such as a private catalog of Network Service descriptors (NSD) and Virtual Network Function Descriptors (VNFD), editor, validator, service monitoring, emulation toolkit etc., which allows defining complex media services consisting of multiple VNFs. The editor is the main user interface (UI) for developing, validating and onboarding media applications' or network services' by building MSFGs and deploying the NS package to the private/platform catalog and instantiating the NS in the SVP. In addition, the editor UI visualizes topological dependencies and the interconnection of the involved VNFs in the MSFG and descriptions of individual VNFs. The validator is responsible for validating MSFG from the perspectives of syntax, integrity and topological connectivity. A Service Monitoring tool guides the developer by providing performance data of the media application and its components in a quantitative manner. This tool collects and visualizes network monitoring data of a service.

Depending on the SVP monitoring functionalities, a generic monitoring tool includes both resource consumption metrics of a VNF (i.e. CPU, memory, etc.) and VNF specific traffic analysis metrics (i.e. hit ratio etc.). Profiling tool allows developers to stage a service in a local SDK environment to overcome possible issues related to either configuration or implementation of a service before being deployed in the production environment of the SVP. The performance of NFV based services including VNF implementations relies on different metrics such as the underlying software platform (e.g. programming language, compiler, etc.), implementation quality, the architecture of underlying hardware platform (e.g. memory, number of cores, storage, etc.), the potential variability of the interconnecting networks between VNFs due to their flexibility of deployments on different locations in the network. The emulator tool enabled developers to prototype and test complete network services in realistic end-to-end multi-PoP scenarios prior to deployment. The emulation platform allows the direct execution of real network functions, packaged as containers, in emulated network topologies running locally on the network service developer's machine. The SDK also extends OSM's vim-emu to support FaaS development and staging. Finally, Packaging tools are integrated in the SDK as part of the Operation Support Tools in a seamless manner. Following a DevOps model, these tools automatically builds and collects the necessary software artefacts and generates packages in the form of lightweight containers, unikernels images or virtual appliances to be available in the Private Catalog for staging.

V. Service Virtualization Platform

The Service Virtualization Platform consists of three main components: i) the Media Service MAPE component, enabling continuous optimization of MSFGs; ii) the NFV MANO Service Orchestration, encompassing the NSD/NFD Catalogue and NS/NF repositories together with the NFV MANO stack functions (service orchestrator, NFVM); iii) the NFV MANO Resource Orchestration, enabling the integration with NFVIs through different VIM plugins. In the next paragraphs, we present each component, focusing on the main contributions of the 5G-MEDIA platform on top of the NFV MANO stack.

A. Media Service MAPE

The Media Service Monitoring-Analysis-Planning and Execution components provide the intelligence behind the MANO to allow it to dynamically manage and provide infrastructure resources for the deployed media services according to observed changes in user demand patterns, availability and performance of network and computational resources.

QoS/QoE Monitoring collects data from running NSs/VNFs and the SDN controller in the underlying OpenStack, VMWare and OpenNebula NFVIs and shares data with the Cognitive Network Optimizer through a distributed message queue based on Apache Kafka to enable near real-time analysis. The CNO implements an analysis and prediction engine that uses raw measurement data and employs statistical analysis based on machine learning and optimization techniques for classifying network and computational resource service status, predicting/forecasting future resource conditions, and triggering corrective actions for the running NSs. Its machine learning engine is based around algorithms for data dimensionality reduction, feature selection/extraction, traffic classification, anomaly detection, performance degradation detection and demand prediction. Multi-objective optimization algorithms use the derived statistics and predictions to determine the assignment of the available network and computational resources to maximize performance within operator-defined cost and efficiency constraints.

Algorithms for the optimization of media services are of four main types:

- Service placement optimization to determine which VNFI instance/edge node should house each VNF for a NS by trading-off cost with performance of the network and computational infrastructure.
- MSFG optimization to determine which instances of VNFs should be interconnected to meet performance and cost objectives for specific user session requests.
- Infrastructure adaptation to overcome streaming difficulties, e.g. to reserve network capacity, allocate greater computational capacity for stream processing, establish expedited paths or reroute flows to avoid congested parts of the network.
- Application-specific adaptation and intelligent network-wide congestion avoidance, for example to configure the capturing or transcoding of 3D models to defined quality levels to match dynamically varying network throughput capabilities and available processing capacity along the VNFI nodes and clusters implementing the MSFG instance.

The decisions of the CNO are conveyed to the northbound API of the MANO service orchestrator that enforces them in the resources of the underlying NFVIs: through the management of the instantiation of VNFs and through the dynamic control of MSFGs and load balancing decisions.

B. NFV MANO Service Orchestration

In 5G-MEDIA, we introduce the concept of the *5G App and Service Catalogue*. This new functional element is designed to be NFV MANO platform-agnostic in terms of formats and syntax for NS descriptors and VNF Package information model. The catalogue uses a novel generalized and extendible format for representing NSs and VNFs, and it is capable to onboard NFV service elements as well as Mobile Edge Cloud (MEC) media applications and services and other virtual applications such as SDN applications, and functions implementing the FaaS paradigm. To implement the generalized catalogue packages description, we use the latest ETSI NFV standards for NS/VNF descriptors [1][2], VNF packages [3] and the MEC Application descriptors [4].

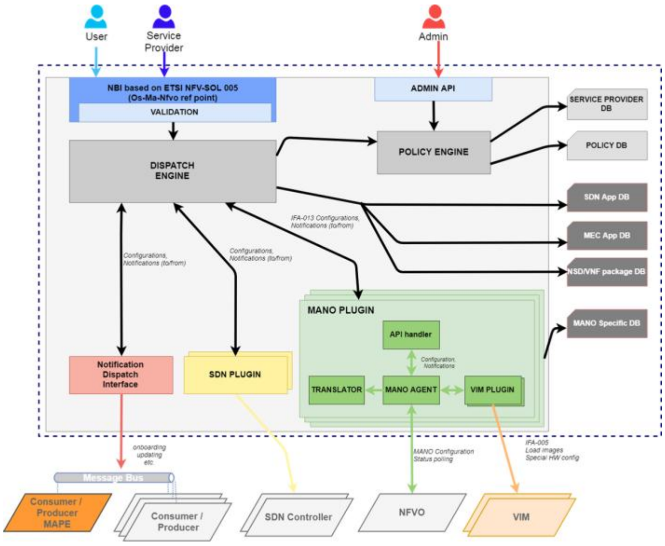


Figure 2: High-level design of the 5G App and Service Catalogue

The high-level design of the 5G App and Service Catalogue is depicted in Figure 2. This design reflects the modularity and the adaptability of the NS and VNF information models to be adopted. In fact, The northbound interface of the catalogue is designed to support the ETSI NFV specification for the Os-Ma-Nfvo reference point [5] thus offering operations for NS and VNF/App package management (e.g. upload, fetch, update, delete and query). An Admin Interface is also offered to allow the management of users with configuration of the related policies. The southbound interface of the Catalogue is composed of different plugins capable of handling the translation of the generalized package/descriptor into the specific format expected at the underlying orchestrator (both NFV and SDN orchestrators could be supported) and actuating onboarding/management operations on the target virtualization platform. In particular, the MANO plugins include:

- a translation module capable responsible for translating the generic descriptor in the format expected at the underlying MANO Service Orchestrator (e.g. packages compatible with the OSM information model specification),
- a set of VIM plugins (e.g. OpenStack plugin, OpenWhisk plugin etc.), one for each VIM in the NFVI administrated by the target MANO stack, for uploading images in the VIM images' storage,
- a MANO agent for collecting feedbacks about onboarding and instantiation operations as well as for notification about, for instance, new VIM instances or new capabilities supported by the MANO framework.

The 5G App and Service Catalogue design foresees also the implementation of a Notification Dispatch Interface for sending service and application specific notifications to a set of consumers listening on a notification bus. In 5G-MEDIA, a specific consumer on the message bus is the MAPE component, which retrieves application specific monitoring parameters used to initiate monitoring jobs once the service/application is instantiated through the MANO stack.

C. *NFV MANO Resource Orchestration*

The NFV MANO Resource Orchestration components allow the integration of the platform with different NFVIs through the support of different VIMs, such as OpenStack, VMWare, OpenVIM and other. In the following, we present the design of the FaaS VIM Plugin, which is one of the main innovation aspects of the 5G-MEDIA platform as discussed in Section II.

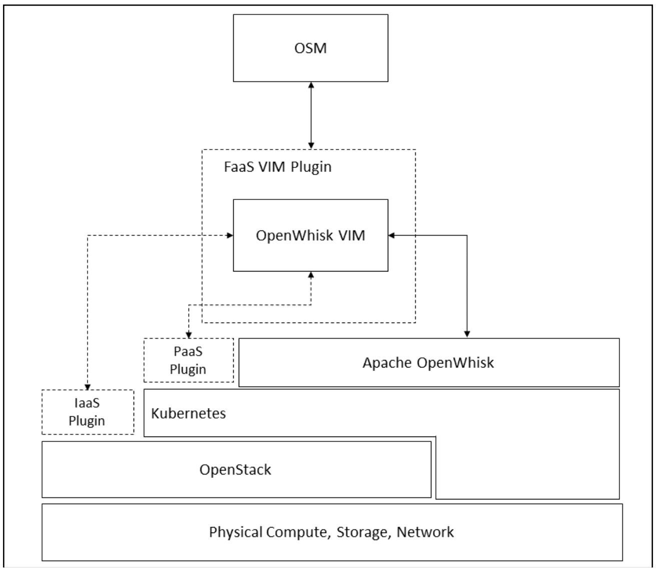


Figure 3: Software Architecture of the FaaS integration Reference Implementation

To harness cost-efficiency benefits and allow for a new programming model, in which a developer only needs to focus on their code rather than on the infrastructure, we propose an ETSI MANO compatible integration of FaaS and NFVI architecture and a reference implementation that uses Apache OpenWhisk [6], Kubernetes [7], and Open Source Mano (OSM) [8]. Figure 3 shows software architecture of FaaS integration with VNFI that is being developed in 5G-MEDIA. The centerpiece of the architecture is a new FaaS VIM OSM Plugin, which surfaces the NFV MANO interface on the northbound and uses CRUD API of Apache OpenWhisk on the southbound. In 5G-MEDIA, Apache OpenWhisk uses Kubernetes as its backend container orchestrator engine. The reason for using Kubernetes instead of the OpenWhisk native container orchestration mechanism is three-fold. First, we take an advantage of Kubernetes support for networking, allowing OpenWhisk actions to communicate with each other as prescribed by the forwarding graph of a service. Secondly, using Kubernetes as a backend, accrues portability. Thirdly, we can potentially take an advantage of the Kubernetes declarative management policies mechanisms to deal with difficult issues such as fault tolerance.

This architecture allows FaaS to be seamlessly integrated with the rest of the platform. No changes to the ETSI MANO concepts are required and VNF descriptors of FaaS VNFs remain the same. The FaaS VIM plugin emulates the behavior of a FaaS action that implements a VNF as if it were a regular virtual appliance. To take advantage of FaaS, a FaaS-based VNF is not started by default when a network service is started. A FaaS VNF is always started by the NFVI orchestrator in response to application level events. As an example, consider a use case where two players engage in a media intensive e-sport game bout. When an event of interest occurs, a clip for spectators is created from the last few seconds of the media stored in a VNF buffering this media. The clip creation VNF can be implemented as a FaaS VNF in 5G-MEDIA. It will only be invoked on demand in response to the event of interest generated by the gaming server.

VI. Conclusions

In this paper, we have presented the design of the main architectural components of the 5G-MEDIA service platform including the Service Development Kit and the Service Virtualization Platform. The realization of the 5G-MEDIA platform leverages on existing open source software for SDN/NFV networks, 5G PPP phase 1 project results as well as 5G/NFV software assets contributed by the project partners. The project will focus on the integration of background assets and their extensions to meet the needs of media industry. The platform has been designed to accommodate future extensions and potential integration with third-party tools to create added value services and maximize its impact.

Acknowledgment

This work has been realized in context of the 5G-MEDIA project. This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 761699.

References

[1] ETSI Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Network Service Templates Specification, IFA014, Available online: http://www.etsi.org/deliver/etsi_gs/NFV-IFA/001_099/014/02.04.01_60/gs_NFV-IFA014v020401p.pdf

[2] ETSI Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; VNF Descriptor and Packaging Specification, IFA011, Available online: http://www.etsi.org/deliver/etsi_gs/NFV-IFA/001_099/011/02.04.01_60/gs_NFV-IFA011v020401p.pdf

[3] ETSI Network Functions Virtualisation (NFV) Release 2; Protocols and Data Models; VNF Package specification, SOL004, Available online: http://www.etsi.org/deliver/etsi_gs/NFV-SOL/001_099/004/02.04.01_60/gs_NFV-SOL004v020401p.pdf

[4] ETSI Mobile Edge Computing (MEC); Mobile Edge Management; Part 2: Application lifecycle, rules and requirements management, MEC 010-2, Available online: http://www.etsi.org/deliver/etsi_gs/MEC/001_099/01002/01.01.01_60/gs_MEC01002v010101p.pdf

[5] ETSI Network Functions Virtualisation (NFV) Release 2; Protocols and Data Models; RESTful protocols specification for the Os-Ma-nfvo Reference Point, SOL 005, Available online: http://www.etsi.org/deliver/etsi_gs/NFV-SOL/001_099/005/02.04.01_60/gs_NFV-SOL005v020401p.pdf

[6] Apache OpenWhisk, https://openwhisk.apache.org/

[7] Kubernetes, https://kubernetes.io/

[8] Open Source Mano (OSM), https://osm.etsi.org/

[9] AWS, “Introducing AWS Lambda”, https://aws.amazon.com/about-aws/whats-new/2014/11/13/introducing-aws-lambda/

[10] Open Baton, http://openbaton.github.io/

[11] Open Platform for NFV (OPNFV), https://www.opnfv.org/

[12] SONATA NFV: Agile Service Development and Orchestration in 5G Virtualized Networks, D2.3 Updated Requirements and Architecture Design, December 2016, available at http://www.sonata-nfv.eu

[13] COGNET 5G-PPP project, http://www.cognet.5g-ppp.eu/