5GASP's approach to the onboarding, deployment and validation of 5G NetApps

Rafael Direito, Diogo Gomes
Instituto de Telecomunicações
Universidade de Aveiro
Aveiro, Portugal
rdireito@av.it.pt
dgomes@av.it.pt

Kostis Trantzas, Christos Tranoris
University of Patras
Patras, Greece
ktrantzas@ece.upatras.gr
tranoris@ece.upatras.gr

***Abstract*—Although 5G technologies are becoming widespread, the development and validation of 5G applications are still challenging. This demo paper showcases 5GASP's onboarding, deployment, and validation methodologies and how they facilitate and speed up this process.**

***Index Terms*—5G, NFV, Unified Input Data Model, Automated Deployment and Validation**

I. Introduction

Software-Defined Networks (SDN) and Network Function Virtualization (NFV) are two of the most critical key enablers of 5th generation (5G) networks, facilitating more straightforward upgrades of network services and providing more reliability and better scalability [1]. In NFV, services are offered through NetApps, a concept introduced in 5G-PPP Phase 3B ICT-41 projects that depicts specific Virtualized Network Function (VNF)-based solutions tailored to verticals.

5G Application & Services experimentation and Certification Platform (5GASP) (ICT-41-2020) is an example of a project pushing the development of NFV added value services, striving to create a community of 5G enthusiasts and vertical-specific researchers and developers that shall broadcast knowledge regarding (i) how to design and develop NetApps, (ii) how to test and validate them in real-world like scenarios, and (iii) how to certify them, providing a new layer of trust before they get delivered to a Network Operator [2].

5GTango (ICT-2016-2) also addressed the validation of NetApps, introducing a Validation and Verification Platform, which provided cutting-edge mechanisms to validate VNFs and Network Services (NSs) [3]. This platform evaluates a network application against various functional tests, also considering the application's performance. Throughout the testing of an application, several metrics are collected and processed, and then presented in a graphical interface alongside the testing phase results. A different approach is used in Vital-5G (ICT-41-2020), where the testing and validation methodology is split into two phases. The first focuses on validating the technologies and frameworks used, while the second handles the confirmation of a NetApp business potential [4]. Regarding the Vital-5G's validation first phase, several methodologies overlap with the ones we used to develop the tools presented in this paper. Both 5GASP and Vital-5G aim to deliver (i) Testing as a Service, (ii) monitoring and metrics collection, and (iii) a graphical interface where the developers can obtain the results of the performed tests.

This paper demonstrates two outcomes of the 5GASP project, the NetApp Onboarding and Deployment Services (NODS) and the NetApp Validation Service, which showcase the methodologies and innovations introduced by this project concerning the onboarding, deployment, and validation of NetApps.

Section II introduces the aforementioned tools, whereas Section III illustrates their novelty. Then, Section V addresses the demonstration we propose, being divided into 2 subsections: (i) a summary of the demonstration and (ii) its outline. Finally, some brief conclusions are presented in Section VII.

II. 5GASP's Onboarding, Deployment and Validation Services

A. NetApp Onboarding Model

In 5GASP, a NetApp is described as a set of services that fulfill certain use cases, in the context of the 5G System. Hence, an onboarding procedure was introduced with a noticeable focus on maintaining the uniformity of its consequent entities, which are based on widely adopted standards, simultaneously fitting the project's goals and assuring interoperability with other similar implementations on the same scope. Hereby, the elected supported model to facilitate the described process consists of three discrete entities, i.e. NetApp artifacts, hosting network slice, and testing artifacts (Fig. 1).

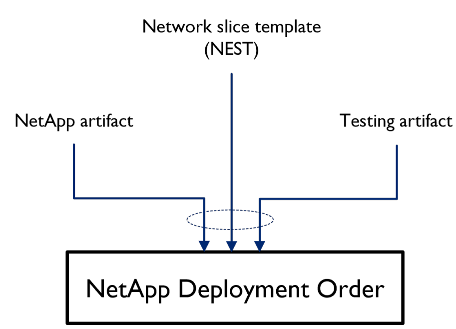


Fig. 1. 5GASP onboarding and deployment model [5]

Shortly, the NetApp artifacts adopt the ETSI NFV model and are mapped to TM Forum's (TMF) Service Specifications [6], thus enabling interaction with external Business/Operations Support Systems (BSS/OSS) through the supported Open API families. As the NFV architecture is guided towards a cloud-native approach, 5GASP aims to support seamless transformation for already containerized applications to VNFs via Kubernetes Helm Charts, leveraging standards-introduced deployment schemes [7].

The network slice templates describe the offered 5G network capabilities that are accommodated by the 5GASP corresponding testbeds. The applicable slice template can either be selected by the developer or it may be appointed automatically, based on the requested Key Performance Indicators (KPIs). The supported network slice templates are based on GSMA's GST [8].

Lastly, for the testing phase, 5GASP designed and implemented its own test descriptors in the form of YAML files. The overall testing pipeline is incorporated in the said descriptors, e.g. type of test executed, required deployment parameters, etc. Currently, along with the pre-defined tests that are publicly available, a developer is also able to outline its own test leveraging the established descriptor, while providing the respective test scripts.

Each of the above model entities is expressed with a standardized set (TMF OpenAPIs) of characteristics. To support this implementation, 5GASP provides a user-friendly portal as part of NODS, which integrates a Service and Network Orchestrator that coordinates the deployments on the underlying facilities and also an interaction point with 5GASP CI/CD Manager, charged with the execution of the CI/CD pipeline. The 5GASP NODS is based on the open-source project Openslice[1].

A more extensive elaboration on the above aspects can be found in [2] and [5], therefore is omitted in this work.

B. NetApp Deployment and Orchestration

The deployment process begins as soon as a previously designed service bundle is appointed for fulfillment. This request is captured by a TMF's Service Order [9] request. The overall process takes place in two layers. At first, the Service Orchestrator receives the unprocessed Service Order and outlines the delivery scheme, followed by the deployment actions delegated in the corresponding testbed's domain. Specifically, the orchestration process may occur in a single testbed or employ Network Orchestrator to lay an interconnecting network mesh between multiple testbeds.

The selection of the testbed(s) where the NetApp is deployed is currently allocated to the NetApp developer, who is expected to pick from a reservoir of available network slices per testbed, later to be automated. As previously stated, the provided network slices are based on GST properties, forming Network Slice Templates (NESTs). Should a specific NEST be selected, the Service Orchestrator transforms its properties to network requirements that are expected to be met at the appointed testbed(s). That said, each testbed is charged to design and maintain the actual NSs to accommodate these requirements.

At the moment, regarding the layer of network services' actuation, every testbed has implemented an ETSI-NFV compliant Management and Orchestration (MANO) stack, namely OSM, which is then made available to NODS. Consecutively, an ETSI SOL005 compliant client is employed within NODS that converts the orchestration scheme to NS lifecycle management actions.

C. NetApp Validation

To validate a NetApp, 5GASP introduced a Validation Service, which is responsible for testing and validating NetApps through the execution of (i) global tests that are hosted in the 5GASP ecosystem and (ii) custom tests onboarded by the developers.

The Validation Service is triggered by the NODS, by submitting a TMF 653 Service Test payload to the CI/CD Manager - the entity responsible for coordinating the validation process - which uses this information to generate a validation pipeline dynamically. To gather test artifacts, such as the Testing Descriptor file, the CI/CD Manager will request them from NODS after the triggering TMF payload, which made available URLs from where these resources could be obtained.

The initial onboarded Testing Descriptor file can not immediately originate a validation process, since it lacks crucial information that can only be obtained after the NetApp's deployment. This information is made available to the CI/CD Manager via the Service Test Payload and will then have to be injected into the Testing Descriptors, in a process we describe as Testing Descriptors Rendering. Fig. 2 further elaborates this process.

Once the Testing Descriptors are rendered, the validation process may begin. Throughout this process, several entities will intervene, although it is possible to highlight the most important ones: (i) the CI/CD Manager, responsible for creating the validation pipelines and coordinating the validation process, (ii) CI/CD Agents, which are deployed in the testbeds and responsible for executing the validation pipelines created by the Manager, (iii) the Local Test Repositories (LTRs), which are deployed alongside the CI/CD Agents and store the tests that live inside the 5GASP ecosystem, and (iv) the Test Results Visualization Dashboard (TRVD), which provides a graphical user interface (GUI) to showcase the outcomes and results of the NetApps' validation.

Based on the rendered Testing Descriptor, a validation process is configured and submitted to the CI/CD Agent that will test the NetApp. The Agent will then gather the required pre-defined tests from the LTR. If developer-defined tests are required, these will be obtained from the CI/CD Manager, which stores these tests and makes them available through a specific API endpoint. The CI/CD Agent will then perform all these tests, and create a Test Report comprising their outcomes and results, to be submitted to the CI/CD Manager.

[1] openslice.io

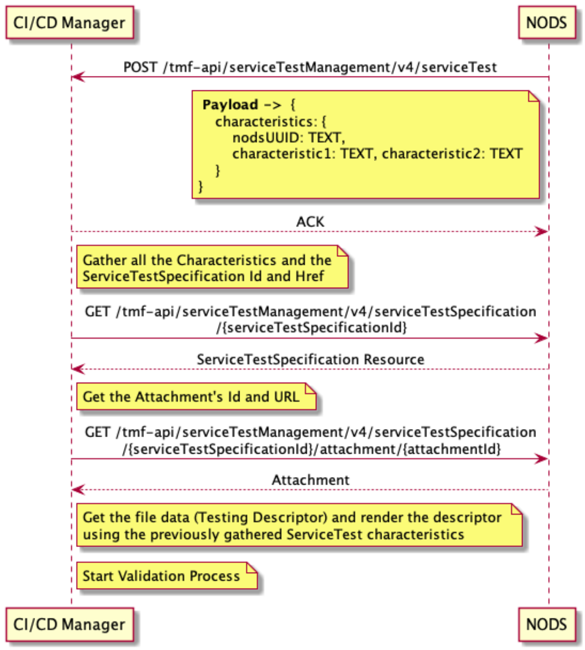


Fig. 2. Interactions Between the CI/CD Manager and the NODS

When receiving a Test Report, the CI/CD Manager will create a validation process visualization in the TRVD to showcase the outputs of the validation process. Then, it will patch the initial TM Forum 653 Service Test instance with a URL that the NetApp developers may use to access the outcomes of the validation of their NetApps.

III. Novelty of the Demonstrated Approach

Our demonstration tackles the onboarding, deployment, and validation of NetApps, which is a highly explored topic in ICT-41 projects. 5GASP introduces a straightforward Onboarding, Deployment and Validation Service that uses various tests to validate NetApps, providing a baseline for a NetApp certification service. To validate a NetApp, it uses YAML Testing Descriptors, which syntax enables the NetApp developers to detail the tests that shall be performed clearly. Besides, 5GASP seamlessly provides several out-of-the-box tests that can be executed with almost no effort. With our approach, we simplify the process of testing and validating NetApps. We also strive to offer a Certification Service based on this validation, which would confer a layer of trust to a NetApp.

IV. Relevance of this Demonstration to MeditCom 2022

The 5GASP project aims to offer a unified semi-public testbed where NetApp developers, mainly Small and Medium-sized Enterprises (SMEs), can experiment with their NetApps. We provide the possibility for the developers to onboard their NetApps to a 5G testbed and validate them according to several 5G KPIs. With this demonstration, we seek to showcase how all this can be done, hopefully providing one more tool that SMEs may use to speed up their NetApps' time-to-market.

V. Demonstration

A. Demonstration Summary

The demonstration addressed in this paper presents several critical workflows of the 5GASP NetApp onboarding, deployment, and validation services. We will showcase (i) how can the developers onboard a NetApp bundle to the NODS, (ii) the internal 5GASP Validation Service's operations, and (iii) how may the NetApp developers gather the results of a validation process. Given this, we can divide our demo into the following phases: (i) NetApp bundle onboarding, (ii) network slice deployment, (iii) NetApp deployment, (iv) NetApp validation process, and (iv) results gathering.

B. Demonstration Outline

We will use a prototype NetApp to showcase all the previously described processes. This prototype can not be considered a NetApp *per se* since it is only comprised of a NS with two VNFs, without any business logic or extensive interaction with the 5G Core Network Functions. Although, this prototype NetApp is sufficient to demonstrate the 5GASP's onboarding and validation processes.

Prior to the deployment and validation process, NetApp's NFV artifacts are expected to be onboarded to NODS and then handed over to the respective NFV Orchestrators. The artifact onboarding process employs a static validation of the provided descriptors and, once the validity is asserted, renders them available to design deployment bundles based on them. In the showcased scenario, we have designed a NetApp bundle that comprises (i) an eMBB 5G slice at Patras' testbed, (ii) the previously described prototype NetApp at the same testbed, and (iii) the execution of a prototype test. The dedicated GUI that enables this bundle stitching is presented in Fig. 3

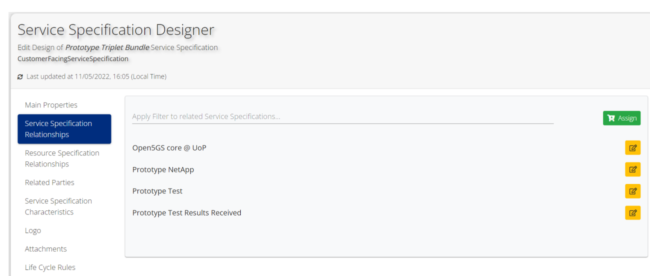


Fig. 3. NetApp bundle design GUI

After the onboarding of the NetApp bundle, we will showcase the deployment of the network slice, where the NetApp will reside. The demonstrated 5G Slice consists of a network service that provides the overall 5G Core functionality, except User Plane Function (UPF), and another network service in charge of deploying multiple discrete instances of UPF, based on specific placement preferences. The UPF instances self-register at the Session Management Function of the deployed

5G Core, simultaneously exposing their datapath, which can be leveraged by a NetApp to stream traffic.

The prototype NetApp will be in line for deployment, only after the availability of the 5G Core and the consecutive exposure of UPF's datapath. Following the same pattern, prototype testing can only be initiated after the NetApp is fully functioning. In that sense, testing requires several deployment information, such as VNF IPs, which are expected to be provided prior to test execution. This can be accomplished by the introduction of a rule engine that is defined in NODS and injects specific business logic throughout the network service lifecycle, e.g. pre-provision, after activation, supervision, etc. In this specific scenario, we will showcase a rule that extracts the deployment IPs from the VNF information acquired by OSM, after a service is activated. The GUI is based on Blockly[2], and provides a graphical representation of executing ad-hoc code, as seen in Fig.4. A similar rule is applied for the overall pipeline workflow that blocks the execution of testing until the expected deployment IPs are provided.

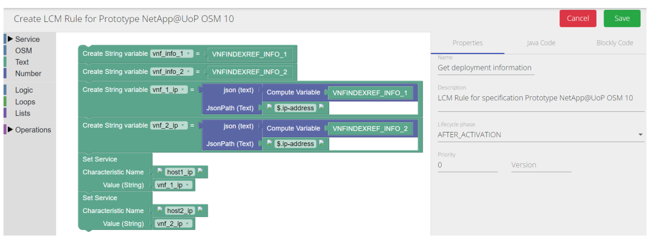


Fig. 4. Rule design GUI

To test the prototype NetApp, we have also designed a prototype test. Like all the 5GASP ecosystem tests, this test was developed using the Robot Framework[3], and evaluates if the bandwidth between two VNFs is greater than a certain threshold, defined in the onboarded Testing Descriptor.

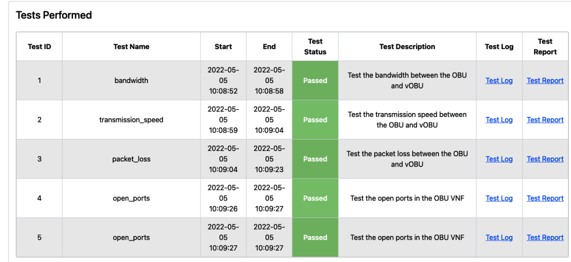
During the 4th phase of our demonstration, we will showcase the internal processes of the 5GASP Validation Service, presenting the dashboard of the CI/CD Agent executing the prototype test.

Finally, we will showcase the TRVD's GUI, demonstrating how the NetApp developers will be able to gather the results of the validation of their NetApps as well as the depth of these results. Since the TRVD is public, the audience will be able to access its GUI and verify the collected results. Fig. 5 showcases a portion of the TRVD's web interface, illustrating an example of the results collected from a validation process.

Given we are still working on implementing the developer-defined tests workflow, this demonstration will not showcase their onboarding and execution.

It is also worth mentioning that this demonstration can be presented in-person or online, via a video conference platform, without changing the demonstration outline.

[2]developers.google.com/blockly
[3]robotframework.org



Tests Performed

Test ID	Test Name	Start	End	Test Status	Test Description	Test Log	Test Report
1	bandwidth	2022-05-05 10:08:52	2022-05-05 10:08:58	Passed	Test the bandwidth between the OBU and vOBU	Test Log	Test Report
2	transmission_speed	2022-05-05 10:08:59	2022-05-05 10:09:04	Passed	Test the transmission speed between the OBU and vOBU	Test Log	Test Report
3	packet_loss	2022-05-05 10:09:04	2022-05-05 10:09:23	Passed	Test the packet loss between the OBU and vOBU	Test Log	Test Report
4	open_ports	2022-05-05 10:09:26	2022-05-05 10:09:27	Passed	Test the open ports in the OBU VNF	Test Log	Test Report
5	open_ports	2022-05-05 10:09:27	2022-05-05 10:09:27	Passed	Test the open ports in the OBU VNF	Test Log	Test Report

Fig. 5. Portion of the TRVD's GUI [10]

VI. Materials Needed For This Demonstration

To showcase this demonstration in MeditCom 2022, we will require the following materials: (i) a table, (ii) a screen so we can present our demo, and (iii) internet access.

VII. Conclusions

This demonstration tackles the onboarding, deployment, and validation of NetApps in the 5GASP ecosystem, showcasing the several phases that the NetApps will undergo before being considered valid. Although 5GASP's services and platforms are still in an embryonic implementation, we believe we have achieved a state where it is possible to showcase our solution's added value in the NetApp validation realm. In the future, we will further elaborate on the creation, onboarding, and execution of developer-defined tests and also implement mechanisms to collect infrastructure metrics, thus increasing the scope of our validation service.

VIII. Acknowledgment

This work was supported by the H2020 European Project 5GASP (grant agreement No. 101016448)

References

[1] M. Chiosi *et al.*, "White paper: Network Functions Virtualisation - An Introduction, Benefits, Enablers, Challenges & Call for Action ," Tech. Rep. 1, October 2012.

[2] K. Trantzas *et al.*, "An automated CI/CD process for testing and deployment of network applications over 5g infrastructure," in *2021 IEEE MeditCom*, Sept. 2021. DOI: 10.1109/meditcom49071.2021.9647628.

[3] P. Twamley, M. Müller, P.-B. Bök, G. K. Xilouris, C. Sakkas, M. A. Kourtis, M. Peuster, S. Schneider, P. Stavrianos, and D. Kyriazis, "5GTANGO: An Approach for Testing NFV Deployments," in *2018 European Conference on Networks and Communications (EuCNC)*, pp. 1–218, 2018.

[4] VITAL-5G, "D1.4 5Testing and validation methodology," December 2021.

[5] 5GASP, "D3.1 5GASP experimentation services, middleware and multi-domain facilities continuous integration," September 2021.

[6] TM Forum, "TMF633 - Service Catalog Management API REST Specification," January 2021.

[7] ETSI NFV ISG, "GR NFV-IFA 029, Architecture; Report on the Enhancements of the NFV architecture towards "Cloud-native" and "PaaS"," November 2019.

[8] GSMA, "Generic Network Slice Template v6.0," November 2021.

[9] TM Forum, "TMF641 - Service Ordering Management API REST Specification," March 2021.

[10] R. Direito *et al.*, "Towards a fully automated system for testing and validating netapps," in *2022 IEEE NetSoft*, June 2022.