A 4G/5G Packet Core as VNF with Open Source MANO and OpenAirInterface

Thomas Dreibholz
SimulaMet – Simula Metropolitan Centre for Digital Engineering
Pilestredet 52, N-0167 Oslo, Norway
dreibh@simula.no

***Abstract*—5G, the fifth generation of mobile broadband networks, is going to make a large range of new applications possible. However, further research is necessary, and the basic step, i.e. setting up a 4G/5G testbed infrastructure, is a complicated and error-prone task. In this abstract and poster, we introduce our open source SIMULAMET EPC Virtual Network Function (VNF), as an easy way to set up a 4G/5G testbed based on OPEN SOURCE MANO and OPENAIRINTERFACE. We would like to showcase how a researcher can use our VNF as part of his own research testbed setup. Therefore, the focus is particularly on the user interface details and features of the SIMULAMET EPC VNF.**

I. Introduction

The usage of *Mobile BroadBand* (MBB) communication is steadily increasing. With the beginning deployment of 5G networks, many new applications, like high-speed media streaming, low-latency vehicular communication, control of unmanned aerial vehicles, energy-efficient Internet of Things communication and reliability-critical emergency networks, will be possible. However, there is still a significant amount of research necessary to let all these applications become reality. The initial step for contributing to such research is, obviously, to set up a basic 4G/5G testbed infrastructure.

Creating an open source software base to realise a full 4G/5G setup is the goal of the OPENAIRINTERFACE project. Together with reasonably low-cost *Software-Defined Radio* (SDR) hardware, the entry into 4G/5G research even becomes feasible for small businesses and universities. However, setting up OPENAIRINTERFACE [1] is neither easy nor straightforward, due to its many branches, necessary bugfixes, and fast-paced ongoing development. Therefore, an automated testbed setup would be highly helpful for researchers.

As part of the 5G-VINNI project, we are developing the SIMULAMET EPC [2], [3] VNF, which is an open source *Virtual Network Function* (VNF) for an *Evolved Packet Core* (EPC) based on OPENAIRINTERFACE. By using OPEN SOURCE MANO (OSM) together with a cloud infrastructure like OPENSTACK, it can be deployed easily in different versions and with varying configurations and parameters. The goal of the SIMULAMET EPC VNF is that a researcher can simply use it as a building block, and attach further functionalities as VNFs into a *Network Service* (NS). For example, a researcher on *Mobile Edge Computing* (MEC) can simply combine MEC functionalities with our EPC VNF, and focus on realising his MEC features instead of dealing with the complex details of setting up and configuring the EPC.

While the focus of [2] is an early proof-of-concept of the SIMULAMET EPC VNF, and [3] examined possibilities of custom metrics handling, we would like to emphasise the user interface details and features of the SIMULAMET EPC VNF in this abstract and on our poster. Particularly, we would like to showcase how a researcher can use our VNF as part of his own testbed setup for research on 4G/5G systems.

II. Open Source MANO

OPEN SOURCE MANO[1] (OSM) is an open source *Management and Orchestration* (MANO) framework [4] for Network Function Virtualisation (NFV). NFV with OSM can be split into three parts [4, Chapter 1]:

1) *NFV Infrastructure* (NFVI) denotes hosting virtual machines and containers, as well as connecting them by *Virtual Links* (VLs). OSM is independent of an underlying hosting mechanism like e.g. OPENSTACK [5], [6].
2) Collection of VNFs and their interconnection and composition into *Network Services* (NS). Furthermore, NSs can be composed and shared to form network slices.
3) The MANO controls the life-cycle of VNFs, NSs and network slices, including configuration and monitoring.

Realising and deploying a VNF as part of a NS is introduced in detail in [2], [7]. In short, A VNF is created in form of a *VNF Descriptor* (VNFD). The VNFD contains a definition of its *Virtual Deployment Units* (VDU), each corresponding to its own *Virtual Machine* (VM). *Connection Points* (CP) define interfaces. CPs of a VDU are network interfaces in a VM instance. CPs can be connected by *Virtual Links* (VL). VDUs of the VNF are connected by using internal CPs. A VL is referenced by its VL Descriptor (VLD), which can be a name like e.g. "S11". CPs of VDUs can also be connected to external CPs of the VNF. Finally, A *NS Descriptor* (NSD) connects VNFs with VLs. Furthermore, it has the possibility to also define external NS CPs, e.g. for attaching them to physical networks in the underlying NFVI.

III. OpenAirInterface

The goal of the OPENAIRINTERFACE[2] project is to develop an open source software solution to realise the EPC, access network and user equipment of cellular networks. In particular, OPENAIRINTERFACE realises the following five components [1] (see also Figure 1; to be explained in Section IV):

a) Home Subscriber Server (HSS): maintaining the information about users and their subscriptions. Its functionalities include mobility management, session establishment, user authentication and access authorisation.

[1]OPEN SOURCE MANO: https://osm.etsi.org.
[2]OPENAIRINTERFACE: https://www.openairinterface.org.

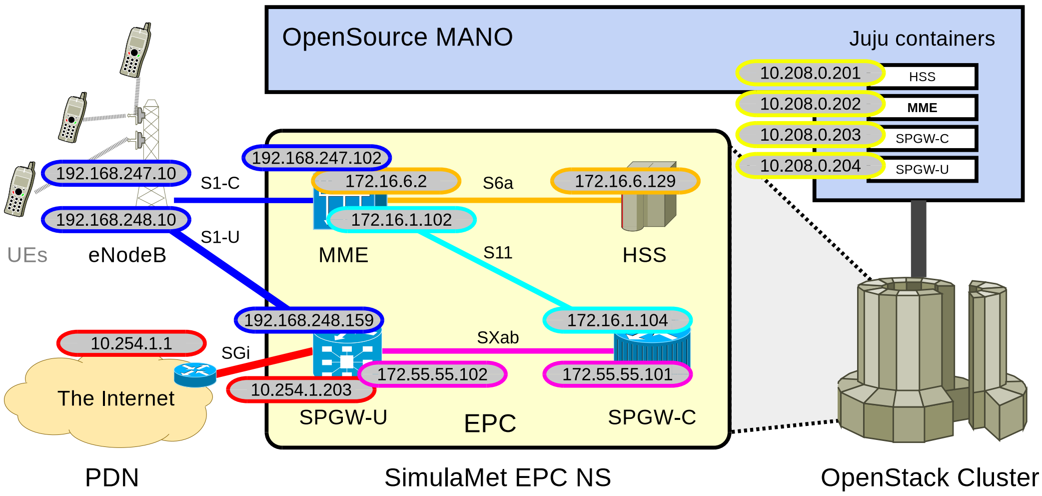

Fig. 1. A SIMULAMET EPC Testbed Example with EPC, OPENSTACK as NVFI, as well as eNodeBs and UEs (based on [2])

b) Mobility Management Entity (MME): handling the procedures of attaching and detaching as well as service requests of *User Equipment* (UE) and eNodeBs.

c) Control Plane of the Packet Data Network Gateway (SPGW-C): the control part of a combined Serving Gateway (SGW) and Packet Data Network Gateway (PGW). That is, OPENAIRINTERFACE combines SGW and PGW, but uses *Control and User Plane Separation* (CUPS).

d) User Plane of the Packet Data Network Gateway (SPGW-U): handling the forwarding of user traffic between the *Public Data Network* (PDN) at the SGi interface (i.e. usually the public Internet) and the eNodeBs.

e) Evolved Node B (eNodeB): realising a base station using *Software-Defined Radio* (SDR) hardware connected to a computer via USB 3.x port. *User Equpments* (UE), e.g. smartphones or computers, connect via MBB radio.

IV. THE SIMULAMET EPC VNF

Figure 1 shows a typical MBB testbed setup [2], consisting of the EPC (i.e. HSS, MME, SPGW-C and SPGW-U) using OPENAIRINTERFACE, NFVI in form of OPENSTACK, eNodeBs and UEs, and the Internet. OSM is the orchestration framework, using JUJU Charms for configuration (i.e. there is one Charm container assigned for the management of each of the VDU VMs). In the following, we describe how to set up this example by using OSM for managing an NS with our SIMULAMET EPC VNF.

Using the VNFD, a VNF can be instantiated as part of an NS described by an NSD. In the simplest case, as shown in Figure 2 in the OSM GUI, it just consists of the SIMULAMET EPC VNF itself with VLs to physical interfaces. For instantiation, all parameters of the EPC (see also [2]) have to be provided, giving the details of the mobile network to be created, i.e. realm, *Mobile Country Code* (MCC), *Mobile Network Code* (MNC), *Operator Code* (OP), *Subscriber Key* (K), as well as the first user's *International Mobile Subscriber Identity* (IMSI) and *Mobile Station International Subscriber Directory Number* (MSISDN). IMSI and MSISDN are incremented for further users. Of course, simcards for the new mobile network have to match these settings. Further parameters include the addressing of the components. In addition, the parameters specify the Git repository and commit for each of the four components (i.e. HSS, MME, SPGW-C and SPGW-U). During instantiation, their software is build from the given sources, i.e. a researcher can exactly specify which version of the software for each component is needed. Particularly, this enables the easy usage of customised software versions, e.g. to include own fixes or experimental extensions like MEC.

Once the VNFD instance is runninng, its components (i.e. HSS, MME, SPGW-C and SPGW-U) provide metrics to the PROMETHEUS monitoring service of OSM, i.e. statistics about the performance (e.g. CPU load, memory utilisation, network usage, etc.) can be queried and visualised using the GUI interface of PROMETHEUS. Figure 3 provides an example for the incoming bytes on the S11 interface in the last 24 h, at MME (green colour) and SPGW-C (red colour).

While many useful metrics can be collected by using PROMETHEUS, it is also possible to directly use SSH access to control the components, i.e. it is possible to establish SSH sessions over the management network and issue arbitrary commands. The example in Figure 4 displays an example for accessing the MME log via SSH to the MME's management address 10.208.0.201. In this case, it shows the log of a freshly set up MME, without any connected eNodeB or UE.

All SIMULAMET EPC VNFD sources, together with the example NSD from Figure 2, build tool-chain and test scripts, operating system image build script, as well as documentation is available as open source from https://github.com/simula/5gvinni-oai-ns (public Git repository).

V. CONCLUSIONS

Our open source SIMULAMET EPC VNF, based on OPENAIRINTERFACE and to be used with OPEN SOURCE MANO (OSM), provides an easy way to instantiate MBB testbeds for research on 4G/5G networks. In this abstract and on our poster, we introduced the SIMULAMET EPC

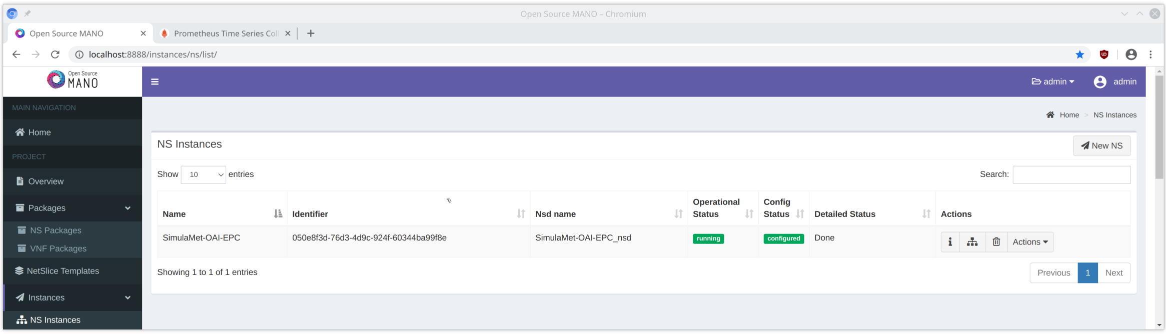

Fig. 2. An Instantiated SIMULAMET EPC VNF in the OSM GUI

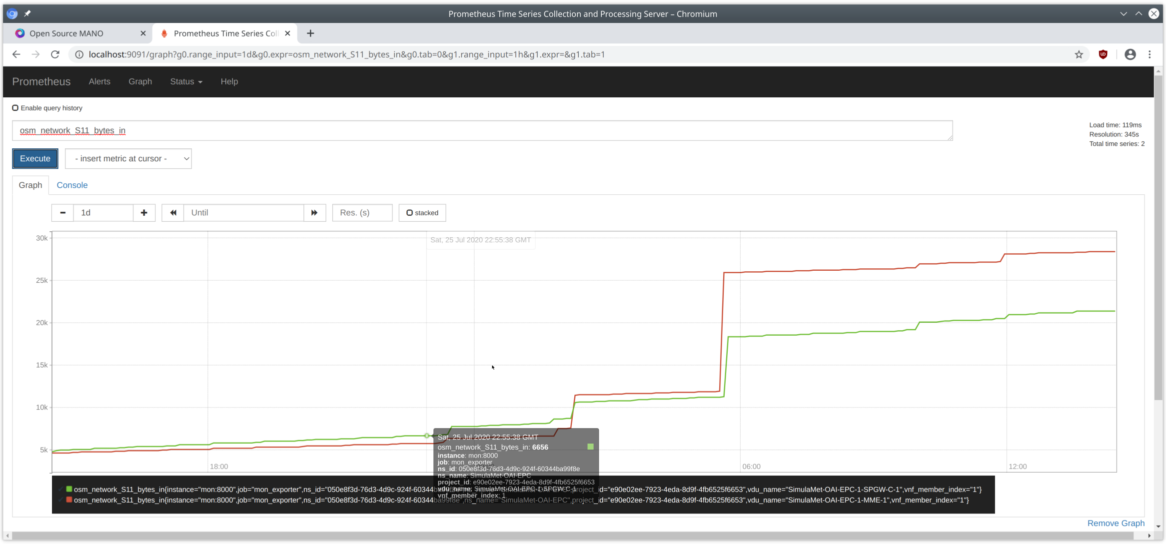

Fig. 3. PROMETHEUS View with the Incoming Bytes of the S11 interface at MME and SPGW-C of an SIMULAMET EPC VNF Instance

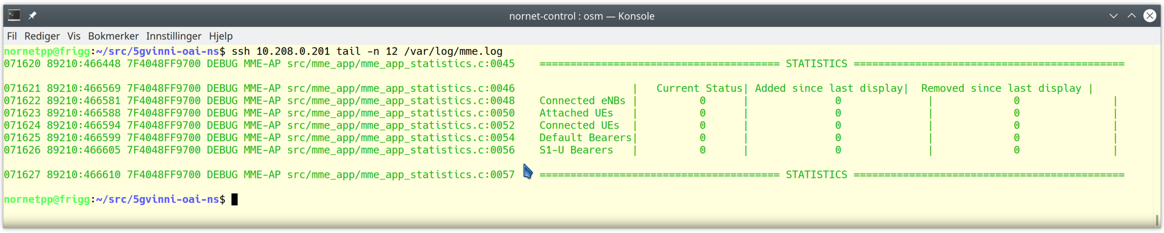

Fig. 4. Getting Shell access to the MME of an SIMULAMET EPC VNF Instance

VNF and its features to interested researchers. Our ongoing work on the SIMULAMET EPC VNF currently focuses on performance evaluation [2], metric collection features [3] and MEC applications [8]. Particularly, we would also like to integrate support for the setup of network slicing.

REFERENCES

[1] OpenAirInterface, *OpenAirInterface Software Installation Support*, Dec. 2019.

[2] T. Dreibholz, "Flexible 4G/5G Testbed Setup for Mobile Edge Computing using OpenAirInterface and Open Source MANO," in *Proceedings of the 2nd International Workshop on Recent Advances for Multi-Clouds and Mobile Edge Computing (M2EC) in conjunction with the 34th International Conference on Advanced Information Networking and Applications (AINA)*, Caserta, Campania/Italy, Apr. 2020, pp. 1143–1153.

[3] M. Xie, J. Pujol-Roig, F. Michelinakis, T. Dreibholz, C. Guerrero, A. Sanchez, W. Y. Poe, Y. Wang, and A. M. Elmokashfi, "AI-Driven Closed-Loop Service Assurance with Service Exposures," in *Proceedings of the European Conference on Networks and Communications (EuCNC)*, Dubrovnik, Dubrovnik-Neretva/Croatia, Jun. 2020.

[4] A. Reid, A. González, A. E. Armengol, G. G. de Blas, M. Xie, P. Grønsund, P. Willis, P. Eardley, and F.-J. R. Salguero, "OSM Scope, Functionality, Operation and Integration Guidelines," ETSI, White Paper, Jun. 2019.

[5] OpenStack, *OpenStack Architecture Design Guide*, Dec. 2019.

[6] ——, *OpenStack Operations Guide*, Dec. 2019.

[7] G. Lavado, "OSM VNF Onboarding Guidelines," ETSI, White Paper, Jun. 2019.

[8] Y. Luo, X. Zhou, T. Dreibholz, and H. Kuang, "A Real-Time Video Streaming System over IPv6+MPTCP Technology," in *Proceedings of the 1st International Workshop on Recent Advances for Multi-Clouds and Mobile Edge Computing (M2EC) in conjunction with the 33rd International Conference on Advanced Information Networking and Applications (AINA)*, Matsue, Shimane/Japan, Mar. 2019, pp. 1007–1019.