Deliverable D2.1

5G-VINNI Solution facility sites High Level Design (HLD) - v1
Munich Facility Site annex
(Release 4)

Editor:	Huawei
Deliverable nature:	Report (R)
Dissemination level: (Confidentiality)	Public (PU)
Contractual delivery date:	N/A
Actual delivery date:	30 June 2021
Suggested readers:	Example: Executives in entertainment companies and banks, investors
Version:	
Total number of pages:	26
Keywords:	5G, Solution Design, High Level Design

Abstract

Description of network services for each facility site. Includes configuration of Infrastructure, RAN, Core and MANO components and of the interconnections among them. Used as a reference for the orchestration and testing activities. Description of the cross-facility site services and interconnection requirements and configurations.

[End of abstract]

Disclaimer

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 815279.

Impressum

Full project title	5G Verticals Innovation Infrastructure
Project acronym	5G-VINNI
Number and title of work-package	WP2 5G-VINNI End-to-End Facility Implementation and Infrastructure Integration
Number and title of task(s)	T2.1 Mapping of the Reference Architecture to Facility sites
Document title	5G-VINNI Solution facility sites High Level Design (HLD) – v1 – Munich Facility Site annex – (Release 4)
Editor: Name, company	Huawei DE
Work-package leader: Name, company	Terje Rød, Ericsson

Copyright notice

List of authors

Company	Author	Contribution
HWDU	Osama Abboud	Established the doc, MEC details of the site, coordination
HWDU	Xun Xiao	Core network details, platform
HWDU	Mohamed Gharba	RAN and radio details of the site
HWDU	Artur Hecker	MANO, NFV, SDN core network details, coordination

Table of Contents

List of figures

List of tables

Abbreviations

5G	Fifth Generation (mobile/cellular networks)
5G-VINNI	5G Verticals INNovation Infrastructure
HLD	High Level Design
LLD	Low Level Design
MANO	Management and Orchestration
NFV	Network Functions Virtualization
VNF	Virtual Network Function
NFVI	Network Functions Virtualization Infrastructure
VIM	Virtualized Infrastructure Manager
RAN	Radio Access Network
eNB	evolved NodeB
gNB	gNodeB (g=Next Generation)
gUE	g User Equipment (g=Next Generation)
SGSN	Serving GPRS Support Node (GPRS= General Packet Radio Service)
MME	Mobility Management Entity
SGW/PGW	Serving Gateway / PDN Gateway (PDN= Packet Data Network)
PCRF	Policy and Charging Rules Function
HSS	Home Subscriber Server
EMS	Element Management System
MEC	Mobile Edge Computing, Multi-access Edge Computing

1 Introduction

1.1 Scope

This document is description of network services for 5G-VINNI specific facility site. Includes configuration of Infrastructure, RAN, Core and MANO components and of the interconnections among them. Used as a reference for the orchestration and testing activities. Description of the cross-facility site services and interconnection requirements and configurations.

1.2 5G VINNI Facility

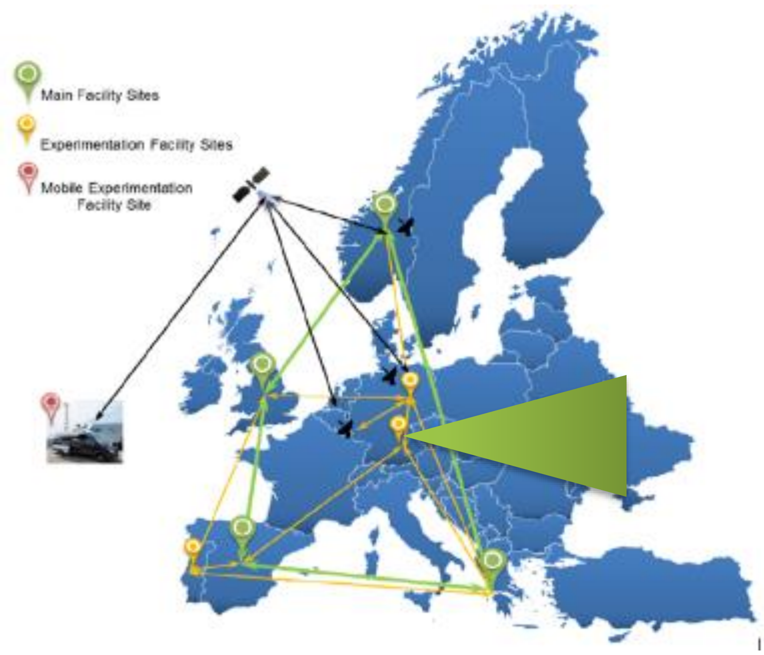

Figure 1 5G-VINNI Facility

1.3 Facility-site: Munich, Germany

Table 1 Facility-site details

Facility-site	
Type	Experimental Facility Site
Locations	Munich
Provider	HWDU
5G-VINNI Release	*Release 2*

Note: "Facility-site" has the same meaning as "Site". Within one site can be one or more "Locations".

1.3.1 Location details

Table 2 Location address

Address	
City, Street	*Riesstr. 25, 80992 Munich*
Building, Room number	*Building C*

1.3.2 Vendors information

Table 3 Location vendors details

Domain	**Vendor**
5G RAN	*HWDU*
5G Core	*HWDU*
NFVI	*HWDU*

2 Facility-site deliverables

2.1 Facility-site use cases

The use case in the focus of investigation of the 5G-VINNI experimental facility site in Munich is:

- eHealth use-case: This use case is based on equiping an ambulance car with 5G capabilities. The ambulance car is considered as the first room of a hospital. When the hospital is far away the diagnosis and treatment in prior to arrival helps to increase patient safety. We provide a Virtual presence of medical expert (located in the hospital). Our target in the Test is to provide 5G based data transfer between moving point of care (ambulance= first responder) and university hospital (medical expert) facilitating bi-directional data transmission for ultrasound, audio and video source.

2.2 Facility-site capabilities

1. 5G Radio Access Network (RAN) at one site, with two sectors, each having
    a. Carrier frequency: 3.4 GHz
    b. Bandwidth: 40 MHz
    c. Transmission power: 5 W
    d. Antennas: up to 8

2. 5G Mobile Terminals (MT)
    a. 5G Core network: HW/SW platform with 5G Core Network on top

3. Hardware: in-house platform comprising of several dozen powerful servers representing a data centre.
4. Software: software networks testbed, comprising extended network emulators (e.g., cluster Mininet with Docker extensions), controllers (e.g., Floodlight), open-source and proprietary switch implementations.
5. Orchestration platform for scenarios relevant to virtualization.
6. 5G Core network supporting functional split – SDN – NFV Orchestration
    a. Distributed data centres for mobile edge computing use cases

2.3 Facility-site services

RAN

- Testbed can deploy virtual networks with different topologies as needed including transport networks
- Virtual networks can be deployed across a server cluster.
- Nodes as compute containers (e.g. Docker) that run distributed agent programs to test/evaluate various proposed protocols.

Radio

- URLLC Service, eMBB

3 Mapping of global architecture from WP1

Note: the recommendations seem not yet defined by WP1 and therefore they will not be listed here. Once the recommendations are finalized, they will be listed here.

Domains: 5G Radio and Core

Table 4 WP1/WP2 architecture mapping 1

Number	WP1 recommendation	WP2 Implementation	Note
1	**RAN Functionality**	*SA* *Throughput DL: 30.8 Mbps , UL: 54 Mbps* *Latency: DL/UL: 0.5ms* *Reliability: 99.999 %* *Carrier Frequency (C-Band) [3.41, 3.43, 3.45, 3.47, 3.49, 3.51, 3.53, 3.55, 3.57, 3.59] GHz*	*1*
2	**LTE – NR Dual Connectivity**	*N.A.*	*2*
3	**Mobility**	*N.A.*	*3*
4	**Uplink/downlink Decoupling**	*N.A.* *Not all sites will support this.*	
5	**LTE-NR Aggregation**	*N.A.* *Not all Sites require LTE-NR aggregation to prove the key KPI of 5G.*	
6	**Multiple Input Multiple Output (MIMO)**	*gNB: can support 8 Antennas, UE can support 4 Antennas.*	
7	**Numerology support**	*Flexible numerology is supported by the Munich site*	

8	**Carrier Bandwidths NR**	*20 MHz: UU* *40MHz: Sidelink*	
9	**Modulation**	*AMC*	
10	**Network slice aware RAN**	*RAN Slicing*	
11	**RAN integration with NR standalone (SA) Core architecture**	*Ok*	
12	**NR QoS Framework**		
13	**5G Fixed Wireless Access (FWA)**		
14	**Dynamic spectrum sharing**		
15	**Energy Efficiency**		
16	**Satellite backhaul**		
17	**Operational efficiency**		
18	**Public safety**		
19	**cMTC and mMTC**		
20	**NR Side Link**	*Sidelink*	
21	**Network Slice aware RAN**		
22	**Decomposition of RAN**		

Domains: MANO and NFVI

Table 5 WP1/WP2 architecture mapping 2

Number	WP1 recommendation	WP2 Implementation	Note
1	MANO and NFVI	In-house MANO and NFVI system in Rel.1	
2			
..			

Domain: Service Orchestration (not applicable)

Table 6 WP1/WP2 architecture mapping 3

Number	WP1 recommendation	WP2 Implementation	Note
1	Service Orchestration	N.A.	
2			
..			

4 Facility-site end-to-end design

4.1 Transport network

Not applicable as this is an experimental facility site and the whole site is in one building

4.2 5G RAN and Core

4.2.1 RAN

The RAN geographical structure are depicted in the map of Figure 2. The RAN site is located in the Riesstrasse 25 in Munich Germany.

Figure 2 5G Munich Facility site

The setup of the RAN site and the interconnections are depicted in Figure 3.

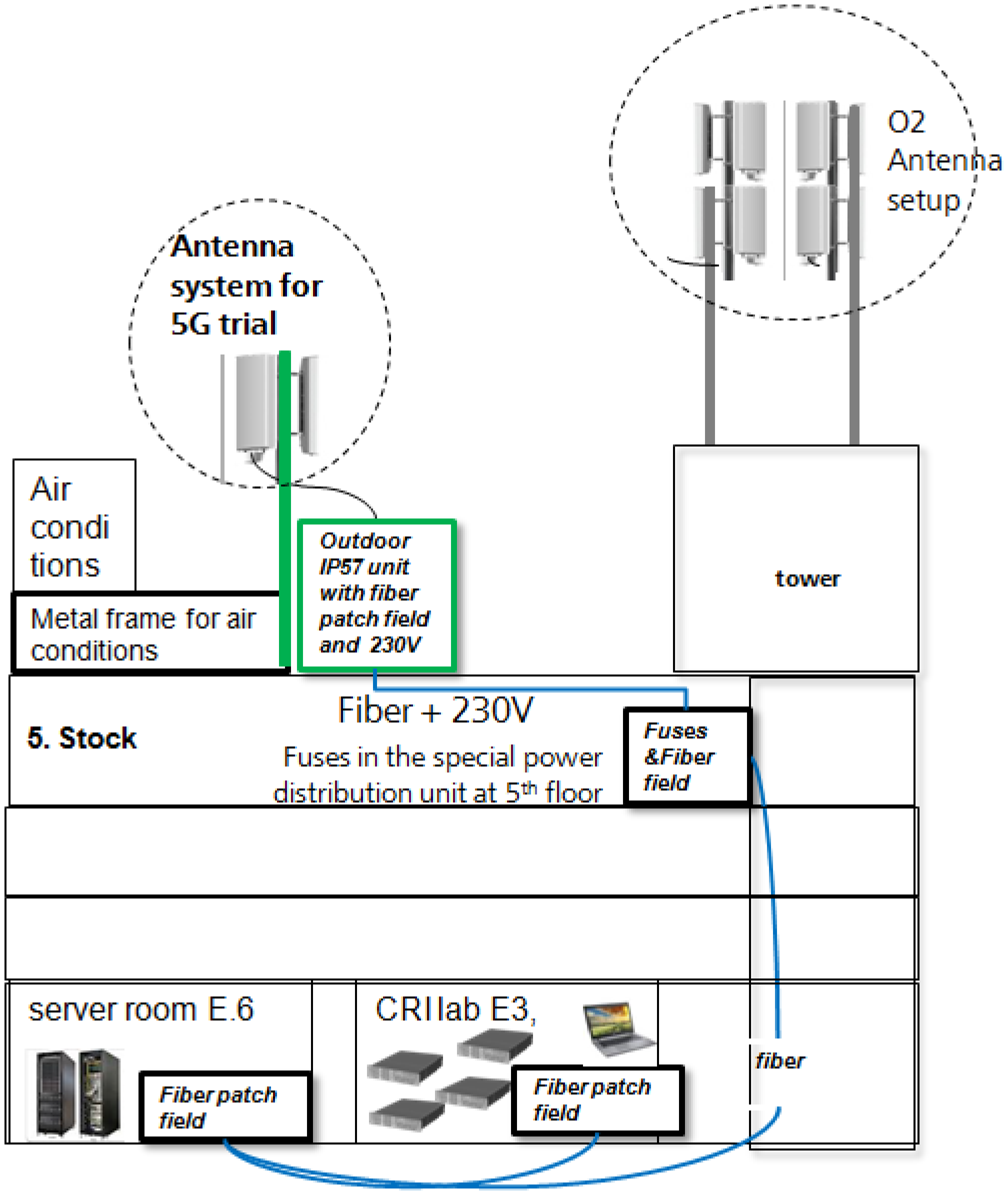

Figure 3 5G Munich setup/configuration

4.2.2 Core Network

Physically, the core network platform consists of a dedicated server farm composed of COTS servers in racks, patch panels and DC switches.

- **CPU and Memory**: Every COTS server has 48 *Intel(R) Xeon(R) E5-2697 v2 @ 2.70GHz* CPUs and each CPU has 12 cores. Every server has 80GB RAM.
- **Ethernet:** Every server has 8 Ethernet ports.
- **Networking**: All servers are interconnected with a maximum bandwidth of 10Gb/s.

4.2.3 RAN/Core Integration

The RAN and Core network have been integrated to allow for end to end testing. The RAN acts as a transport network of frames sent from the UE toward the NFs running in the core . The Base band running at the both ends will use the latest technique in order to achieve the KPIs. the UE is currently fitted into a car in order to test urban V2X scenarios. The Core network runs the applications for the V2X scenario and will be able to execute the test cases.

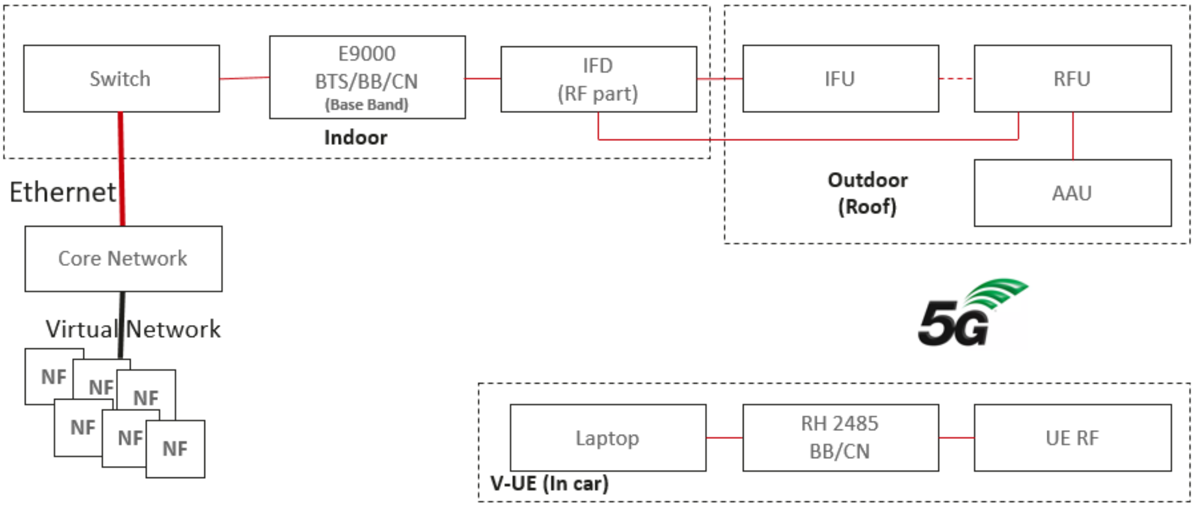

Figure 4 RAN/Core Integration

4.3 MANO and NFVI

4.3.1 NFVI

Docker[1] container and Mininet[2] are utilized together to realize our NFVI multi-host platform on top of our hardware installation. Docker containers provide a virtualized environment, where different NFs can share the same hardware resource from the server machine isolated from each other.

Mininet provides software-defined networking capability to customize the interconnectivity of the virtualized NFs running in Docker containers through Open Virtual Switch (OVS)[3]. Mininet also provides capability to connect to one or multiple software-defined network controllers. Network control application logic can operate the network via the interfaces provided by the controller.

At HWDU experimental facility site, we use a multiple-host capable, mobility-supporting extended Mininet testbed. In this testbed, nodes are Docker containers with an OVS instance inside (as opposed to OVS, as in standard Mininet). Besides, such nodes can be deployed across different physical hosts; our system guarantees that regardless of the virtual node to physical machine assignment, the desired virtual topology constraints are always respected. Finally, nodes can be reattached within the network, therefore emulating mobility events.

4.3.2 MANO

At HWDU experimental facility site, we do not have a formal MANO system in the sense of ETSI NFV definition.

However, we have full control over both the physical platform and the deployed virtual network, including runtime control. We use a special Network Control Graphical User Interface (GUI) developed in house for these purposes.

[1] https://www.docker.com/

[2] http://mininet.org/

[3] https://www.openvswitch.org/

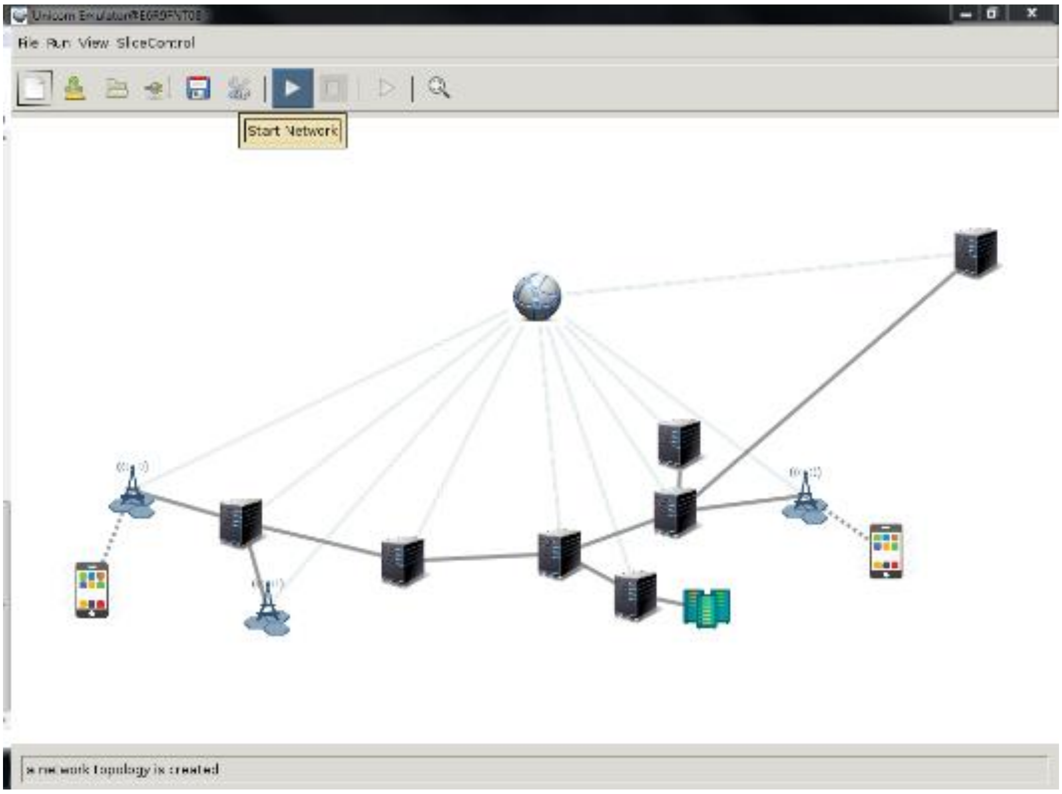

Figure 5 A GUI developed by HWDU Munich to operate NFVI

This GUI allows:

- Virtual network design (topology, number and types of nodes, addressing, assignment to physical hosts)
- Virtual network deployment (start and stop)
- Virtual network runtime control, including topology control (link up/down, node reattachment, node up/down) and running service control (like remote shell capability into the virtual nodes and the changes to the virtual link).

5 Facility-site solution design (HLD)

5.1 Security

Not applicable.

5.2 Transport Network

The transport network will be based on simple Ethernet capability. Reason being that the base station and the core network are found within the same building.

5.3 NFVI

5.3.1 Hypervisor

We use Ubuntu 12 LTS or 14 LTS on our platform servers. The hypervisor therefore is the Linux kernel, and, in principle, any available kernel hypervisor could be used (kvm/qemu, xen, lxc). We mostly use container-based virtualization and Docker containers, which rely on ns and cg capabilities of the Linux kernel, i.e. namespaces, control groups, chroot, etc.

5.3.2 Computing

Linux/Docker is used as the computing environment, where NF application are encapsulated into a Docker container corresponding to its resource requirement.

5.3.3 Network

Mininet is used to provide programmable networking capability, where VNF applications running in individual Docker containers can be interconnected. Networking will be operated by control applications running over network controllers.

5.3.4 Storage

Storage resource is allocated with the configuration of the Docker container that hosts the NF application. Such a storage configuration can be dynamically changed if more storage capability is needed.

5.3.5 EMS

Realized through the mentioned graphical network control GUI (see section 4.3.2).

5.4 MANO

5.4.1 NFV Orchestrator

The network control GUI presented above serves as a simple yet efficient orchestrator used for deployment and removal of both the whole virtual network and of all individual elements thereof (nodes or links).

Previously at HWDU, we have researched and implemented highly efficient placement algorithms (VNE solutions based on MIP solvers with particular problem and optimization formulations), however we currently do not use them in our testbed. These are available and could be integrated and tested, if required.

Since our testbed is oriented towards research and testing of different mechanisms, the lifecycle management is following the experiment logic: a new experiment is deployed as a virtual network of particular VNFs instantiated depending on their realization; a scenario is initiated automatically after the deployment; the results/measurements are gathered as part of the scenario during its execution; finally, the network is stopped and removed after the experiment.

5.4.2 VNF Manager

In our system, the VNFs can be either:

Compiled native software, e.g., access controllers, traffic shapers, NAT, gateways, proxies, accelerators, etc. on the data plane (to instantiate a UPF of a particular kind), packaged as specially crafted Docker container images. These images interface with our runtime coordination system, which therefore can enforce specific start and stop behavior of services and interact with the installed software e.g., over (interactive or scripted) remote shell.

Java control applications, packaged as compiled Java classes, to be installed and run on top of the SDN controller system, i.e. FloodLight with our 5G Layer.

Combinations of the above, i.e. SDN Control Applications and particular software within different nodes. This mode can be used to implement distributed NF realizations.

In practice, all three are used. The maintenance of the images, Java classes and the hybrid applications is done by a build system.

5.4.3 VIM

The virtual network control GUI is used as a VIM, as it is possible to interact with the running VNFs from the GUI. For specific interactions, we use GUI exports, which become available as graphical menu properties of elements. For general interactions, a remote shell can be opened to any node.

5.4.4 SDN

The virtual network is using OpenVSwitch (OVS) as reference implementation of a (virtual) OpenFlow SDN switch. Although our testbed can use different SDN controllers, we currently rely on FloodLight.

Pox, Ryi, OpenDayLight and ONOS could be used alternatively, if required.

On top of FloodLight we are using a particular “5G Layer”, which includes common routines for processing and dispatching 5G events to different 5G NFs.

5.4.5 EMS

In our experimental facility site, we do not use any EMS for MANO elements per se.

5.5 Service Orchestration

Not applicable.

5.6 5G RAN and Core

5.6.1 eNB

No 4G equipment is used in the Munich experimental facility site.

- gNB
- Carrier Frequency: 3.4 GHz
- Bandwidth: 20 MHz
- Maximum UEs: 3
- Antenna gain: 12dBi
- TX power per antenna: 28dBm

5.6.2 Provisioning

As it is an experimental facility site, we do not use formal provisioning at the Munich experimental facility site. However, virtual networks can be scaled up and down, including in runtime.

5.6.3 EMS

Realized through the mentioned graphical network control GUI (see section 4.3.2).

We are implementing (Rel0 and Rel1) a home-made GUI to interact with the virtual network at the backend servers. The GUI considers management and operation tasks to control the virtual networks deployed at the backend servers. The GUI utilizes a 3rd-party network graph-tool as a visualization widget to visualize the virtual networks running at the backend servers. The network graph-tool of the GUI is also used when a new virtual network is designed including different types of network nodes, links and their topology before the virtual topology is deployed to the backend servers. The GUI also supports interactions with the network nodes and links such as node/link on and off, etc.. Each network node is implemented as a docker container integrated with OVS, every network link is a virtual link either intra-host or inter-host linux virtual link. Interactions with network nodes are done by RPC requests from the frontend to the backend, where a proxy server is responsible to distribute an RPC to corresponding network nodes

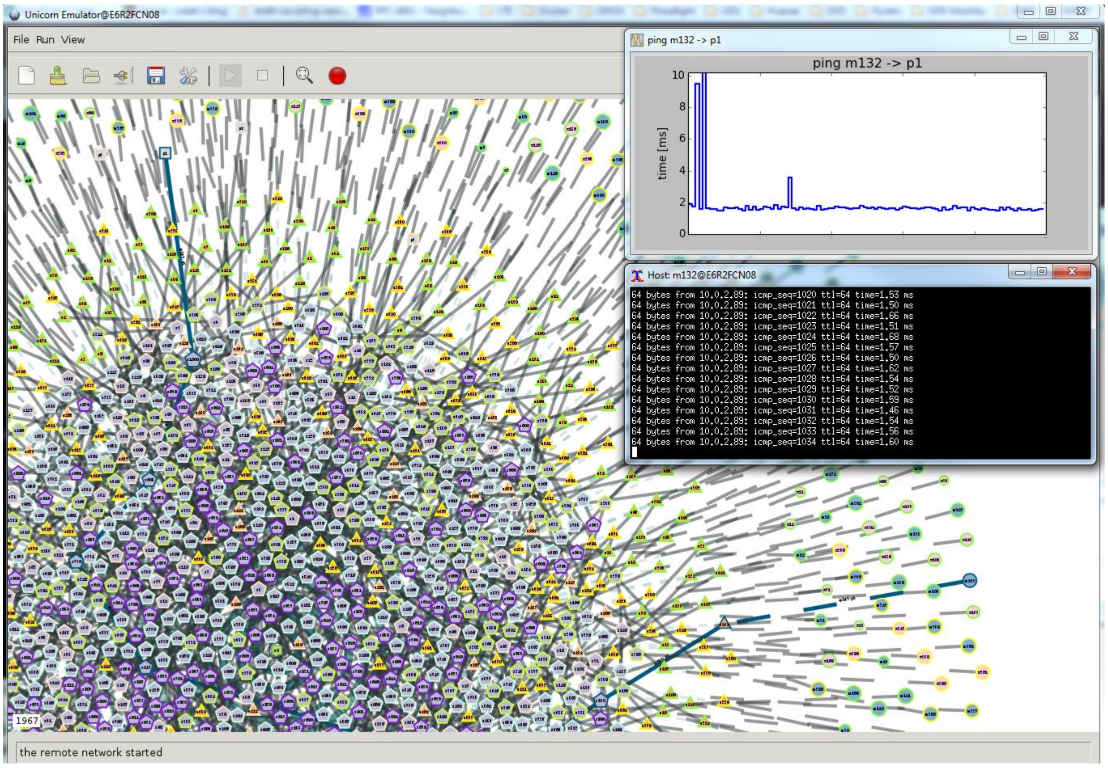

Figure 6 Network Control GUI running a virtual network of several thousands nodes

5.7 MEC

As this is experimental facility site, no ETSI MEC implementation is available. However, it is possible to deploy Docker containers and virtualization at the edge of network in order to provide low latency communication.

5.7.1 Architecture

5.7.2 Applications

5.8 Test Equipment

Not applicable as this is an experimental facility site.

5.9 Facility-site configuration (LLD)

Not applicable as this is an experimental facility site.

6 Dimensioning and BOM

Not applicable as this is an experimental facility site.

6.1 gUE

6.2 Transport Network

6.3 MANO and NFVI

6.4 5G RAN and Core

6.4.1 eNB

6.4.2 gNB

6.4.3 MME

Table 7 VNF vSGSN-MME vBOM

VNF Name	VM Type	AFFINITY RULES - 'INTRA VNF'	AFFINITY RULES - 'INTER VNF'	VM Workload type	No. of VMs per type	No. of vCPU per VM	RAM (GB) per VM	Storage (GB) per VM
vSGSN-MME	NCB							
	FSB							
	GEP							
	LC							

6.4.4 SGW/PGW

6.4.5 PCRF

6.4.6 HSS & Subscriber Database

6.4.7 Provisioning System

6.4.8 EMS

6.5 Test Equipment

Not applicable as this is an experimental facility site.

7 Cross-Facility-sites end-to-end design

Not applicable as this is an experimental facility site.

7.1 Facility-sites <this site name> and <2nd site name>

7.1.1 Provided services

7.1.2 Interconnection requirements

7.2 Facilities-site <this site name> and <3rd site name>

7.2.1 Provided services

7.2.2 Interconnection requirements

8 API

Not applicable as this is an experimental facility site.

8.1 Orchestration API

8.2 ENM API

[end of document]