HORIZON 2020
ICT - Information and Communication Technologies

Deliverable D3.1
Survey of Software Tools and Computing Architectures for Advanced Wireless Platforms

Project Acronym:	**EMPOWER**
Project Full Title:	**EMpowering transatlantic PlatfOrms for advanced WirEless Research**
Grant Agreement:	**824994**
Project Duration:	**36 months (Nov. 2018 - Oct. 2021)**
Due Date:	**31 October 2019 (M12)**
Submission Date:	**25 March 2020 (M17)**
Dissemination Level:	**Public**

Disclaimer

The information, documentation and figures available in this deliverable, is written by the EMPOWER project consortium under EC grant agreement 824994 and does not necessarily reflect the views of the European Commission. The European Commission is not liable for any use that may be made of the information contained herein.

D3.1 – Survey of Software Tools and Computing Architectures for Advanced Wireless Platforms

Executive Summary

This deliverable provides a survey of software tools, deployment and testing methodologies and computing platforms used in major current EU and US platforms. The study provides the initial EMPOWER software catalogue, documentation and collaboration methods and objectives for transatlantic mutualisation of components and systems for wireless experimentation. We provide an overview of open-source Core Network, Radio-Access Network and Edge Computing software packages which can be used to build proof-of-concept experimental networks. We also provide an overview of existing facilities offering experimental infrastructure which can be used to prototype 3GPP radio and core network technologies.

Table of Contents

1. Introduction

Advanced wireless platforms now benefit from growing open-source ecosystems that can be used to build their underlying components and on which to base their computing architecture. As a result, by reaching out and, furthermore, contributing to these ecosystems, research communities can mutualize software components and automatically induce collaboration on a global scale. This document surveys current efforts for providing open-source software components for building experimental 3GPP-compatible (i.e. 4G/5G/6G) wireless experimentation platforms. We restrict our treatment to 3GPP-compatible components since, from a wireless perspective, the latter represent the most significant part of deployed networks today and caters to the objectives of public-research funding initiatives such as H2020 in Europe and the NSF in the United States of America. Furthermore, the 3GPP standardization community is slowly opening up to the idea of working with open-source communities in order to develop proof-of-concept designs alongside the standardization process. This has already been highlighted for the evolution of 5G technologies recently by ITU-T SG13 publications (Lehmann, 2018). 3GPP implementations are also primary targets for industry-driven initiatives driven by the Linux foundation, such as OMEC and O-RAN.

Section 2 provides an overview of a selection of open-source 3GPP software implementations and some deployment frameworks that can be used to build proof-of-concept (PoC) designs. Section 2.1 first provides a high-level description of 4G and 5G core network architecture for readers not familiar with these technologies. It then covers the majority of open-source 4G and 5G core network components spanning from small-scale (laboratory) to medium-scale deployment scenarios. This includes details on the types of implementations, their community and collaboration models and their testing framework. Section 2.2 deals with the radio-access network (RAN) components and starts with an overview of common software-defined radio platforms that are used for PoC designs. In particular the capabilities in terms of radio-technologies and approximate cost are compared. We provide an overview of the two main open-source RAN initiatives, OpenAirInterface (OAI) and srsLTE, their community and collaboration model and testing framework. Finally we provide some initial information regarding the very recent O-RAN Amber release. In Section 2.3 we provide information about the open-source mobile edge computing initiatives Mosaic5g and Linux Foundation AKRAINO.

Section 3 provides an overview of some publicly-available infrastructure projects which provide experimental services for 3GPP technologies. In particular, we focus on the three EU ICT-17 platforms 5G-EVE, 5G-VINNI and 5GENESIS and provide details of the open-architecture components of these infrastructure initiatives. We also provide a summary of the 3 PAWR nodes in the USA, COSMOS, POWDER-RENEW and AERPAW as well as the Colloseum PAWR facility in Boston. Two additional sites OneLab-R2LAB in France and NITOS in Greece are also described.

2. Overview of Software Packages and Deployment Frameworks

This section provides an overview of a selection of open-source 3GPP software implementations and some deployment frameworks that can be used to build proof-of-concept (PoC) designs. If incomplete, the description provided here will be updated over time. We cover the core and radio-access network elements in use in various projects today as well as some ongoing projects to develop next generation implementations (i.e. 5G networking elements). Our survey is limited to 4G and 5G networking components.

In comparison to costly proprietary and closed-source 3GPP-compliant software, when combined with readily-available laboratory RF solutions, open-source solutions allow experimenters to build very low-cost small-scale PoC designs which allow the use of commercial cellular terminals (user-equipment or UE) as opposed to emulators or non-standards-compliant equipment. In addition, the open-source approach allows for interconnection of commercial (i.e. vendor) networking elements. The latter proves to be useful in collaborative projects with industry partners, in addition to providing testing solutions for industry research and development labs.

2.1 Open-source code Core Network Elements

The core network in a 3GPP radio system constitutes the collection of protocols needed to interconnect radio-access networks which provide services such as telephony and internet access to remote mobile users. The main entities comprising the network include both control plane and user plane functions.

2.1.1 Overview of the 4G Core Network

An overview of the 4G core network entities is shown in Figure 1 and Figure 2.

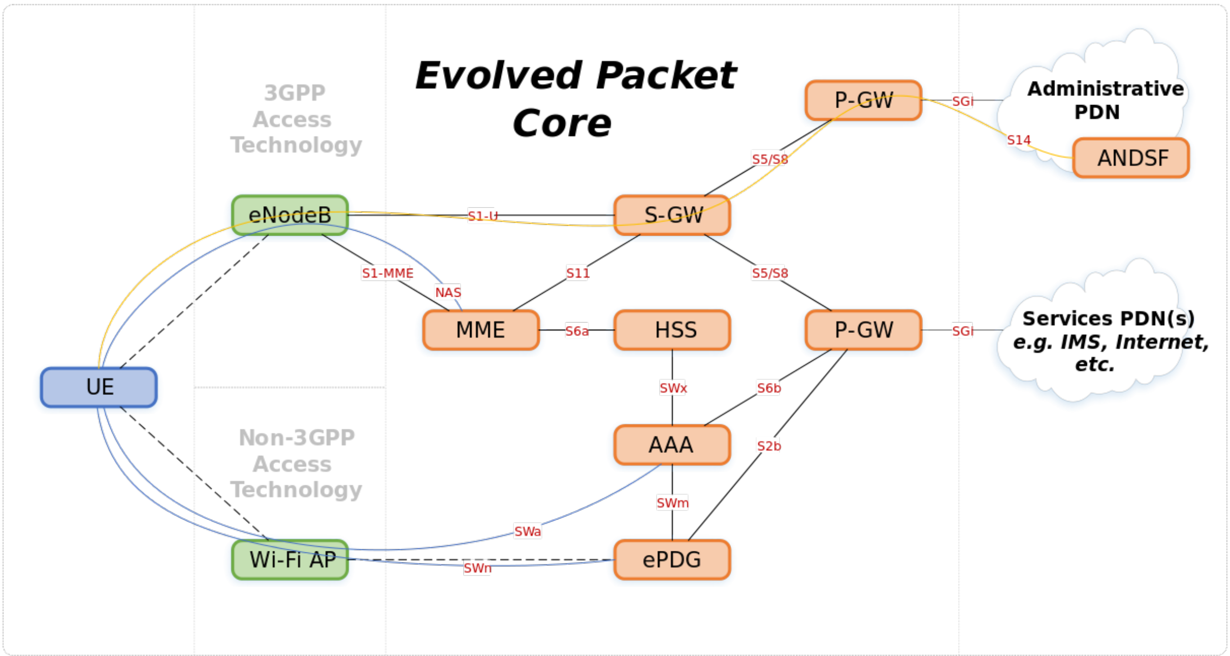


Figure 1: EPC Architecture (without charging/policy control interfaces) (By Joe Deu-Ngoc - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=37142420)

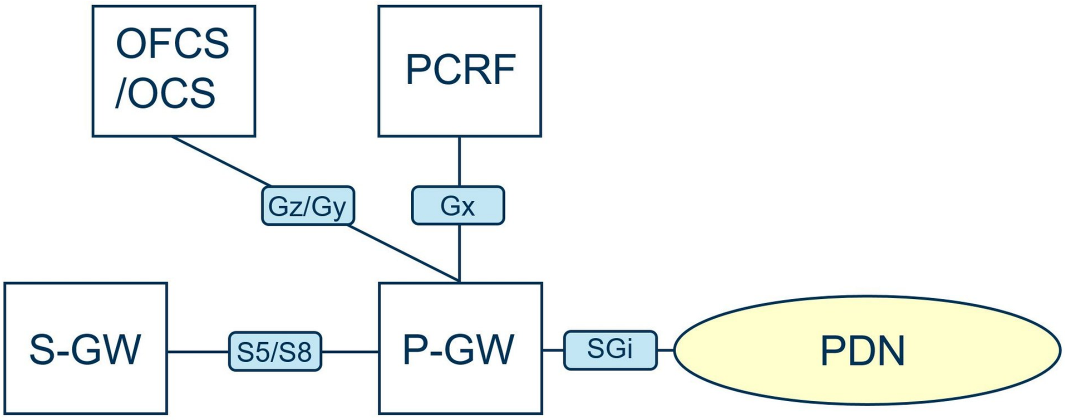


Figure 2: Additional user-plane functions for charging and QoS control (By Jralbala - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=63249858)

The 3GPP specifies the following entities comprising the 4G enhanced packet core (EPC):

- Home-Subscriber Server (HSS) : a database function which contains subscriber profile and authentication information. It communicates with MME for user authentication using the S6a interface and the Diameter protocol.
- Mobility-Management Entity (MME) : this is the main signalling entity in the 4G core network. It is responsible for all authentication and mobility services and is the endpoint for the so-called non-access-stratum (NAS) protocol which is used to communicate with the UE. The MME communicates with the radio basestation or eNodeB using the control-plane S1C interface using the S1AP application protocol on top of the SCTP transport protocol. The MME comprises both session managements (ESM) and mobility management (EMM) functions.
- Serving-Gateway (S-GW): this is a user-plane function which routes and forwards packets and is the mobility anchor during handovers between basestations. The S-GW communicates with the radio basestation (eNodeB or gNodeB for 5G non-standalone operation) on the S1U interface using the GTPU tunnelling protocol. It communicates with the MME using the S11 interface using the GTPC tunnelling protocol. Finally, it communicates with the P-GW using the S5/S8 control/user-plane interfaces.
- Packet Data Network (PDN)-Gateway (P-GW) is the gateway to external networks (e.g. the Internet) and provides means for policy enforcement, per-user packet filtering, charging support, lawful interception and packet screening. It communicates with the S-GW on S5/S8 control/user-plane interfaces using the GTPC/GPTU protocols and with external networks on the SGI interface. For the purpose of charging and QoS policy enforcement, it uses the Gx/Gy/Gz interfaces and diameter protocols.

In some small-scale implementations, the S-GW and P-GW functions are converged into a single entity and explicit interfaces and protocols are dropped. In order to support IMS (IP Multimedia System) services with an external IMS core network, an additional network element, the Policy and Charging Rules function (PCRF) (Figure 2) is also required. Operator-specific charging functions (OFCS/OCS) can also be used in commercial networks.

Later releases (as of 3GPP Release 14) of the 4G core network architecture provided for a so-called control-user plane separation (CUPS) whereby the S/P-GW functions are split into two entities. This added an additional native 3GPP packet-forwarding control plane (PFCP) on top of the UDP transport protocol as shown in Figure 3. The benefit of this, is to allow a distributed architecture for the user-plane entities (S/P-GWU) while keeping the ability to centralize the control plane components. In this separation scenario, additional traffic detection function (TDF) network entities located at the ingress and egress point to the packet data network can be seen. These are responsible for service data flow traffic detection, application detection, deep packet inspection and event reporting.

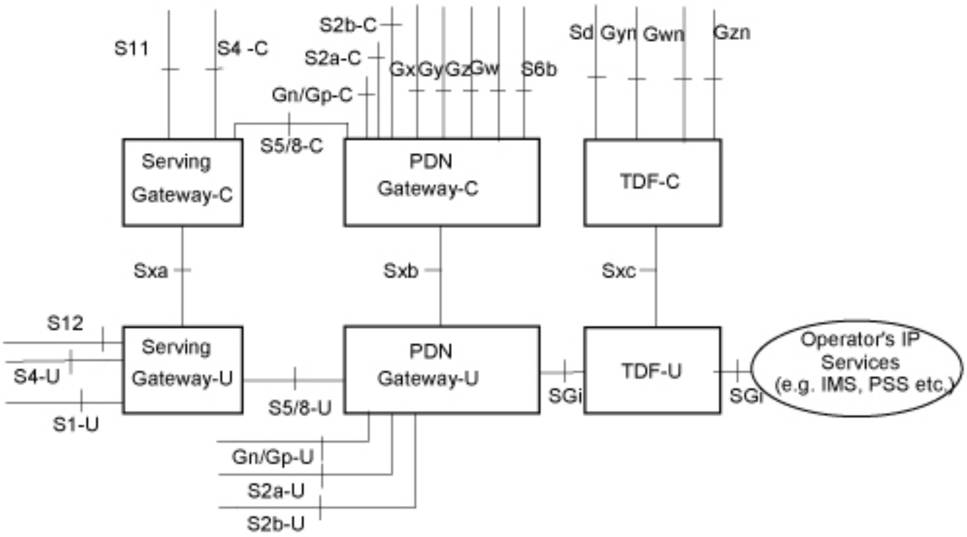


Figure 3: Control-plane User-Plane Separation

For the purpose of non-standalone operation with 5G radio technology, the 4G core network in used by adding an additional S1U interface between the S-GW and the 5G gNodeB. No core network control plane interfaces are provided for the gNodeB which only acts as a high-throughput pipe for user-plane traffic. The great majority of 5G networks will operate in this fashion during the roll-out phase of 5G. More information will be given in Section 2.2.

2.1.2 Overview of the 5G Core Network

An overview of the 5G core network entities is shown in Figure 4. It can be seen as a simplification of the 4G core with built-in optimization for cloud-native functionality.

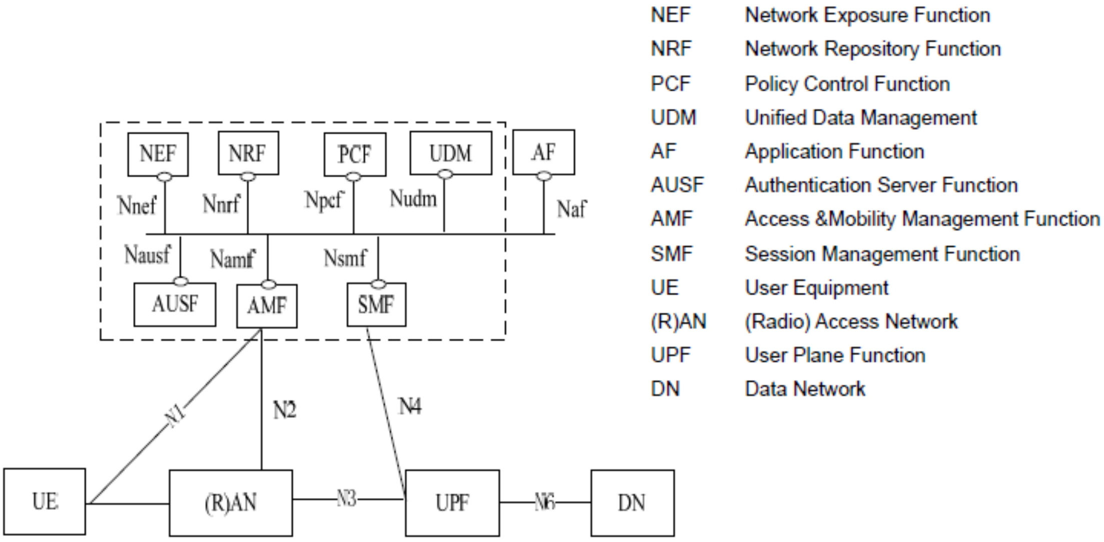


Figure 4: Service-based 5G Core Architecture

In this respect, the AUSF is functionally equivalent to the HSS and the PCF to the PCRF. The new Network Exposure (NEF) and Network Repository (NRF) functions are added in a cloud-native implementation of the 5G Core. The latter constitute secure means for exposing and querying network services and capabilities of deployed 3GPP network functions. All entities in the zone in the dashed rectangle are intended to be implemented in a cloud-native fashion. As a result, many of the specific interfaces between network entities are implemented using a message bus architecture instead of point-to-point protocols.

Closer to the radio network entities, the two basic functions of the MME (Mobility and Session Management) are split into two entities, the Access & Mobility Management Function (AMF) and the Session Management Function (SMF). The AMF connects to the radio network using the N2 interface which is functionally equivalent to the S1C interface in the 4G EPC. Moreover, it uses an application protocol, the NGAP, which is similar to the S1AP 4G protocol. The NAS protocol with the UE is conveyed via the tunnelled N1 interface. The user-plane function (UPF) plays the role of a converged S/P-GWU with the S/P-GWC function included in the SMF. The UPF communicates with the radio-basestations using the N3 protocol using the GTPU transport protocol. The latter is similar to the S1U 4G user-plane interface and protocol. The UPF communications with the SMF using the N4 interface using the PFCP protocol which provides the same functionality as the Sx 4G interface in the CUPS EPC.

2.1.3 openairCN

Openair-CN is a community-based implementation of a subset of the 3GPP Release-15 EPC specifications. More specifically, it contains the implementation of the MME, HSS, S-GW+P-GW. It comes in two flavours, the simplest one is shown in Figure 5 where the S-GW and P-GW functions are not separated into their control and user-plane components.

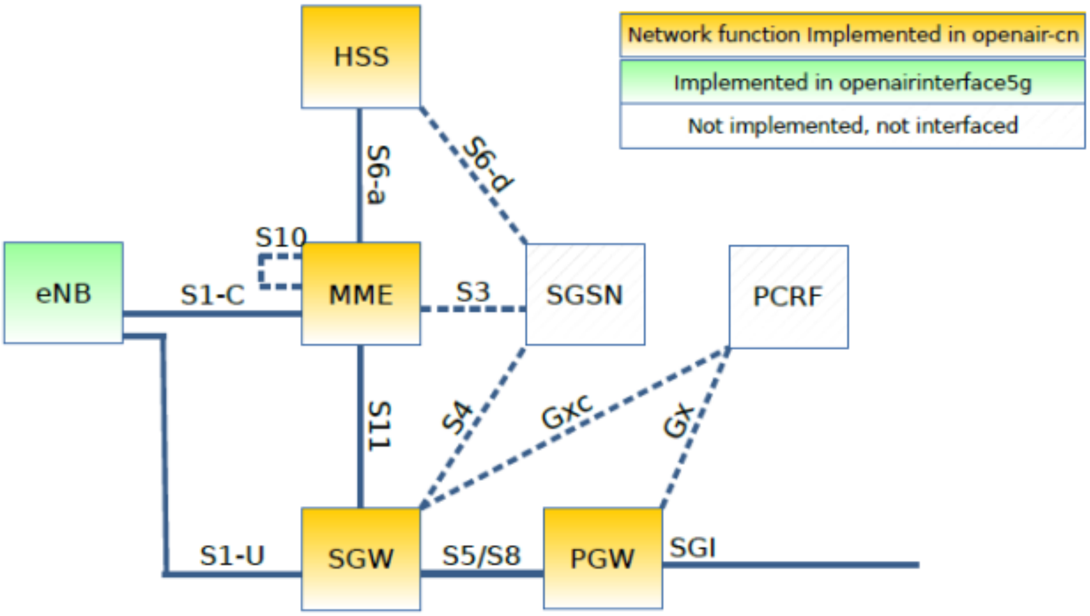


Figure 5: openairCN

A second flavour of the S/P-GW functions which implements the CUPS separation described in Section 2.1.1. The CUPS components are similarly found on github (https://github.com/OPENAIRINTERFACE/openair-cn-cups).

openairCN is distributed via github (https://github.com/OPENAIRINTERFACE/openair-cn) under an Apache V2.0 license and is maintained by the OpenAirInterface Software Alliance (OSA). which also provides a continuous-integration continuous-delivery testing framework (CI/CD) for community contributions to the codebase. CI is achieved using servers located at EURECOM's open5GLab in Sophia Antipolis (see open5gLab infrastructure in Section 3.1.1) and CD both at open5GLab and at INRIA's R2Lab (see Section 3.5). The openairCN CI framework at open5GLab makes use of commercial emulated RAN testers and OAI eNodeB/gNodeB with commercial smartphones. The CD framework at R2Lab uses the OAI eNodeB and commercial smartphones.

openairCN was originally developed by EURECOM but later other groups contributed to the development and testing of different entities. Facebook Connectivity and its partner Radisys contributed to an earlier version of the MME which later was integrated into Magma (see Section 2.1.4). Blackned (https://blackned.de/) and B-COM (https://b-com.com/) later contributed to the MME and the CUPS S/P-Gw. Sprint (https://www.sprint.com/) upgraded the orginal HSS implementation to Rel-14 compliance and integrated it into the OMEC EPC implementation (See Section 2.1.6) along with other non-openairCN entities.

2.1.4 Facebook MAGMA

MAGMA is an open-source project maintained as a product by Facebook connectivity. It is a small-scale EPC similar to openairCN but augmented by network management, orchestration and performance metric measurement functions. In addition it provides a Federation Gateway to interface with 3GPP standardized interfaces in an operators existing Core Network. It is designed to be deployed in a cloud-native environment making use of the virtualbox hypervisor and vagrant. The main target of this project is to provide simple-to-deploy connectivity in rural or remote areas of the globe without existing 4G coverage. It also includes non-3GPP interfaces for authentication and user-plane functions to simplify deployment for small markets.

Magma currently uses an older version of openairCN for the EPC but all other network functions are specific to this project. Facebook maintains an internal CI/CD framework for contributions to Magma. All non openairCN functions in Magma are BSD License licensed. openairCN is a version forked from the openairCN repository in 2018. An ongoing effort by EURECOM and Facebook in 2020 aims to maintain and distribute a merged version of Magma in the openairCN repository

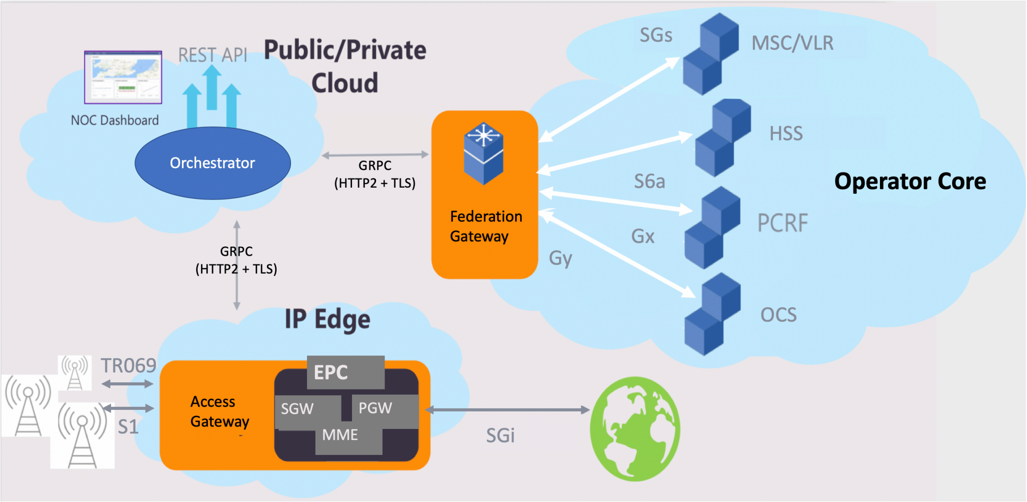


Figure 6: Facebook Connectivity Magma

2.1.5 nextEPC

NextEPC is an open-source implementation of a Release 13 3GPP core network including MME, S-GW, P-GW, HSS and PCRF. The NextEPC MME provides a standard compliant S1U/S1C interfaces to the eNodeBs and S11 interface to SGW as well as S6a to the HSS. NextEPC's SGW implements the S11 interface which is connected to MME, and S5 interface to the PGW. NextEPC PGW plays a role as an edge router in IP networks. It equipped with the S5 interface and SGi interface towards the Internet, and S7 interface to PCRF. NextEPC HSS is the user subscription database. It implements the S6a interface towards MME using the DIAMETER protocol. NextEPC PCRF controls the policies and rules for QoS of LTE users and bearers. It provides the Gx interface to PGW.

NextEPC is written in C language and is dual-licensed as a commercial product. A publicly available version distributed on github (https://github.com/nextepc/) uses a restrictive open-source license, Gnu Affero General Public License (AGPL) and allows the software to be used freely, modified and redistributed under the same license. The creators also relicense the code for commercial purposes. Unlike the other Core Network projects described in this document, because of the dual-licensing strategy, it does not seem to be a community-driven project.

2.1.5.1 NextEPC 5G roadmap

A new version of NextEPC to be released in 2020 will include both EPC and 5GC in a single software package for making a seamless transition from 4G to 5G, so that it will be versatile in any type of early 5G deployment scenario. One the other hand, NextEPC will promote a distributed deployment with a dynamic service discovery technology of cellular core services so scalability and robustness can be achieved.

2.1.6 OMEC

OMEC claims to be the first full-featured, scalable, high performance open-source EPC. It is an open-networking foundation (ONF) project and part of the CORD (central-office rearchitected as a data-center) family of projects. It is developed primarily by an operator-driven consortium and is an industry-led open-source community. The primary contributors are Sprint, AT&T, Intel and Deutsche Telekom. OMEC has been optimized to handle a large number of devices in order to cater to dense 5G and IoT deployments. It is designed to be used as a stand-alone EPC, and is also an upstream project for the COMAC platform that (among other things) is integrating both mobile and fixed subscriber management functions.

OMEC is built using an NFV architecture and includes:

- Complete connectivity, billing and charging capabilities;
- 3GPP Release 13 compatibility;
- Support large numbers of subscribers with a high performance DPDK based dataplane;
- Optimization for lightweight cost-effective deployments and IoT applications;
- Integrated CI/CD test and verification capabilities.

2.1.6.1 OMEC Architecture

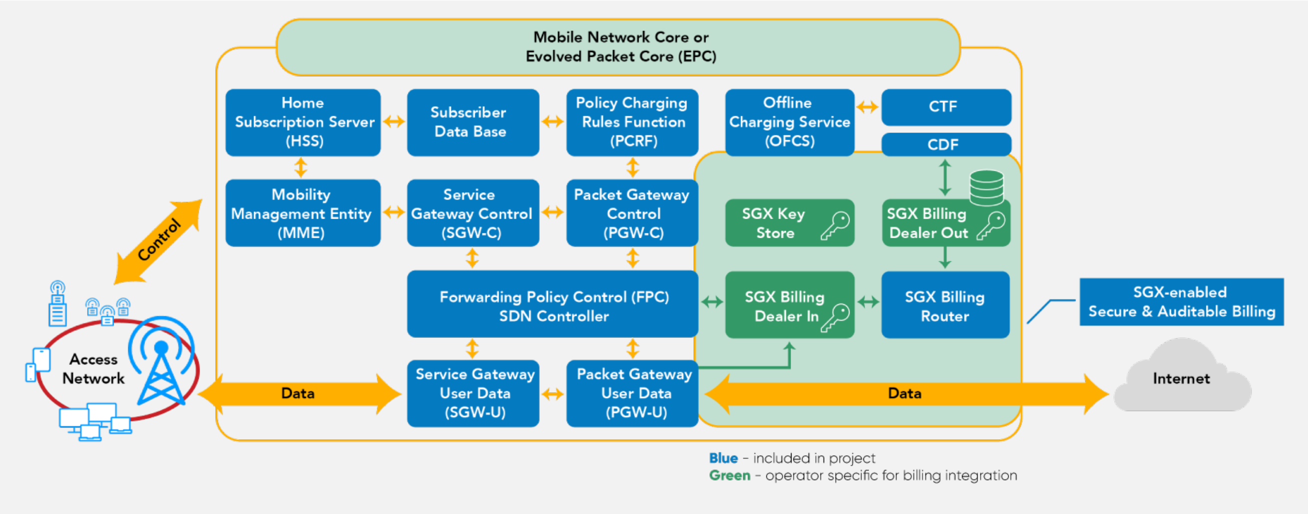


Figure 7: OMEC EPC and Charing Components

- SGW-C, PGW-C (Includes embedded PCEF), SGW-U, PGW-U, MME, HSS, HSS Database, Diameter Capability, PCRF, Forwarding Policy Control SDN Controller, CTF, CDF, SGX Billing Router CLI, Logging and Statistics Interface APIs to VNFs;
- DPDK based traffic generator for testing S1u and Sgi user planes;

- Deployment automation for bringing up core network VNFs, associated networking, package installation, provisioning, x and configuration.

OMEC is distributed on a public github repository (https://github.com/omec-project) and is written primarily in C/C++. It includes a sophisticated Jenkins-based CI/CD framework for developers to the github repository.

2.1.7 Open Core Network Initiative (Telecom Infra Project)

The Telecom Infra Project (TIP) recently announced an initiative dubbed the *Open Core Network group (OCN)* which aims to develop an open, cloud-native, and converged core that is a collection of microservices implementing various core network functions ("open, flexible and extensible"):

- Running on standardized software and hardware infrastructure ("infrastructure agnostic");
- Supporting 3GPP 5G Core (5GC) and LTE Evolved Packet Core (EPC) for licensed, unlicensed (e.g. Wi-Fi), and shared spectrum (e.g., CBRS) networks ("access agnostic");
- Enabling seamless migration from 4G EPC to 5GC in both Non Stand-Alone (NSA) and Stand-Alone (SA) modes.

The goal is to innovate on the packet core technologies across any access wireless networks operating on licensed, unlicensed and shared spectrum; develop microservice, orchestration and automation frameworks on OCN platform, and support an ecosystem of developers, OEMs, SIs, MNOs, and ISPs around OCN based solutions. As shown in Figure 8, the project aims to focus firstly on Applications and Services, and specifically to design, develop and deliver a set of production-grade microservices implementing 3GPP 4G/LTE, 5G, shared spectrum (e.g., CBRS) and Wi-Fi core network functions and APIs between such functions as well as between these functions and RAN elements or Network Management Systems. Secondly, the project considers an orchestration framework for integrating, deploying and managing OCN microservices, including FCAPS, metrics, analytics and monitoring for OCN microservices only. Finally, the OCN group will design, develop and deliver an infrastructure and test automation, i.e., CI/CD framework for running and testing OCN microservices only.

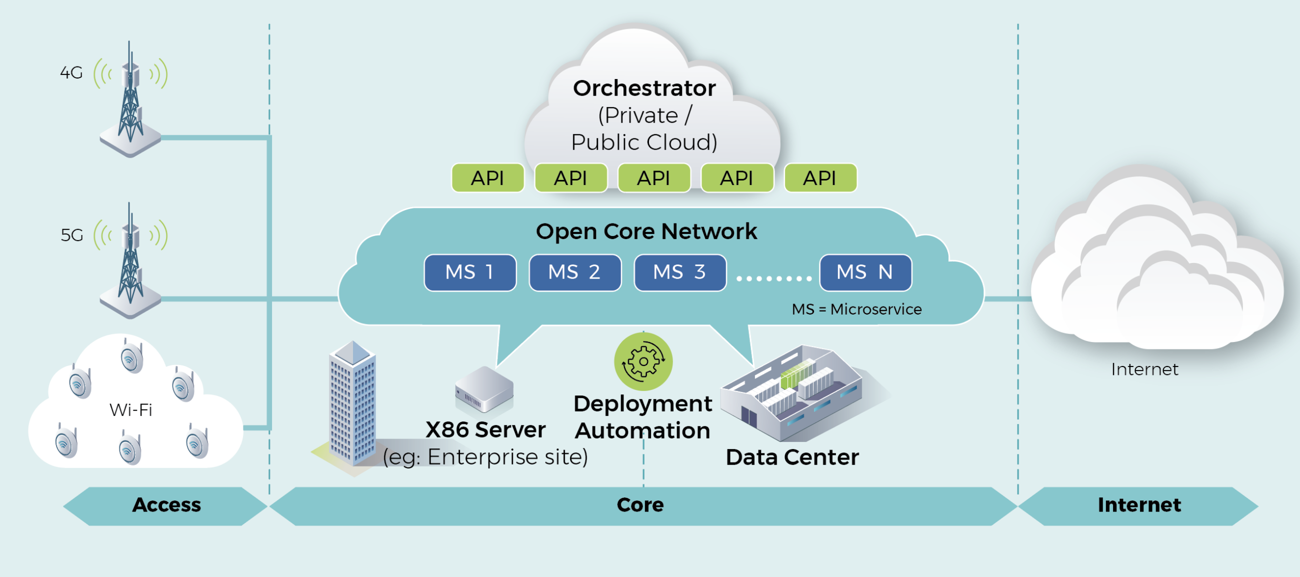


Figure 8: TIP Open-Core Network Project

The group will work on a quarterly cadence and produce the following set of Deliverables, as appropriate, based on the workstream defined milestones:

- Draft product requirements specification of open core based on prioritized use cases;
- Software Microservices implementation of core network functions and;
- components Orchestration of microservices;
- API specifications;

- Standardized hardware blueprints;
- Automation tools, frameworks and scripts.

The initial 2020 portion of the software project is planned to focus on delivering a “Minimum Viable Core (MVC)” that will address the UPF, AMF, SMF, AUSF and UDM 5GC functions in the form of microservices (and the relevant EPC functions as a “converged core”).

The group will collaborate via liaison agreements with other industry groups working on core network technologies, such as:

- OpenAirInterface Software Alliance (https://www.openairinterface.org/): Contributing seed code and;
- development services for OCN platform;
- OpenStack Foundation (https://www.openstack.org/): Contributing cloud-native infrastructure, including containers, container orchestration and automation framework (not necessarily OpenStack platform);
- Open Network Foundation (https://www.opennetworking.org/): Contributing components of OCN platform (to be determined);
- Linux Foundation (https://www.linuxfoundation.org): To be determined.

and incorporate advances in core networking technologies at the industry level, especially at the advent of 5G, CBRS and Wi-Fi6 technologies.

The timeline for the project work is shown in Figure 9.

Figure 9: OCN Group Project Timeline

2.2 Open-source code Radio-Access Network Elements

We describe here a very high-level overview of the 5G 3GPP radio-access network functions. The 4G equivalents are similar in terms of their basic functionality and are not treated in detail in what follows. The 5G RAN elements and their nearest neighbours in the 5G core network are shown in Figure 10. The latter comprise the so-called gNodeB and ng-eNodeB network entities. The ng-eNodeB is an enhanced version of the 4G radio basestation or eNodeB. The RAN nodes are interconnected directly via the Xn (X2 in 4G) interface and interact with the 5G core network on the NG interface (see Section 2.1.2.)

These RAN entity are not necessarily integrated in a single computing unit as is the case in currently deployed 4G networks. They can be disaggregated and distributed across a wide-area network or within a data-center. One disaggregated implementation using the standardized F1 interface is shown in Figure 11. Here one of the gNodeB entities is split into the CU (Centralized Unit) and DU (Distributed Unit) entities. This F1 interface allows the DU to perform lower-layer functions (RLC layer and below) in smaller geographic regions (small cell or micro-cell radius of up to a few kilometers) and the CU to perform aggregated higher layer services over large regions (approximately 200 km radius.)

Another non-3GPP interface for RAN disaggregation known as NFAPI (Forum, 2017) is specified by the small-cell forum. This interface option is shown in Figure 12 and shows the split between the so-called Physical Network Function (PNF) which comprises the physical-layer and the so-called Virtualizable Network Function (VNF) which comprises the remainder of the RAN functions.

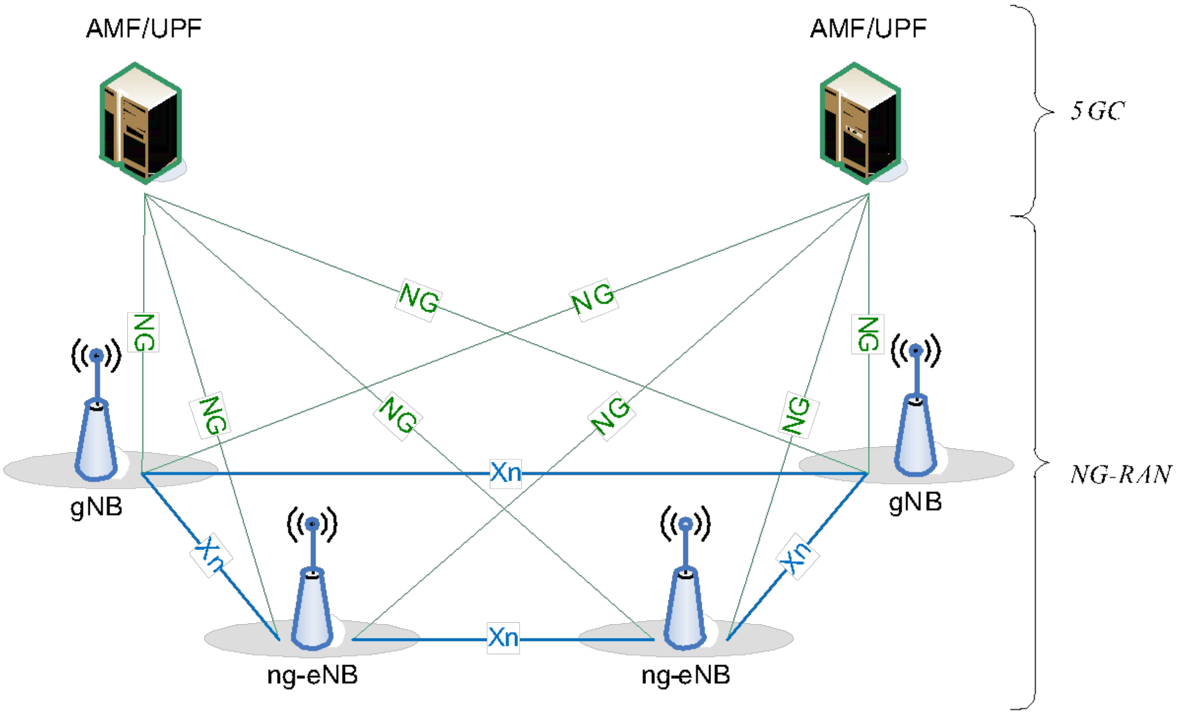


Figure 10: 5G RAN Elements

The responsibilities of the gNodeB/eNodeB are summarized as

- Functions for Radio Resource Management: Radio Bearer Control, Radio Admission Control, Connection Mobility Control, Dynamic allocation of resources to UEs in both uplink and downlink (scheduling);
- IP header compression, encryption and integrity protection of data;
- Selection of an AMF at UE attachment when no routing to an AMF can be determined from the information provided by the UE;
- Routing of User Plane data towards UPF(s);
- Routing of Control Plane information towards AMF;
- Connection setup and release;
- Scheduling and transmission of paging messages (originated from the AMF);
- Scheduling and transmission of system broadcast information (originated from the AMF or O&M);Measurement and measurement reporting configuration for mobility and scheduling;
- Transport level packet marking in the uplink;
- Session Management;
- Support of Network Slicing;
- QoS Flow management and mapping to data radio bearers;
- Support of UEs in RRC_INACTIVE state;
- Distribution function for NAS messages;
- Radio access network sharing;
- Dual Connectivity.

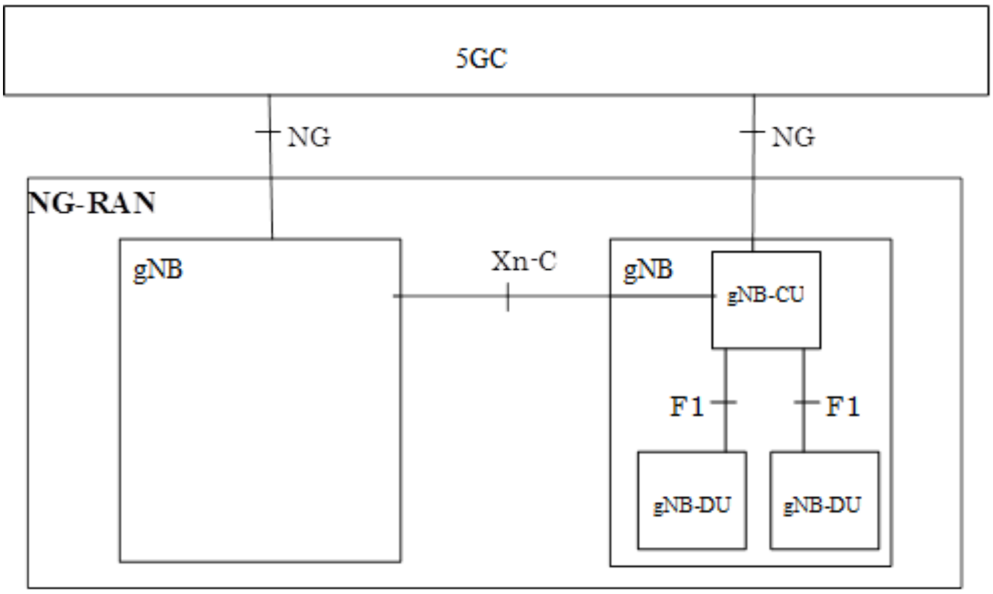


Figure 11: Disaggregated RAN (CU/DU split)

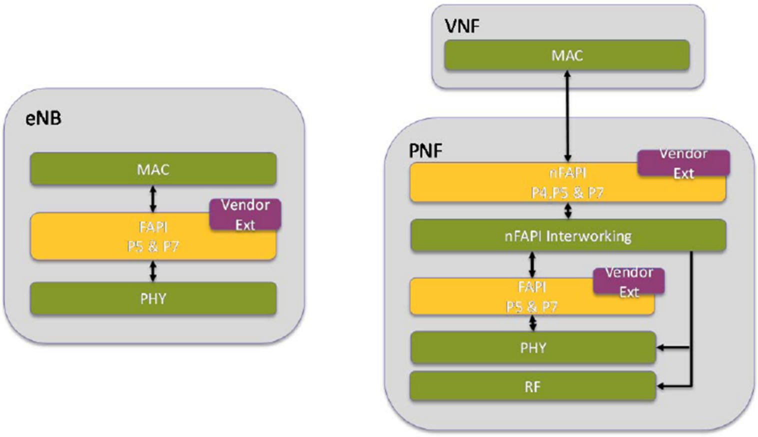


Figure 12: Disaggregated RAN (NFAPI interface)

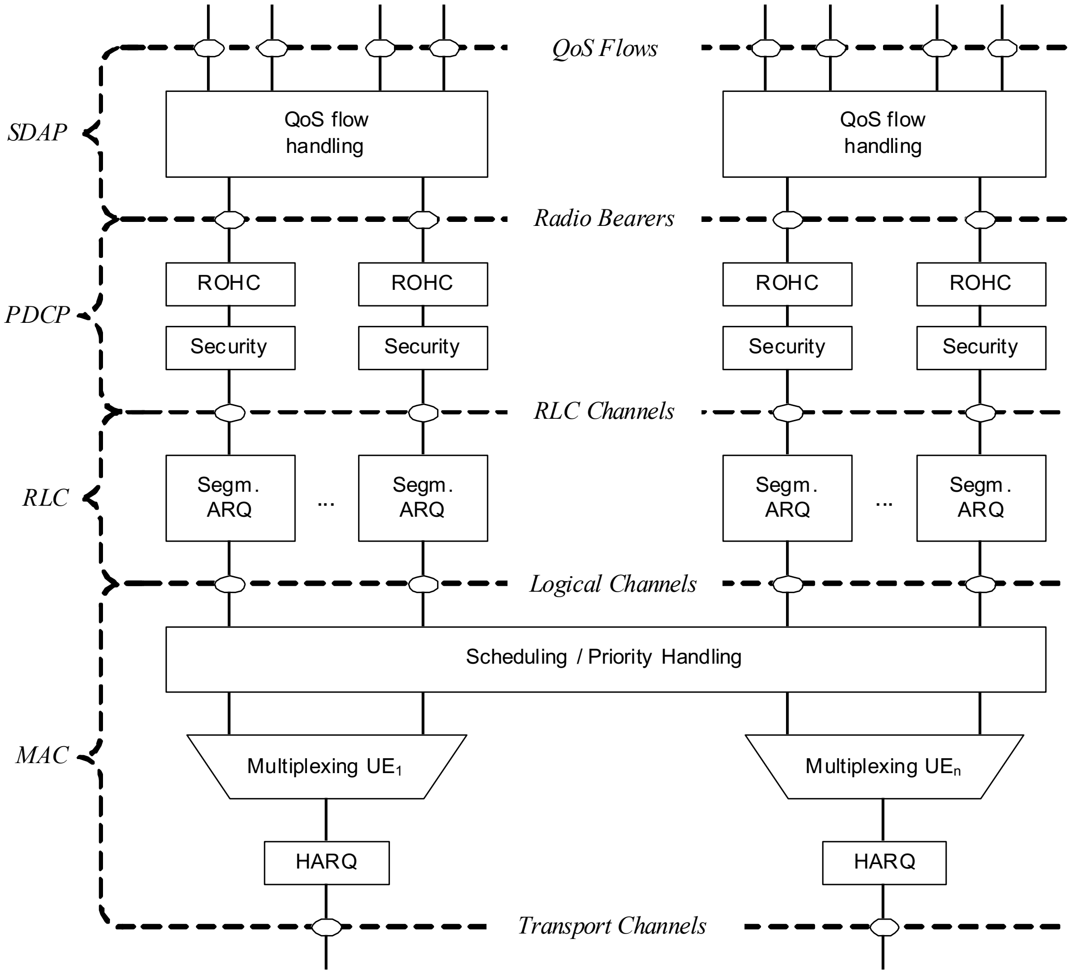


Figure 13: Layer 2 gNodeB/eNodeB Functions

The 3GPP RAN Functions correspond to the procedures which are executed by the eNodeB for 4G or gNodeB for 5G. The data-plane functions and their interconnections are shown in Figure 13. The overall set of functions are:

- Physical layer (PHY);
- Medium-Access Control (MAC) Layer;
- Radio-Link Control (RLC) Layer;
- Packet Data Convergence Protocol (PDCP);
- Service Data Adaptation Protocol (SDAP, 5G and enhanced 4G RAN);
- Radio resource control (RRC).

The main services and functions of the MAC sublayer include:

- Mapping between logical channels and transport channels;
- Multiplexing/demultiplexing of MAC SDUs belonging to one or different logical channels into/from transport blocks (TB) delivered to/from the physical layer on transport channels;
- Scheduling information reporting;
- Error correction through HARQ (one HARQ entity per carrier in case of CA);
- Priority handling between UEs by means of dynamic scheduling;
- Priority handling between logical channels of one UE by means of logical channel prioritisation;
- Padding.

The basic logical channels supported by the gNodeB/eNodeB for conveying signalling and user-data are

- Broadcast Control Channel (BCCH): a downlink channel for broadcasting system control information;
- Paging Control Channel (PCCH): a downlink channel that transfers paging information and system information change notifications;
- Common Control Channel (CCCH): channel for transmitting control information between UEs and network. This channel is used for UEs having no RRC connection with the network;
- Dedicated Control Channel (DCCH): a point-to-point bi-directional channel that transmits dedicated control information between a UE and the network. Used by UEs having an RRC connection;
- Dedicated Traffic Channel (DTCH): point-to-point channel, dedicated to one UE, for the transfer of user information. A DTCH can exist in both uplink and downlink.

The RLC sublayer supports three transmission modes:

- Transparent Mode (TM)
- Unacknowledged Mode (UM)
- Acknowledged Mode (AM)

and its main services and functions of the RLC sublayer depend on the transmission mode. These include:

- Transfer of upper layer PDUs
- Sequence numbering independent of the one in PDCP (UM and AM)
- Error Correction through ARQ (AM only)
- Segmentation (AM and UM) and re-segmentation (AM only) of RLC SDUs
- Reassembly of SDU (AM and UM)
- Duplicate Detection (AM only)
- RLC SDU discard (AM and UM)
- RLC re-establishment
- Protocol error detection (AM only)

The main services and functions of the PDCP sublayer for the user plane include:

- Sequence Numbering;
- Header compression and decompression: ROHC only;
- Transfer of user data;
- Reordering and duplicate detection;
- PDCP PDU routing (in case of split bearers);
- Retransmission of PDCP SDUs;
- Ciphering, deciphering and integrity protection;
- PDCP SDU discard;
- PDCP re-establishment and data recovery for RLC AM;
- Duplication of PDCP PDUs.

The main services and functions of the PDCP sublayer for the control plane include:

- Sequence Numbering;
- Ciphering, deciphering and integrity protection;
- Transfer of control plane data;
- Reordering and duplicate detection;
- Duplication of PDCP PDUs.

The SDAP entity is new to 5G RAN and has been added to the enhanced 4G RAN. It is a critical function for 5G QoS and slicing procedures providing for differentiated services and traffic classification. Its main services and functions of SDAP include:

- Mapping between a QoS flow and a data radio bearer;
- Marking QoS flow ID (QFI) in both DL and UL packets.

The 5G RRC entity is responsible for

- Broadcast of System Information related to AS and NAS;
- Paging initiated by 5GC or NG-RAN;
- Establishment, maintenance and release of an RRC connection between the UE and NG-RAN including;
- Addition, modification and release of carrier aggregation;
- Addition, modification and release of Dual Connectivity in NR or between E-UTRA and NR;
- Security functions including key management;
- Establishment, configuration, maintenance and release of Signalling Radio Bearers (SRBs) and Data Radio Bearers (DRBs);
- Mobility functions including;
- Handover and context transfer;
- UE cell selection and reselection and control of cell selection and reselection;
- Inter-RAT mobility;
- QoS management functions;
- UE measurement reporting and control of the reporting;
- Detection of and recovery from radio link failure;
- NAS message transfer to/from NAS from/to UE.

It is essentially a state-machine handling all RAN signalling exchanges with the UE, encoding/decoding of ASN.1 messages and relaying NAS information to the UE on behalf of the AMF.

2.2.1 Open Software-Defined-Radio Devices

The canonical example of the software radio device is the Universal Software Radio Peripheral (USRP) family originally commercialized by Ettus Research (www.ettus.com) which is now a subsidiary of National Instruments (www.ni.com). Ettus produces a large variety of devices which can be used to build experimental 4G/5G radio systems and are supported by the main open-source software packages described in this section. The Ettus devices are generally of high quality in terms of RF performance and software stability. Ettus maintains an open-source driver and development environment, USRP Hardware Driver (UHD), for interfacing with the USRP family of device. UHD is licensed with a GPL V3.0 for non-profit research or commercial implementations making use of compatible licensing technology. It can be licensed differently for closed and/or general commercial applications when used with or without USRP devices. In general, Ettus does not guarantee real-time performance and achieving robustness comparable to commercial remote-radio head solutions depends heavily on the system implementation (host computer, bus technology, and application software). It can be said that solutions based on Ettus devices are sufficient for proof-of-concept designs and non-critical small-scale deployments. The networked devices (X3x0,N3x0) have the advantage of being able to be remotely deployed with respect to the host processing system. At the very least, the USB3.0-based devices require a host computer to be locally deployed with the radio-device.

Nuand (www.nuand.com) produces another family of similar low-cost prototype devices known as the BladeRF. In terms of its RF and acquisition technology, the BladeRF 2.0 is essentially the same as the B210 device from Ettus in a different package and with different software API to interface with the host computer. Nuand also produces compatible power and low-noise amplifiers to use with the BladeRF 2.0.

Finally, we can mention the LimeSDR family of devices which use RF and acquisition technology based on Silicon from Lime Microsystems. The boards were crowd-funded and are still distributed on CrowdSupply (www.crowdsupply.com). A community-based development effort (https://myriadrf.org/) provides the driver as an open-source library that can be linked to the application executable. These devices have a more limited frequency range but are sufficient for 4G/5G experimentations below 4 GHz. The RF quality is inferior to the other devices.

We list the primary devices used in various projects along with their basic 4G/5G-related capabilities and approximate cost in Table 1.

Table 1: SDR Transceivers for Experimental 4G/5G Networking

Device Name	Manufacturer	Interface	Capabilities	Approximate Cost
USRP B200/B205/B200-mini/B205-mini	Ettus	USB3.0	4G (<=20 MHz 1x1) 5G (<=40 MHz, 1x1)	600-1200 €
USRP B210	Ettus	USB3.0	4G/5G (<= 20 MHz, 2x2) 5G (<= 40 MHz, 1x1)	1200-1500 €
USRP X300/X310	Ettus	10G Ethernet, PCIe	4G/5G (<= 80 MHz, 2x2)	4000-6000 €
USRP N300	Ettus	2x10G Ethernet	4G/5G (<= 100 MHz 2x2)	6500 €
USRP N310	Ettus	2x10G Ethernet	4G/5G (<= 100 MHz, 4x4)	~10000 €
USRP N320/N321	Ettus	2x10G Ethernet, 1x40G Ethernet	4G/5G (<= 200 MHz, 2x2)	~10000 €
Blade RF 2.0	Nuand	USB3.0	4G/5G (<= 40 MHz 1x1) 4G/5G (<= 20 MHz, 2x2)	450 €
LimeSDR-mini	Crowdsupply	USB3.0	4G/5G (<= 50 MHz, 2x2)	150 €

2.2.2 *Openairinterface5g*

Openairinterface5g is an open-source implementation and development community producing a full software implementation of the 3GPP radio-access network components. Specifically, the following elements are provided:

- 3GPP 5G NR (including RU,DU and CU node functions);
- 3GPP 4G LTE (including RU,DU-LTE and CU-LTE node functions);
- 3GPP 4G LTE-M;
- 3GPP 4G NB-IoT;
- 3GPP 4G LTE-Sidelink (ProSe / V2X).

The software targets both readily-available SDR solutions (see Table 1) and some high-performance commercial remote-radio units. Openairinterface5g provides several industry-standard interfaces (NFAPI, FAPI, ECPRI) and can also be interfaced with third-party libraries for proprietary interfaces. openairinterface5g is distributed on EURECOM's gitlab server (gitlab.eurecom.fr/oai/openairinterface5g.git) and community development is integrated with a CI/CD framework that triggers testing at various sites. This is depicted in Figure 14 where it is shown that code changes committed to the

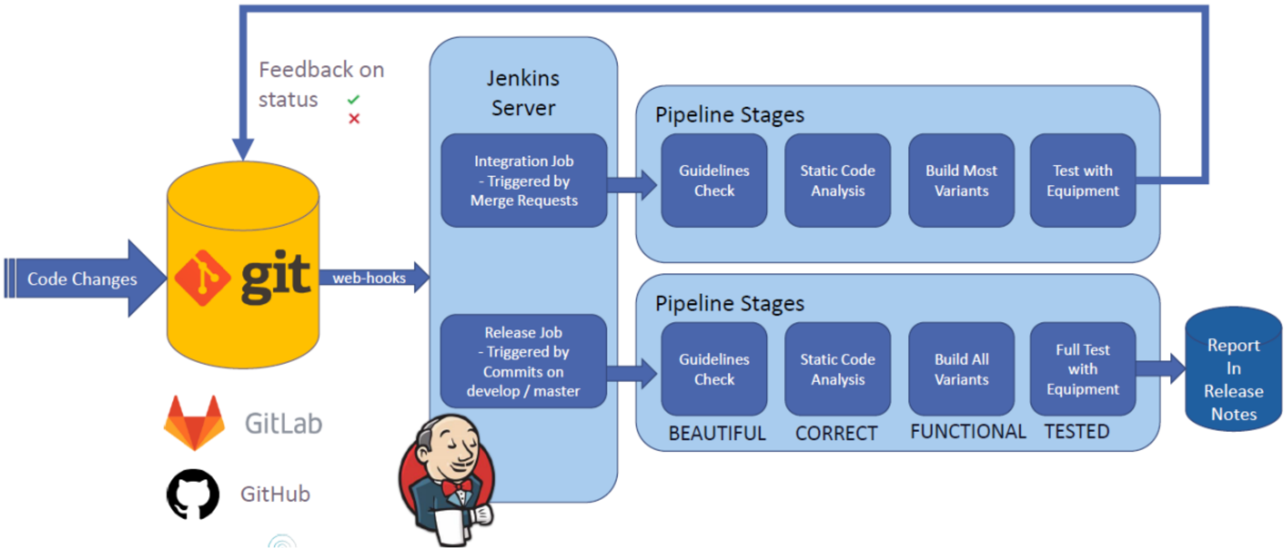


Figure 14: openairinterface5g CI/CD framework

gitlab server triggers automatic testing procedures. These include guideline checking, static code analysis (cppcheck), building of common variants and testing with equipment and simulation targets. Today CI/CD testing occurs on EURECOM's open5gLab infrastructure (see Section 3.1.1). The CD procedure occurs weekly by reserving time on the R2Lab infrastructure (see Section 3.5). The OAI community is currently also integrating the CD component with the POWDER (see section 0), COSMOS (see section 3.6) and NITOS (see section 3.4) infrastructure. In the context of 5G-EVE the CD component is also being integrated with Orange's infrastructure (see section 3.1.2) and B-COM's Wireless Edge Factory (see section 3.1.3).

The current functional decomposition of the OAI computing elements is shown in Figure 15. OAI supports several interfaces for disaggregating the RAN functions. Each block in the figure is a function that can either be containerized using Docker containers with real or virtual network interfaces or can be executed on a bare-metal server with real network interfaces.

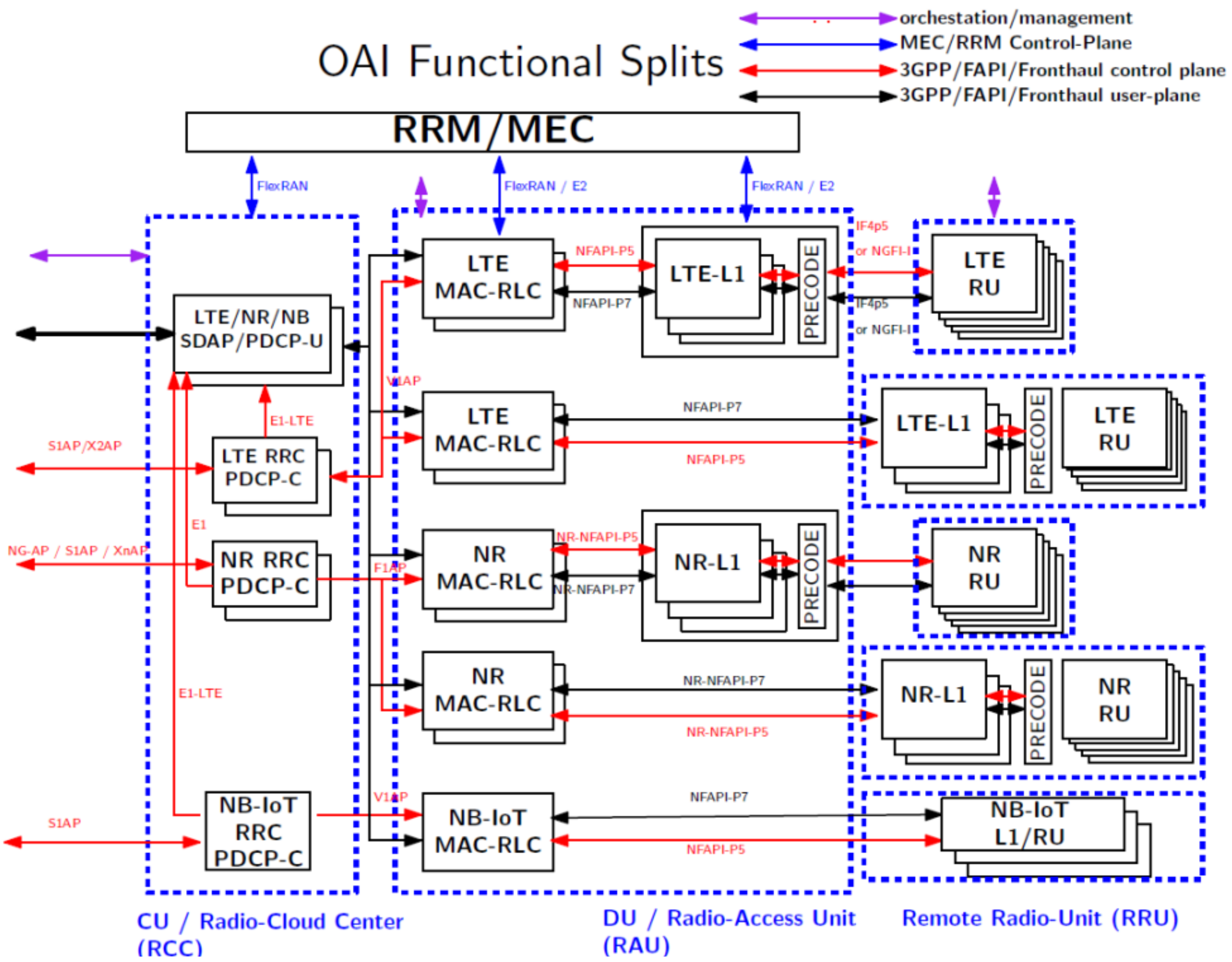


Figure 15: openairinterface5g functional decomposition

2.2.3 srsLTE

srsLTE is a free and open-source LTE software suite developed by SRS (www.softwareradiosystems.com). It is distributed on github (https://github.com/srsLTE/srsLTE) but, as a product, it can be licensed commercially.

It includes:

- srsUE - a complete SDR LTE UE application featuring all layers from PHY to IP;
- srsENB - a complete SDR LTE eNodeB application;
- srsEPC - a light-weight LTE core network implementation with MME, HSS and S/P-GW;
- a highly modular set of common libraries for PHY, MAC, RLC, PDCP, RRC, NAS, S1AP and GW layers.

srsLTE is released under the AGPLv3 license and uses software from the OpenLTE project (http://sourceforge.net/projects/openlte) for some security functions and for NAS message parsing. Today it offers LTE Release 10 functionality and operates using various software-defined radio devices such as USRPs and LimeSDR.

2.2.4 O-RAN Amber

O-RAN Amber (https://wiki.o-ran-sc.org/display/TOC/Project+Readiness+for+Amber+Release) is a partially open-source software suite for RAN prototyping and proof-of-concept demonstration of the O-RAN interface specifications. It was released in December 2019 and currently contains a 4G implementation and some elements of the 5G Standalone protocol stack. The software includes both RAN implementations based on the O-RAN license (FRAND license) and closed binaries implementing physical layer procedures. It also includes initial implementations of the so-called RAN Intelligent controllers which are released under an Apache V2.0 license.

2.3 Mobile Edge and SDN Elements

2.3.1 Mosaic5g

Mosaic5G is a non-profit initiative that provides open, flexible and agile service platforms for rapid prototyping and experimentations in 4G/5G and beyond. It aims to share an ecosystem of open-source platforms and use cases for 4G/5G system research leveraging software-defined networking (SDN), network function virtualization (NFV), cloud native, and multi-access edge computing (MEC) technology enablers. Mosaic5G spans across six software components (c.f. http://mosaic-5g.io/projects/) as shown in the Figure 16 below:

1. **JoX** as a 4G/5G service management and orchestrator;
2. **Store** as a repository of reusable network functions, applications and datasets;
3. **LL-MEC** as a low-latency edge and core network (CN) controller;
4. **FlexRAN** as a real-time radio access network (RAN) controller;
5. **OpenAirInterface** RAN and CN software packages and cloud images;
6. **Kube5G** as a rapidly deployable 4G/5G integrated platforms for cloud-native and AI/ML experimentations and prototyping.

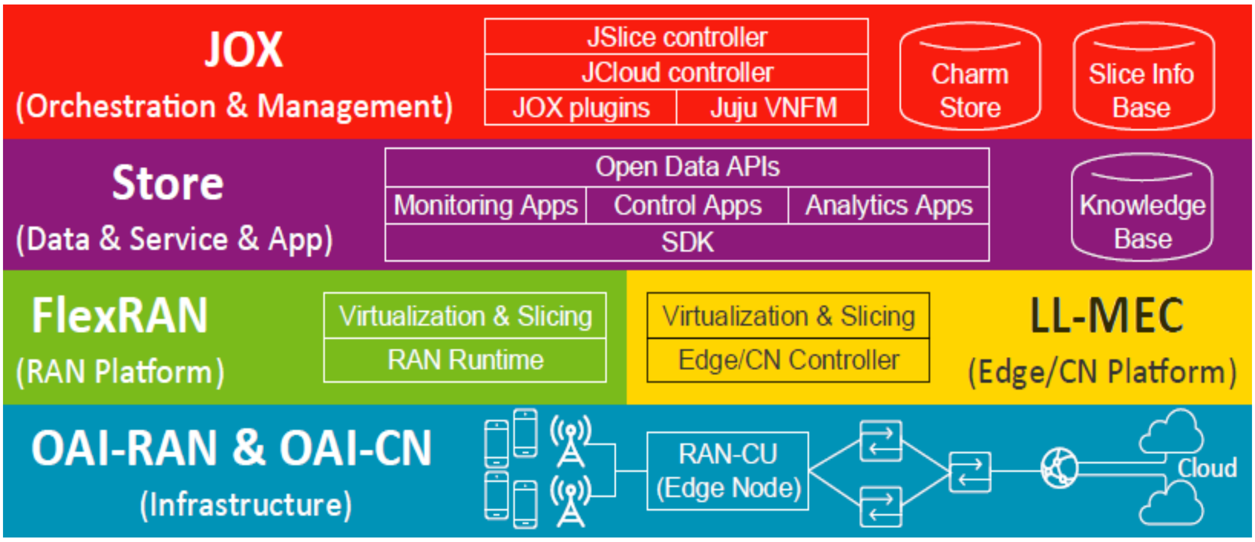


Figure 16 - Mosaic5G

2.3.2 AKRAINO

Launched in 2018, Akraino Edge Stack aims to create an open source software stack that supports high-availability cloud services optimized for edge computing systems and applications. The Akraino Edge Stack is designed to improve the state of edge cloud infrastructure for enterprise edge, OTT edge, and carrier edge networks. It will offer users new levels of flexibility to scale edge cloud services quickly, to maximize the applications and functions supported at the edge, and to help ensure the reliability of systems that must be up at all times.O-RAN (RIC NRT-RIC)

Currently, the chosen operating system (OS) is Ubuntu 16.04 and/or 18.04. The infrastructure orchestration of IEC is based on Kubernetes, which is a production-grade container orchestration with rich running eco-system. The current container network interface (CNI) solution chosen for Kubernetes is project Calico, which is a high performance, scalable, policy enabled and widely used container networking solution with rather easy installation and arm64 support. In the future, Contiv/VPP or OVN-Kubernetes would also be candidates for Kubernetes networking.

- The high-level design of IEC architecture uses containerized networking environment;
- The edge applications run as containers with container orchestration engine and high-performance networking support;
- The integrated edge cloud platform provides management interface and programming interface to deploy/manage edge applications quickly and conveniently;
- The platform supports the applications of IoT gateway, SD-WAN, edge AI and etc.;
- Under the current architecture, 2 kinds of use cases are supported;
- Telco/enterprise Edge cloud – for example, MEC or branch office data centre, etc.;
- Telco/enterprise remote edge locations – edge platform with limited resources, for example, SD-WAN, IoT gateway, etc.

The IEC reference stack architecture is given with the following figure:

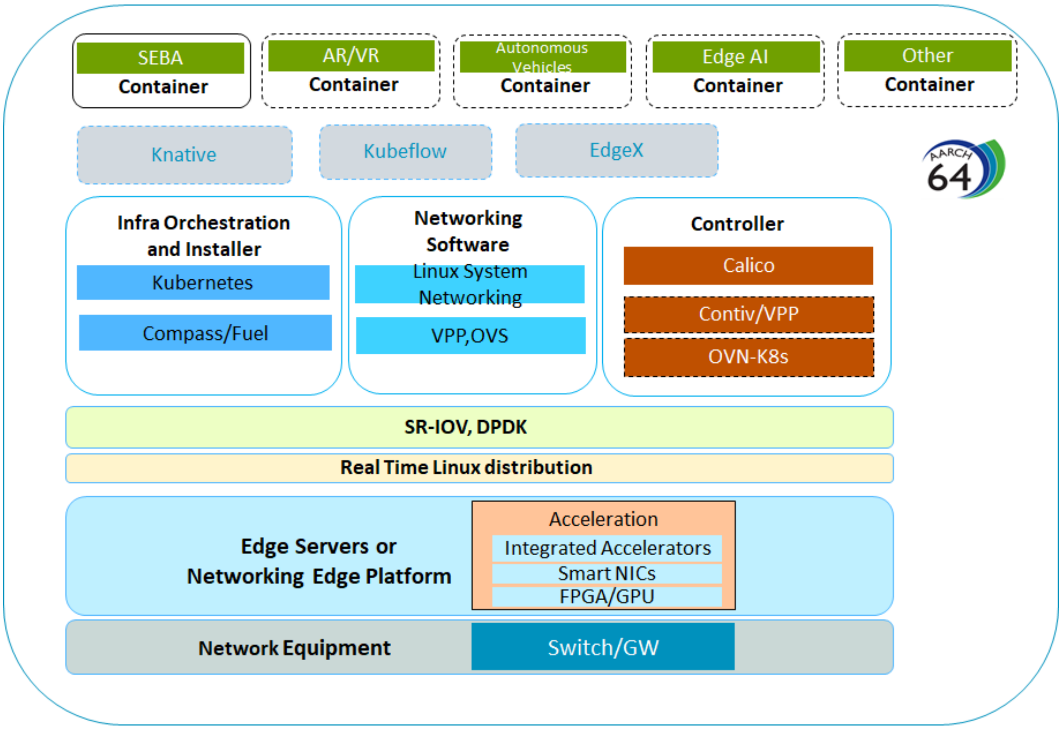


Figure 17: IEC Reference stack architecture

Currently, IEC provides the following functions under its reference architecture:

- The IEC supported hardware are edge servers mainly based on arm64, such as Huawei Taishan, Marvell ThunderX, Ampere Arm64 servers; at the far edge, the supported edge end devices would be Marvell MACCHIATObin Double Shot or other arm based boxes/devices. The desired network connections are above 10Gbit/s which may satisfy most current IEC applications requirement.
- The installation scripts which deploys Kubernetes cluster, Calico CNI, Helm/Tiller and related verifying Kubernetes applications/services with 1 master and 2 slave nodes. The scripts can be run from the jumpserver, or with manual installation from the servers on which it run. The installation methods is introduced in IEC Blueprints Installation Overview.
- Currently IEC uses project Calico as the main container networking solution which provides high performance, rich network policy, widely supported from Linux system and easy installation. In the future, Contiv/VPP and OVN-Kubernetes can be used as a high performance substitute since those 2 solutions can support DPDK enabled high speed interface access.
- IEC support Akraino CI/CD requests: IEC Daily jobs (scheduled to run recurrently) deploy IEC using one of the agreed installers; run testing suites; collect logs and publish them.

- Currently IEC suppors the SDN Enabled Broadband Access(SEBA) as its first use case. The installation scripts for SEBA on arm and its related source repositories are developed and/or integrated in IEC source code repo. We had ported SEBA components to arm64 servers with Helm chart installation support
- Until now IEC has 3 approved types: Type 1, Type 2 and Type 3 as its supported running types, and other types: Type 3 and Type 5 are under review. IEC is still enriching its use cases with the progress of developing.

The IEC reference software platform architecture is given with the following figure:

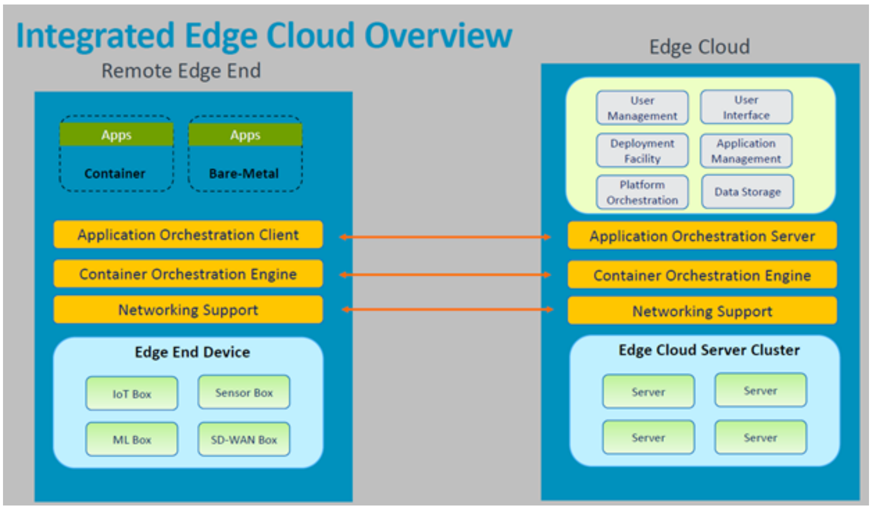


Figure 18: Akraino Software Architecture

3. Experimental Networks for EU-US Collaboration

3.1 Open Experimental Sites in ICT-17 5G-EVE

3.1.1 Open5GLab (EURECOM/RedHat)

Open5GLab at EURECOM is one of 3 experimental 5G sites in France in the context of the 5G-EVE ICT-17 project. Construction began in July 2018 and 5G experimentation is now available. The site is interconnected with similar sites in Europe in the 5G-EVE network. It is also one of the test sites for the OPNFV VCO 3.0 (Virtual Central Office) project and as such is interconnected with sites in North Carolina, USA and Montreal, Canada.

Open5gLab provides experimental 5G services including so-called Enhanced Mobile Broadband (eMBB) and massive machine-type communications and is based on fully open-source tools and open-architecture design. It is the main experimental playground for OpenAirInterface (OAI) and Mosaic-5g (M5G) software packages. The site's cluster computing resource makes use of RedHat's OpenShift 4.2 Kubernetes container platform and benefits from technical support from RedHat. The cluster is used for radio-access, core network and mobile-edge functions shown at a high-level in Figure 19. The cluster's switching fabric currently make use of three Edgecore 7312-54XS Data Center switches running CumulusOS.

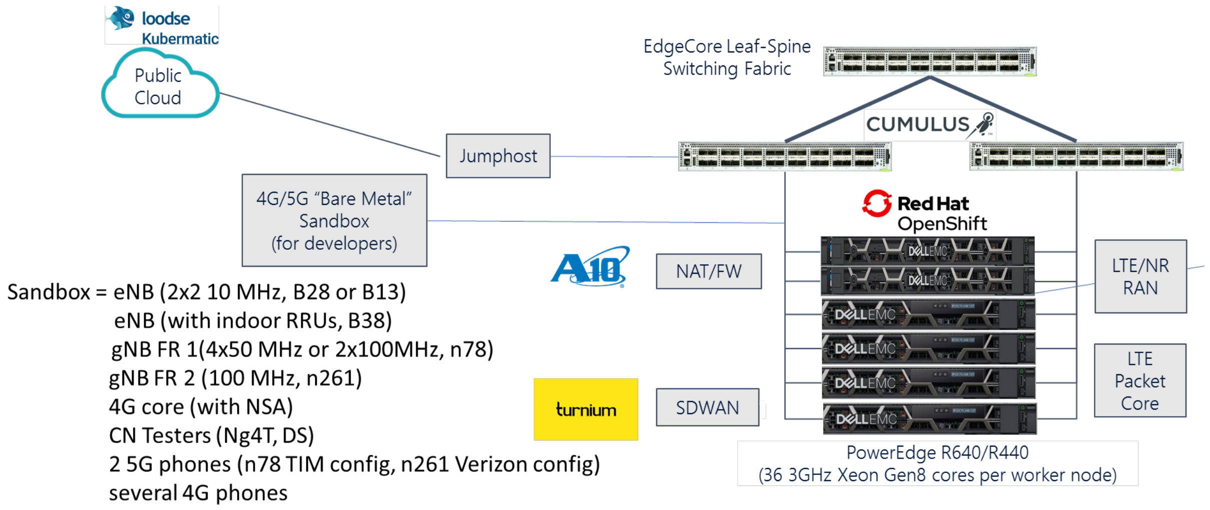


Figure 19: Current Open5GLab computing and testing elements

Some bare-metal nodes with in-lab 5G-capable radio devices (FR1 and FR2) are available as a sandbox that can be used by experimenters and developers and are interconnected with the Kubernetes cluster. External access for onboarding software, collecting measurement data and developing basic software for the site is available for partners using secure-shell access. Interfaces for an external orchestrator (e.g. ONAP) is currently being integrated in the Open5GLab. The nodes of the site are also used by OpenAirInterface Jenkins-based continuous integration / continuous delivery (CI/CD) framework.

Open5GLab's radio infrastructure includes indoor and high-power outdoor radio-units operating in several 4G and 5G bands in the immediate vicinity of the test site, specifically Band 28 (700 MHz), Band n38 (2.6 GHz TDD), Band n78 (3.5 GHz TDD), Band n258 (25 GHz TDD). The outdoor units are interconnected with the switching fabric using 300m fiber (10/25 Gbit/s). The units are a combination of in-house designs and commercial remote radio-units. The outdoor installation is shown in Figure 20.

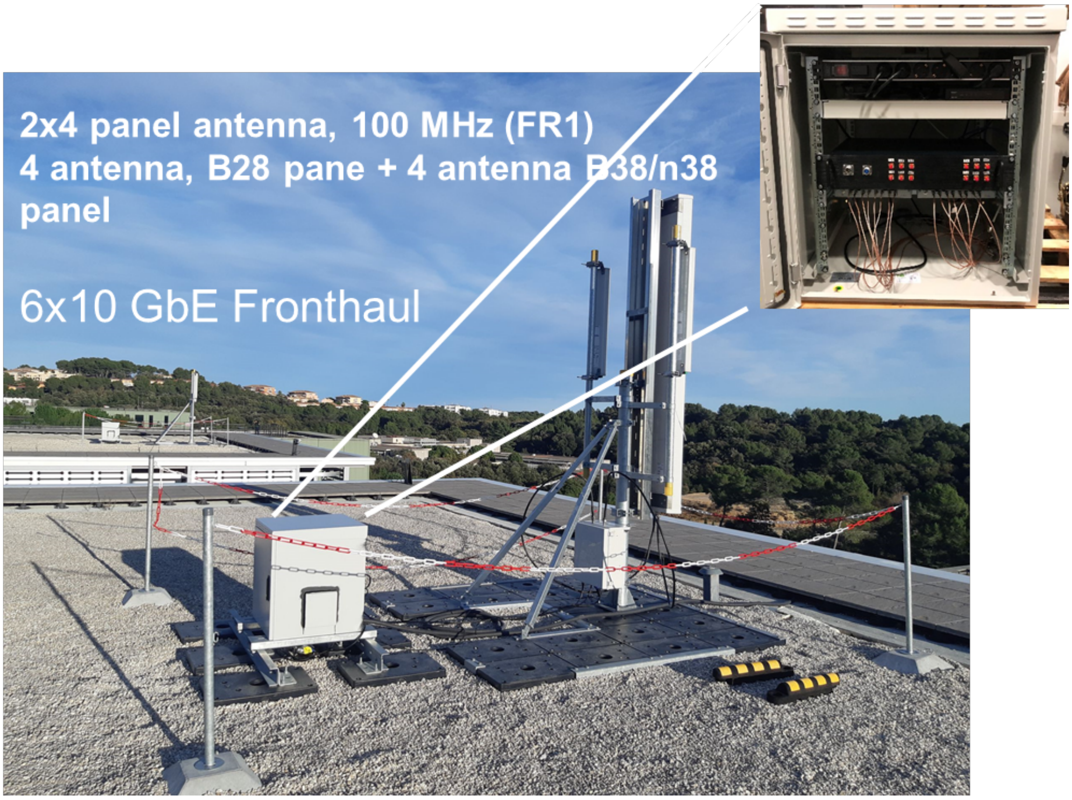


Figure 20: Open5GLab outdoor radio equipment

Open5GLab provides remotely-controllable 4G and 5G user-equipment, including both off-the-shelf smartphones and cellular IoT modules. This allows experimenters to control and extract measurements from the user-equipment in a running experiment. With the help of regional partners, Open5GLab will provide embedded vehicular user-equipment nodes in 2020. Two drones are also equipped with 4G and soon 5G user-equipment and OAI-based 4G radios for mobile basestation experiments. With the help of EURECOM, software can be on-boarded into the user-equipment devices.

The deployment framework for openair5gLab is full open-source and distributed on the openair-k8s github (https://github.com/OPENAIRINTERFACE/openair-k8s). openair-k8s allows building high-quality OCI-compliant container images for the OpenAirInterface 4G/5G radio access (eNB/gNB) and core networks (EPC) and deploying these components on OpenShift or other enterprise-grade Kubernetes distributions. More recently VMWARE has provided support for vanilla kubernetes on freely available CentOS and Ubuntu distributions. The RAN and Core network networking configuration provided by openair-k8s is shown in Figure 21. Networking functions are split between hard real-time functions and soft real-time functions and execute on specific worker nodes in cluster configured accordingly. The workers themselves are Kubernetes Pods and make use of the Multus container network interface (CNI) plugin for Kubernetes. This allows for all types of networking interconnections control/user plane separation as well as fronthaul and midhaul for the hard real-time components.

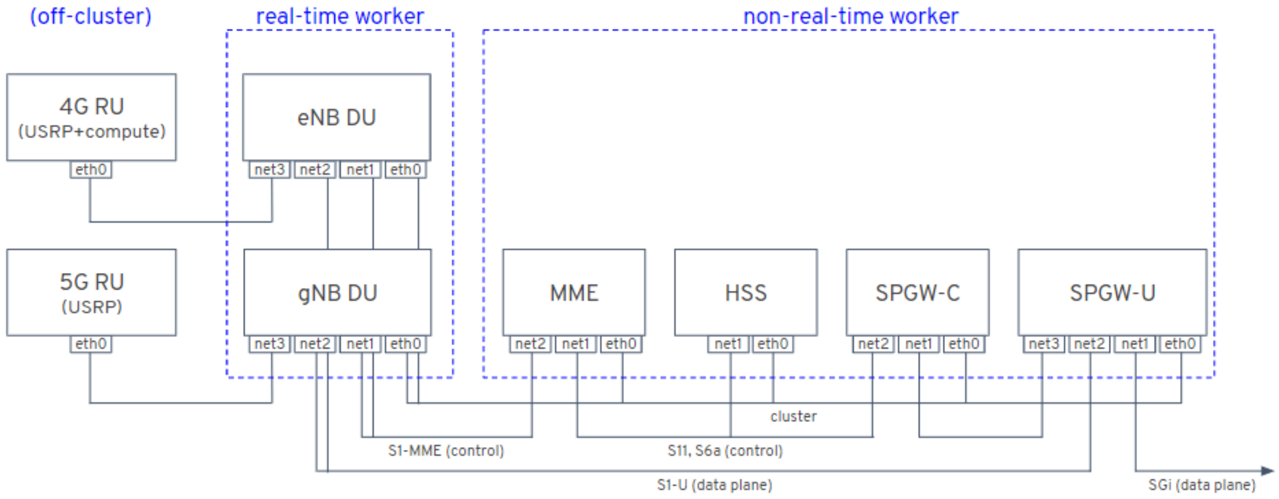


Figure 21: openair-k8s networking

3.1.2 Plug'In (Orange)

The Plug'in Plartform is provided by Orange.
In terms of hardware and software resources, Plug'in is running on multiple OpenStack virtual machines and 3 bare-metal servers.

In terms of software development and integration frameworks, some building blocks of the Orange 5G research platform Plug'in are available. As shown in Figure 22, the platform offers an integrated set of tools that can be seen as an Integrated Development Environment (IDE) for continuous: Learn Develop, Document, Experiment, Publish, and Reuse.

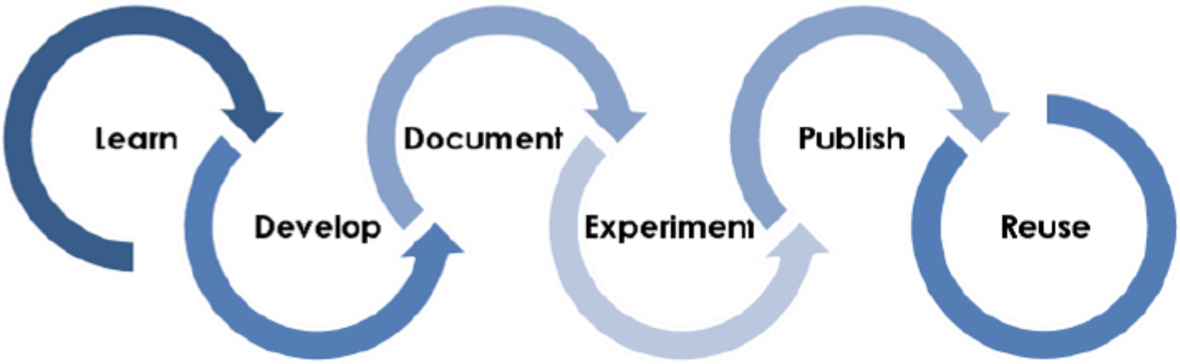


Figure 22: Plug'in Platform Cycle

AtomStore: this tool is the entry point to the IDE. It's the place where users and developers can find the list of all atoms, classified in categories for a better user experience. Each atom is presented with a visual illustration and brief description to understand at a glance its basic idea. Additional information is provided using links to redirect the user to the Atom's documentation or its home page. Some metadata on the Author of the Atom, the creation date, version, or rate are also provided. The AtomStore is not only a Graphical User Interface (GUI) but also an API service. Currently, the REST API provides basic operation to manage Atom information for example.

AtomDocs: this tool provides a cleaned view of the documentation of each Atom. The workflow of generating an Atom's documentation is automated and puts together: a file server, a pre-processing engine, a conversion engine and finally a web server. It also integrates team collaboration tools such as "slack" or "mattermost", to notify the developer when the documentation is updated. The idea of AtomDocs is to allow developers to document their software using their development environment, always deliver the latest version of the documentation, and never worry about the HTML formatting and rendering of the documentation which helps them focus on their developments.

AtomGen: as stated before, in order to provide a homogeneous development environment, a tool called “AtomGen”, which is a project skeleton generator, has been defined. The idea of this tool came from the open source world, where different sub-modules of the same project are not always organized or even coded with the same “style”. This introduces an additional effort to explore and understand the code in the project. To tackle this issue, “AtomGen” provides a standardized structure of all Atoms where users and developers can quickly locate different information.

Toolbox: the concept of toolboxes is about providing a starter toolkit for multiple domains that researchers can use to quickly get involved in those domains. For example, a SDN toolbox may contain an SDN controller such as OpenDayLight or ONOS, a network emulator such as Mininet to create virtual switches, and a set of network topologies, benchmarks, and/or deployment scripts to help a first year PhD to get a ready-to-use SDN environment. Multiple SDN toolboxes may be provided, each with a different set of tools or use-cases.

Formulas: this tool is HTML GUI and a REST API to provide a set of formulas that can be deployed on the PlayGround with a single button click. A Formula is basically a docker stack that deploys unitary software. For example, it is possible to create a Formula for Kibana, Jupiter notebook, Prometheus, etc.

Cloud Native SDK: Since the NFV world is accelerating towards Cloud Native Computing design, a Cloud Native SDK is provided to help researcher to adopt the design and to think cloud native from early development stages of their VNFs. The SDK provides a VNF life cycle management REST API. This API allows most of the management operations such as configure, start, stop, health check and status. This work is in progress and next version may integrate scaling out option for example. At the beginning, this API was developer to wrap the OpenAirInterface eNodeB and Core Network but it evolved to a more generic development kit.

PlayGround: this tool is one of the corner stones of the platform. It offers computing, networking and storage resources to researchers to test and experiment not only on Atoms, but basically any software. The “PlayGround” provides ephemeral computing sessions in which Alpine is instantiated, Ubuntu or CentOS – based on computing instances. The idea of the “PlayGround” is to enhance reproducible research mindset and share resources.

Wall: the “wall” is a publishing tool that helps record experiment provenance, publish results, share experience, write notes, publish tutorials, and more. It is equivalent to a social network for the platform’s community. It provides a HTML GUI and a REST API through which posts are managed.

AtomInsight: this tool is designed for learning purpose. Using a HTML GUI interface and the “PlayGround” SDK, the “AtomInsight” provides a set of tutorials to help developers easily learn about new technologies. The “PlayGround” SDK is used to allocate computing resources from the “PlayGround” to execute a given tutorial. A stable version of this tool is available. In future releases, the “AtomInsight” will be able to be automatically populated by a given Atom’s tutorials that are written by the Atom’s developer himself.

Figure 23 illustrates the different tools of the Plug’in platform:

Figure 23: Overview of the Platform's tools and the underlying technologies

In terms of monitoring tools, Plug'in platform currently use the following open source tools: **Prometheus**, PCP and Vector. Prometheus is used to monitor multiple metrics of the PlayGround such as the number of sessions, and the number of created compute instances, the garbage collection cycles, and more. The Performance Companion (PCP) and Vector are used in common to monitor our bare-metal servers: CPU and memory usage, load average, networking, I/O to name a few. This combination could be used to supervise all sorts of computes with metrics ranging from system to containers. Other tools could be considered, in no order of importance, among them:

The TICK suite from influxdata: Telegraf (a metric collector or probe), InfluxDB (timeseries database), Chronograf (a visualization dashboard), and Kapacitor (a data processing engine for alerting) [59];

Cadvisor (metrics probe) with Grafana (dashboard);

Cadvisor with an elastic stack (Elastic search for metrics and Kibana for dashboard);

Statsd (metrics probe) and Graphite (time series database for metrics).

Continous Integration: The Plug'in platform tools leverage multiple DevOps tools for Continuous Integration/Continuous Delivery (CI/CD). For example, Gitlab is used to host all the code of the platform tools, Atoms, Toolboxes, and Formulas. Gitlab-ci and Jenkins are also used to automate the documentation generation as illustrated in the Figure 24 below.

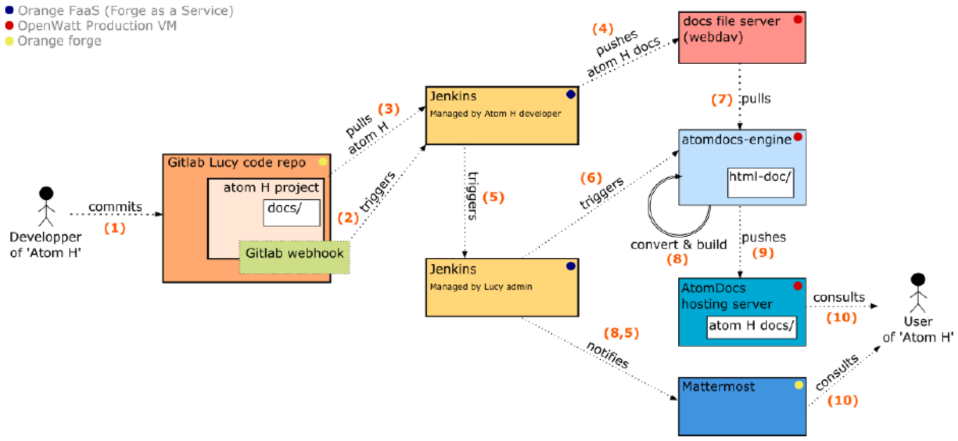


Figure 24: CI CD Workflow of the AtomDocs

Table 2: Plug'in DevOps Tools

Properties	Tools	Class
Version control	git	Publish
Code hosting	Gitlab, Github	Publish
CI/CD	Gitlab-CI, Jenkins-CI, Travis-CI	Schedule
Infra. deployment	Ansible, Docker Compose, Vagrant	Deploy
Artifacts hosting	JFrog Artifactory, local deb/rpm repos	Publish
Communication and Collaboration	Attlassian Jira, Mattermost, Slack	Collaborate
Logging	Elastic Stack	Trace
Monitoring	Prometheus, Netflix Victor	Measure

3.1.3 Wireless Edge Factory (B-COM)

3.1.3.1 Overview

BCOM Wireless Edge Factory is an SDN based private network framework enabling end to end broadband, IoT and WebRTC critical communications to be carried out in a fully secure manner in small to medium size buildings or industrial sites.

Current specifications are:

- SDN (OpenFlow v1.3, OpenDaylight controller);
- Preloaded with Full LTE EPC (MME, S/P-GW, HSS) based On OpenAirInterface CN;
- WLAN 802.1x protocols;
- Preloaded with LoRa Core Servers (NS, AS, CS);
- EAP-AKA, EPS-AKA SIM based authentication mechanisms.

New releases of **Wireless Edge Factory** are regularly issued by b<>com Research Institute of Technology and is moving to a 5G SBA implementation. The actual VNF version of the **Wireless Edge Factory** proposes the distributed multi-site architecture depicted in the figure below:

3.1.3.2 Infrastructure

The NFV Infrastructure in b<>com site to host the *Wireless Edge Factory* VNF and future experimentation VxFs is composed of Dell PowerEdge R640 servers with the characteristics described in Section 5.6.2 of this document. A Ceph cluster provides 12 TB of distributed storage connected to the OpenStack VIM with a 2x10 Gbps aggregated link. Internal VIM datapath is 2 x 10Gbps aggregated link, while the external VIM connectivity is 1 Gbps.

3.1.4 5TONIC

The following information is a summary of the facility description and roadmap provided in 5G-EVE deliverable D2.6.

The Spanish site, 5TONIC, has some special characteristics in the sense that it pre-existed the launch of the 5G EVE project and has among its members companies that are not part of the project consortium. In this sense, the laboratory supports the implementation of use-cases both from other EU projects, like 5G Vinni or 5Genesis, and also for companies that are not involved in EU funded projects.

3.1.4.1 Architecture

The 5TONIC site architecture can be described as composed by several layers that encompass a set of functionalities that allow the implementation of different use-cases for 5G. These layers are:

3.1.4.1.1 Infrastructure layer

This layer incorporates the network elements/functions that provide the connectivity required for the implementation the different use-cases in a self-contained way, i.e., not requiring the support of external elements.

There are several components of this layer that is continuously evolving in order to incorporate the latest developments of the 5G standard:

- Radio access, which provides 4G coverage and 5G NSA coverage and support, LTE, NR NSA and NB-IoT access in different frequency bands. 5G SA support is expected to be supported during the first quarter of 2020.
- Packet core network, which in the current implementation supports 4G virtualized EPC and 5G NSA virtualized core, ready to evolve to 5GC SA. The core also implements the elements required for the support of different policies and the management of the users' data.
- Transport network, that implements or emulates different transport alternatives for fronthaul and backhaul.

3.1.4.1.2 Service and application layer

This layer incorporates the network elements/functions that provide the capabilities to implement different use-cases on top of the infrastructure layer. For these purposes, the site provides processing capabilities that can be deployed either at the edge of the network or in the cloud. It also provides the mechanisms to measure and evaluate different indicators in order to characterize the services and applications performance in different operational conditions.

It should be noticed that in some cases elements from the infrastructure and service layers share the same hardware platform.

3.1.4.1.3 Control and management layer

The main function of this layer is to support the flexible configuration of the infrastructure and the service layers in order to support the requirements of different layers. In this sense, the layer is based on the use of cloud, software, and virtualization techniques, including the support of container-based deployment of VNFs. It also incorporates architectural elements from SDN in order to manage the 5TONIC transport network.

The orchestration platform, that constitutes the core of this layer, is currently based in ETSI OSM, with different options as VIM (OpenStack, SONA).

3.1.4.1.4 Interaction and security layer

This layer is in charge of facilitating the interaction with potential user, as well as other sites, in a secure way. This layer will evolve to incorporate the 5G EVE developed solutions.

The security layer is also in charge of guaranteeing the privacy requirements between the parties that may be involved in the support of different use cases, as well as preventing any intrusion or attack from outside.

3.1.4.2 Main features

3.1.4.2.1 Infrastructure

The infrastructure deployed at the 5TONIC site is represented in the following Figure 25:

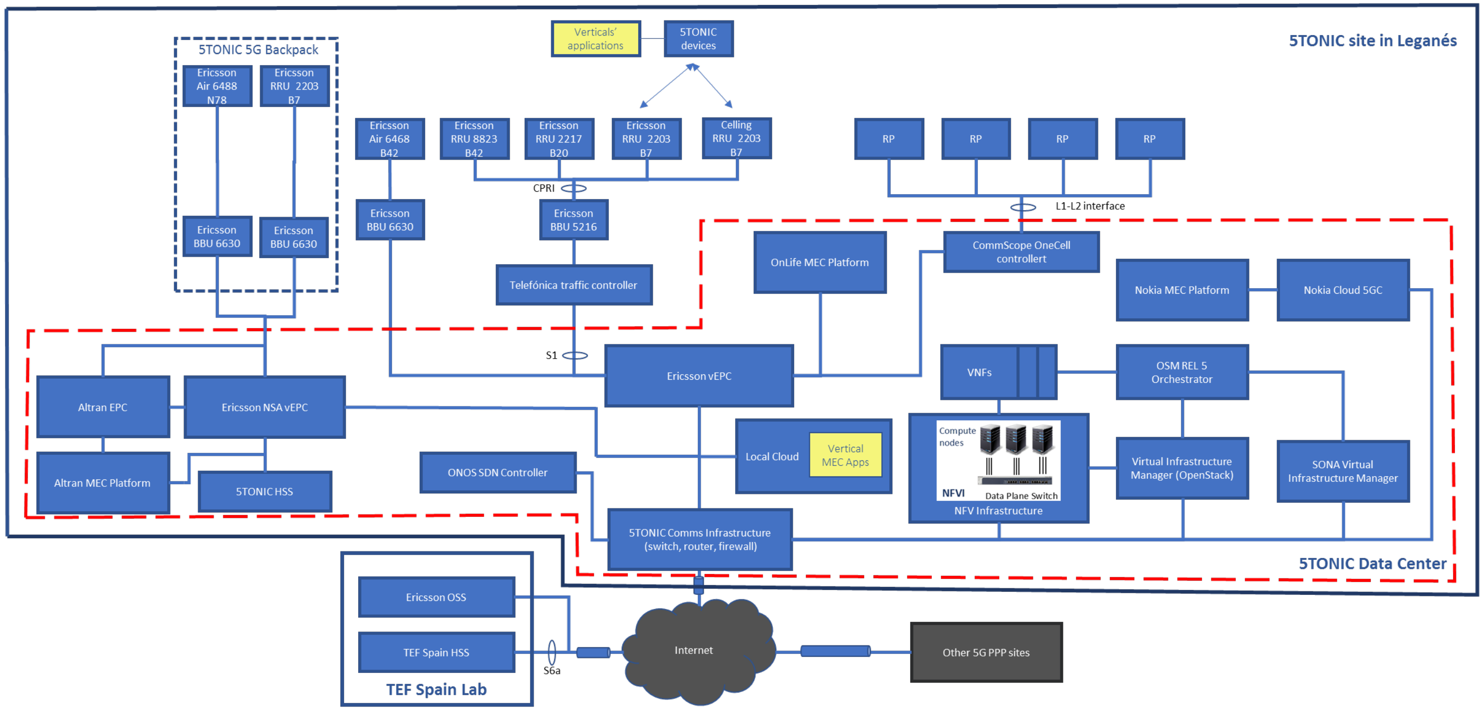


Figure 25: Spanish site facility deployed infrastructure

Current 5TONIC radio infrastructure incorporates the following elements:

- 4 baseband processing units (BBUs) from Ericsson, 3 of them 6630 (that support NR processing) and 1 5216 (that currently does not support NR processing);
- 3 Ericsson RRU 2203 that support LTE in band 7;
- 1 Ericsson RRU 2217 that supports LTE in band 20;
- 1 Ericsson RRU 8823 that supports TD-LTE in band 42;
- 2 active antenna systems, both with 64 TRX for supporting multiuser MIMO, 1 Air 6468 that supports TD-LTE in band 42 and 1 Air 6488 that supports 5G NR in band 43/n78.

Several of these elements are expected to be software upgraded (e.g., RRU 8823) to support NR radio interface. New 5G RAN elements are expected to be deployed in the laboratory during 2020.

To support the radio connectivity, the lab has acquired several devices that include:

- 4G smartphones, including 3 that support band 42 and also 256QAM
- 4G LTE routers, including one that supports 256QAM
- 5G NSA smartphones from different vendors (LG, Samsung, Xiaomi, Huawei)
- Raspberry Pi boards with LTE NB-IoT HATs.

In terms of core infrastructure, 5TONIC implements two virtual EPC from Ericsson, one of them just supporting 4G access and the second one supporting 5G NSA access. The first one is deployed on a Dell commercial server, while the second is based on Ericsson Hyperscale Datacenter System 8000.

5GC supporting SA connectivity is scheduled to be deployed by the end of Q1, 2020.

The transport infrastructure of the site includes 4 Ericsson routers 6675, as well as several HP SDN enabled switches.

For supporting the control and interaction layers a set of servers have been deployed at the site.

3.1.4.2.2 Services

Hitherto, the services provided by the site have been adapted to the specific requirements of the different use-cases that have been deployed at 5TONIC. In this sense, there are several options that have been supported (see Figure 25)

- Provision of basic 4G and/or 5G connectivity, e.g., to test the impact of 5G connectivity on a UE application performance. Because of the development of a portable 5G backpack, connectivity;
- Provision of performance parameters measurement capabilities, e.g., to evaluate KPIs like throughput, latency or reliability;
- Provision of network configuration capabilities in order to define the test's topology as well as other network parameters;
- Provision of service and applications support capabilities, so the user of the site services can rely of the lab processing infrastructure for supporting the implementation of the use cases;
- Provision of adequate space and facilities for deploying its own infrastructure by the site user;
- Maintenance services of the infrastructure deployed by the user;
- Provision of security services.

3.1.4.2.3 Test environment

5TONIC premises have several environments to carry out different kinds of experiments, including both indoor (in two different rooms) and outdoor coverage at IMDEA Networks premises. Also there is an open area that allows the movement of small vehicle.

5TONIC also has carried out tests on other premises, like Telefónica headquarters in Distrito C, Madrid, and the campus of the UC3M in Leganés.

UC3M has also a location in Madrid city center where it is feasible to carry out tests in dense urban areas.

Also, thanks to the implementation of the 5G backpack, it will be feasible to bring end-to-end 5G connectivity to other areas for testing purposes.

The use of the spectrum required for these tests has been granted by Telefónica.

3.2 Open Experimental Sites in ICT-17 5G-VINNI

The following information is a summary of the 5G-VINNI public deliverable D2.1.

5G-VINNI consists of in total 8 facilities. A 5G-VINNI Facility is the deployment of the 5G-VINNI architecture in one administrative domain (e.g. one operator). The 5G-VINNI Facility-sites are classified into two different types:

- *Main Facility-sites*: E2E 5G-VINNI Facility that offers services to ICT-18-19-22 projects with well-defined Service Level Agreements.
- *Experimentation Facility-sites:* 5G-VINNI sites that provide environments for advanced focused experimentation and testing possibilities on elements and combinations of elements of the E2E model.

The 5G-VINNI Facility-sites are illustrated in Figure 26 with the Main Facility-sites (**Norway, UK, Spain, Greece**) and the Experimentation Facility-sites (**Portugal, Germany/Munich, Germany/Berlin, Luxembourg**).

Figure 26: 5G-VINNI Facilities

5G-VINNI facilities are deployed and evolved in Releases as shown in the global timing diagram in Figure 27.

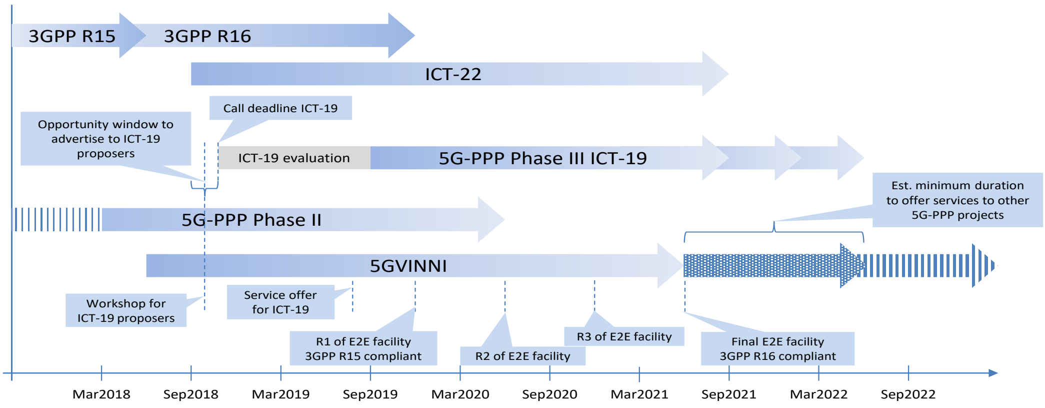


Figure 27 5G-VINNI Global Timing

3.2.1 5G-VINNI Facility-sites Actors

5G Network infrastructure for different Facility-sites is provided by different vendors. Table below includes supplies of individual 5G-VINNI Facility-sites.

Table 3: Facility-site actors

Facility-site	*Norway*	*UK*	*Spain*	*Greece*	*Portugal*	*Germany (Berlin)*	*Germany (Munich)*	*Luxemburg*
Operational partner	Telenor	BT	TID	University of Patras	Altice Labs	Fraunhofer FOKUS	Huawei Technologies Dusseldorf GmbH	SES
5G RAN Vendor(s)	Ericsson, Huawei	Samsung	Ericsson, Limemicro, SRS, OAI	Limemicro, SRS	OpenAirInterface Software Alliance	Under NDA	Huawei	N/A
5G EPC/5G Core Vendor(s)	Ericsson	Samsung	Ericsson, OAI, Fraunhofer FOKUS (under consideration)	Fraunhofer FOKUS	Fraunhofer FOKUS	Fraunhofer FOKUS	No EPC, Huawei 5G NFs	Fraunhofer FOKUS
NFVI	Nokia	Samsung	OpenStack	OpenStack	OpenStack	OpenStack	Huawei	Open-Source (OpenStack based)
MANO	Nokia	Samsung	OSM	OSM	SONATA	Open Baton - Fraunhofer FOKUS	Huawei	Open Baton - Fraunhofer FOKUS
Service Orchestration	Nokia	Nokia	5GinFIRE portal	5GinFIRE portal	ONAP	Fraunhofer FOKUS	Huawei	Fraunhofer FOKUS
Transport Network	Telenor	BT	TID (optical mesh)	N/A	N/A	N/A	N/A	SES, VT iDirect

3.2.2 *Facility-sites Deliverables*

The technical services offered by 5G-VINNI refer to the resource facing services (RFS) offered by 5G-VINNI Facility. Customer facing services will be agreed and evolved according to the requirements of the verticals/ICT-19 projects. To exemplify, mobile-broad-band-on-demand can be considered as a customer facing service which is composed of eMBB slice which is an RFS. The eMBB slice RFS in turn provides flexible backhaul RFS, autonomous edge RFS leveraging the resources in the access network, transport network and core network.

Table 4: List of technical services from 5G-VINNI which can be exposed to ICT-19 or verticals

No.	*Technical Services offered in Release 1 or beyond Release 1*	*Norway*	*UK*	*Spain*	*Greece*	*Portugal*	*Germany (Berlin)*	*Germany (Munich)*	*Luxemburg*
1	eMBB slice	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, beyond Rel-1	YES, in Rel-1
2	URLLC slice	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, beyond Rel-1	YES, in Rel-1	YES, in Rel-1	NO
3	mMTC slice	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, beyond Rel-1	YES, in Rel-1	NO	YES, in Rel-1
4	Autonomous core in the edge / Self-contained network (a)	YES, beyond Rel-1	NO	NO	NO	NO	YES, beyond Rel-1	NO	YES, beyond Rel-1
5	Fixed wireless access	YES, in Rel-1	YES, in Rel-0	YES, in Rel-1	YES, in Rel-1	NO	NO	NO	NO
6	Firewalling (Layer4-7)	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	NO	NO	NO
7	Flexible backhaul for redundancy (say via satellite) (b)	YES, beyond Rel-1	NO	NO	NO	NO	YES, in Rel-1	NO	YES, in Rel-1
8	Interconnection with Public cloud (c)	YES, beyond Rel-1	NO	NO	YES, Beyond Rel-1	YES, in Rel-1	NO	NO	NO
9	Data fabric service involving correlation, aggregation and analytics (d)	YES, beyond Rel-1	YES, in Rel-1	YES, in Rel-1	NO	Yes, beyond Rel-1	NO	NO	NO
10	Test and KPI validation	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	YES, in Rel-1	NO	NO	NO

No.	*Technical Services offered in Release 1 or beyond Release 1*	*Norway*	*UK*	*Spain*	*Greece*	*Portugal*	*Germany (Berlin)*	*Germany (Munich)*	*Luxemburg*
11	3rd party VNF hosting	YES, beyond Rel-1	YES, in Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	NO	NO	NO
12	Edge cloud	YES, beyond Rel-1	NO	YES, beyond Rel-1	YES, beyond Rel-1	YES, in Rel-1	NO	YES	NO
13	Interconnection with other 5G-VINNI Facility-sites	YES, beyond Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	YES, in Rel-1	NO	YES, in Rel-1
14	Interconnection with non-5G-VINNI Facility-sites (to be offered based on demand)	YES, beyond Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	NO	NO	NO
15	Individual device connectivity (both eMBB and IoT) to 5G-VINNI Facility via default slice	YES, in Rel-1	YES, in Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	YES, beyond Rel-1	NO	NO	NO

Description of the technical services:

(a) Autonomous core in the edge / Self-contained network: This service involves spinning up a mobile core (Both control and data plane) in the edge for example in case the backhaul connection is broken which is essentially a self-contained network

(b) Flexible backhaul for redundancy: This service involves providing redundancy in the backhaul for example via Satellite link.

(c) Interconnection with Public Cloud: The possibility of hosting the network functions in public cloud or extending the network slice in the public cloud.

(d) Data Fabric service: Service to extract, compute/transform and move data across the distributed network Facility (edge, fog, core).

3.2.3 Capabilities to be supported

Table 5 show from what Release specific capability is supported.

Table 5: Capabilities to be supported in 5G-VINNI facilities

No.	*Capability name*	*Norway*	*UK*	*Spain*	*Greece*	*Portugal*	*Germany (Berlin)*	*Germany (Munich)*	*Luxemburg*
1	5G New radio	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	No
2	Integrated low power wide area networks	Rel-1	Rel-1	Rel-1	Rel-1	No	No	No	No
3	5G-Core	Beyond Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1 non-3gpp compliant	Rel-1
4	Network slicing based on 5G-EPC	Rel-1	Rel-0 (maybe removed by Rel-1	Rel-1	N/A	No	No	No	No
5	Network slicing based on 5G-Core	Beyond Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Beyond Rel-1	Rel-1 non 3gp	Beyond Rel-1
6	NFVI	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1 In House	Rel-1
7	MANO	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1 In House	Rel-1
8	E2E Service Orchestration	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	Rel-1	NO	Rel-1
9	Edge computing	Beyond Rel-1	Rel-1	Beyond Rel-1	beyond Rel-1	Rel-1	Rel-1	Rel-1	Rel-1
10	MEC Compliant edge computing	-	-	Beyond Rel-1	beyond Rel-1 (ICT-19 depend)	Beyond Rel-1	Rel-1	No	Rel-1
11	On-boarding containerized workloads	Rel-1	Beyond Rel-1	Rel-1	Rel-1	Rel-1	No	Yes, for internal use	No
12	SD-WAN	-	-	No	-	No	Rel-1	No	Rel-1

No.	*Capability name*	*Norway*	*UK*	*Spain*	*Greece*	*Portugal*	*Germany (Berlin)*	*Germany (Munich)*	*Luxemburg*
13	Control user Plane separation (CUPS) architecture for maximum topology flexibility	Rel-1	Rel-1	Beyond Rel-1	Beyond Rel-1	Rel-1	Rel-1	Beyond Rel-1	Rel-1
14	Secure architecture with infrastructure zoning and with L4-7 Firewalling capabilities	Rel-1	Beyond Rel-1	Beyond Rel-1	Beyond Rel-1	Beyond Rel-1	No	No	No

3.2.4 *Cross-facility-sites end-to-end solutions*

Norway and UK facility sites plan to be interconnected beyond Release1. Inter-working between the UK Facility-site and the Norway Facility-site will be implemented at the E2E Service Operations layer in the 5G-VINNI reference architecture. This will involve delivering function support for the four API's defined for the INTERLUDE (SOF:SOF) Management Interface Reference Point in the reference architecture. These interfaces are:

- Service Ordering API
- Activation and Configuration API
- Service Inventory API
- Service Catalogue API

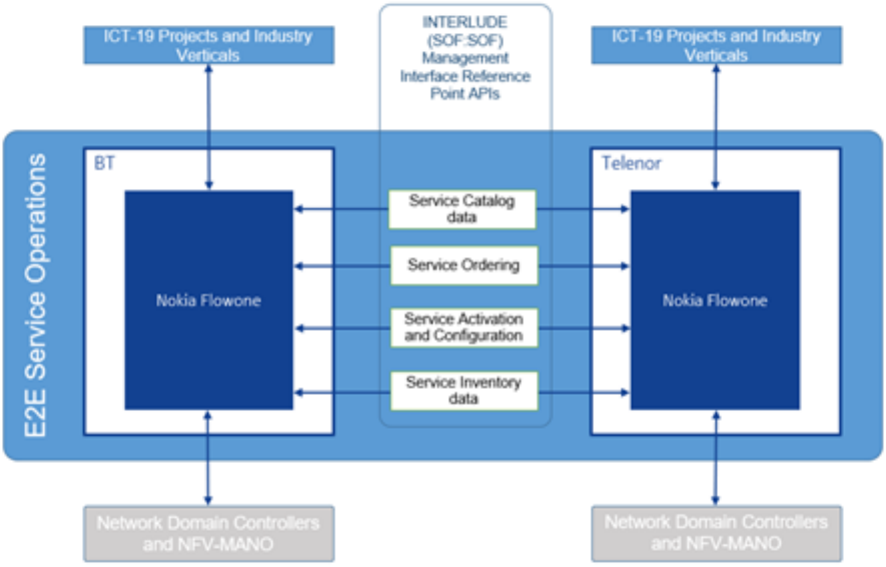


Figure 28: Cross Facility-sites Norway and UK

There will also be an interconnection between the Greek and Spain facilities provided by GEANT beyond Rel-1 using the expertise and infrastructure already deployed in 5TONIC by the three common partners (TID, UoP and UC3M) in both the 5GinFIRE and 5G-VINNI projects.

The potential interconnection towards the outside is mainly managed by the 5TONIC Communications Infrastructure module that encompasses a number of elements that allows the management of the internal communications in 5TONIC premises, the secure connectivity with Internet and other sites. The security infrastructure deployed allows for the implementation of VPNs with IPSec.

5TONIC infrastructure has been already used for the support of several multisite trials associated to H2020 projects. The following figure depicts the inter-site connectivity that supports remote orchestration. In this case, Site 1 was located in Portugal and Site 2 in UK.

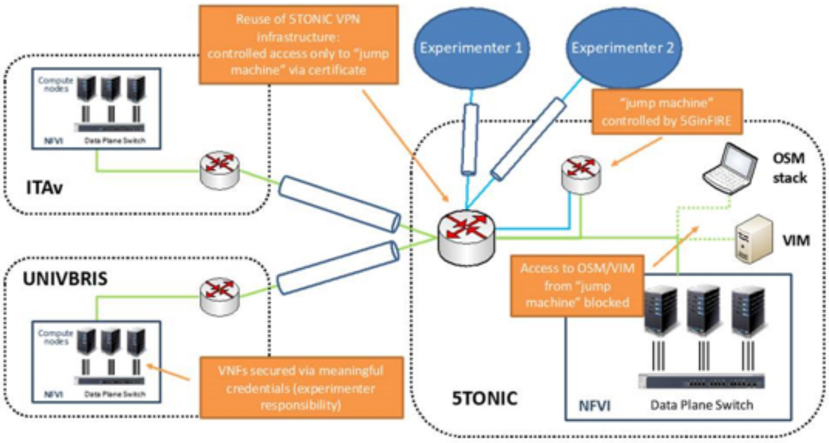


Figure 29: Intra-site connectivity in 5GinFIRE

The 5G-VINNI moving experimentation Facility-site will be satellite interconnected with the 5G-VINNI Berlin experimentation Facility-site (located at Fraunhofer FOKUS's premises in Berlin) and the SES's teleport Facility-site located in Betzdorf (Luxembourg) as depicted in Figure 30.

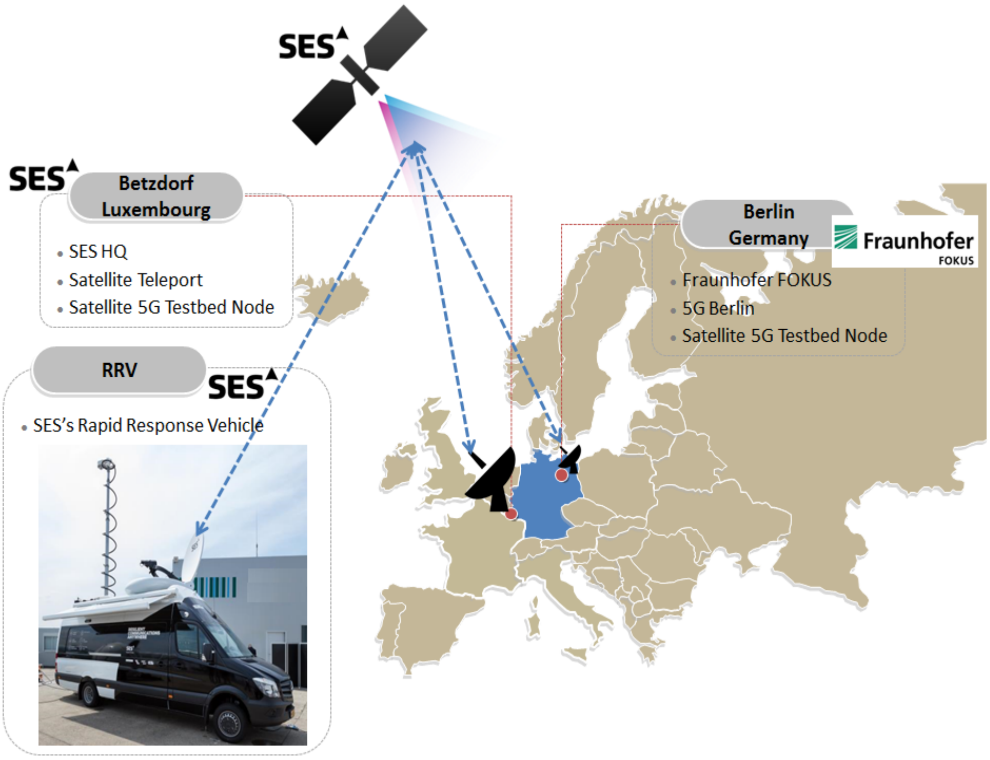


Figure 30: Satellite interconnection between 5G-VINNI moving experimentation Facility-site and 5G-VINNI Berlin experimentation Facility-site

3.2.5 *5G-VINNI Test-as-a-Service (TaaS) Concept*

There will be a corpus of test cases to be used as is or as templates for customization. A test portal will be available to allow for the creation and customization of individual test cases, management of test campaigns and results processing and analytics. The 5G-VINNI test framework also includes the possibility to integrate third party components and systems under test via open SDK, and consulting services for testing and experimentation.

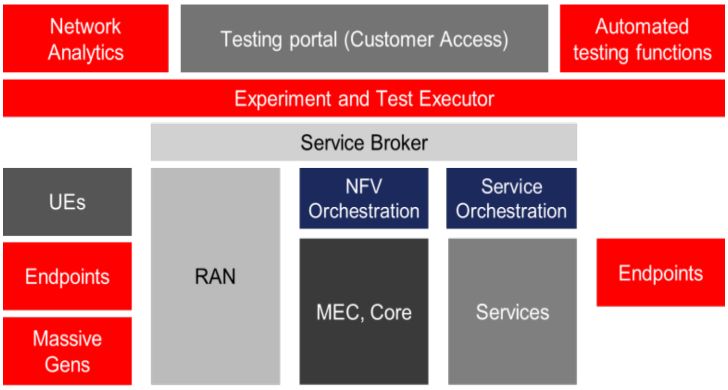


Figure 31: High level test architecture

3.3 Open Experimental Sites in ICT-17 5GENESIS

The main goal of the 5GENESIS will be to validate 5G KPIs for various 5G use cases, in both controlled set-ups and large-scale events. As such the following facilities are available;

1. The **Athens** Platform. An edge-computing-enabled shared radio infrastructure (gNBs and small cells), with different ranges and overlapping coverage that are supported by an SDN/NFV enabled core, to showcase secure content delivery and low latency applications in large public-events.
2. The **Málaga** Platform. Automated orchestration and management of different network slices over multiple domains, on top of the 5G NR and fully virtualised core network to showcase mission critical services in the lab and in outdoor deployments.
3. The **Limassol** Platform. Radio interfaces of different characteristics and capabilities, combining terrestrial and satellite communications, integrated to showcase service continuity and ubiquitous access in underserved areas.
4. The **Surrey** Platform. Multiple radio access technologies that can support massive Machine Type Communications (mMTC), including 5G NR and NB-IoT, combined under a flexible Radio Resource Management (RRM) and spectrum sharing platform to showcase massive IoT services.
5. The **Berlin** platform: Ultra dense areas covered by various network deployments, ranging from indoor nodes to nomadic outdoor clusters, coordinated via advanced backhauling technologies to showcase immersive service provisioning.

An overview of the common architecture of the 5GENESIS platforms is shown in Figure 32. The evolution of the platforms are split into the following phases

- Phase 1: Delivery of an end-to-end 4G network, NFV/SDN capable, also featuring edge computing capabilities. To that end, Phase 1 focuses on the deployment of the 4G core and radio components (EPC and EUTRAN) as well as the edge computing platform. Phase 1 will also integrate a first version of Slice Manager operating over NFV and WAN domains, supporting slicing over 4G by utilizing different APNs for each slice (limitation is that each mobile can only connect to one slice). Keysight TAP (implementing the Experiment Lifecycle Manager – ELCM) will be integrated in the platform, but with limited capabilities (i.e. establish connection with the slice manager in order to request a slice by sending the NST request).
- Phase 2: Upgrade from 4G to 5G by switching from EUTRAN to 5GNR and also from EPC to 5GC. It will also integrate the network infrastructure over the NMS platforms and demonstrate slicing features, using an upgraded version of the Slice Manager. During this phase, integration of Monitoring and Analytics frameworks will be finalized. Furthermore, first integrations of the use case components will be available.

- Phase 3: integrate the Coordination layer components and the Portal and upgrade the Slice Manager and 5GNR/5GC components in order to demonstrate end-to-end experiment orchestration and lifecycle management over multiple network slices, using a mobile remote 5G hotspot.

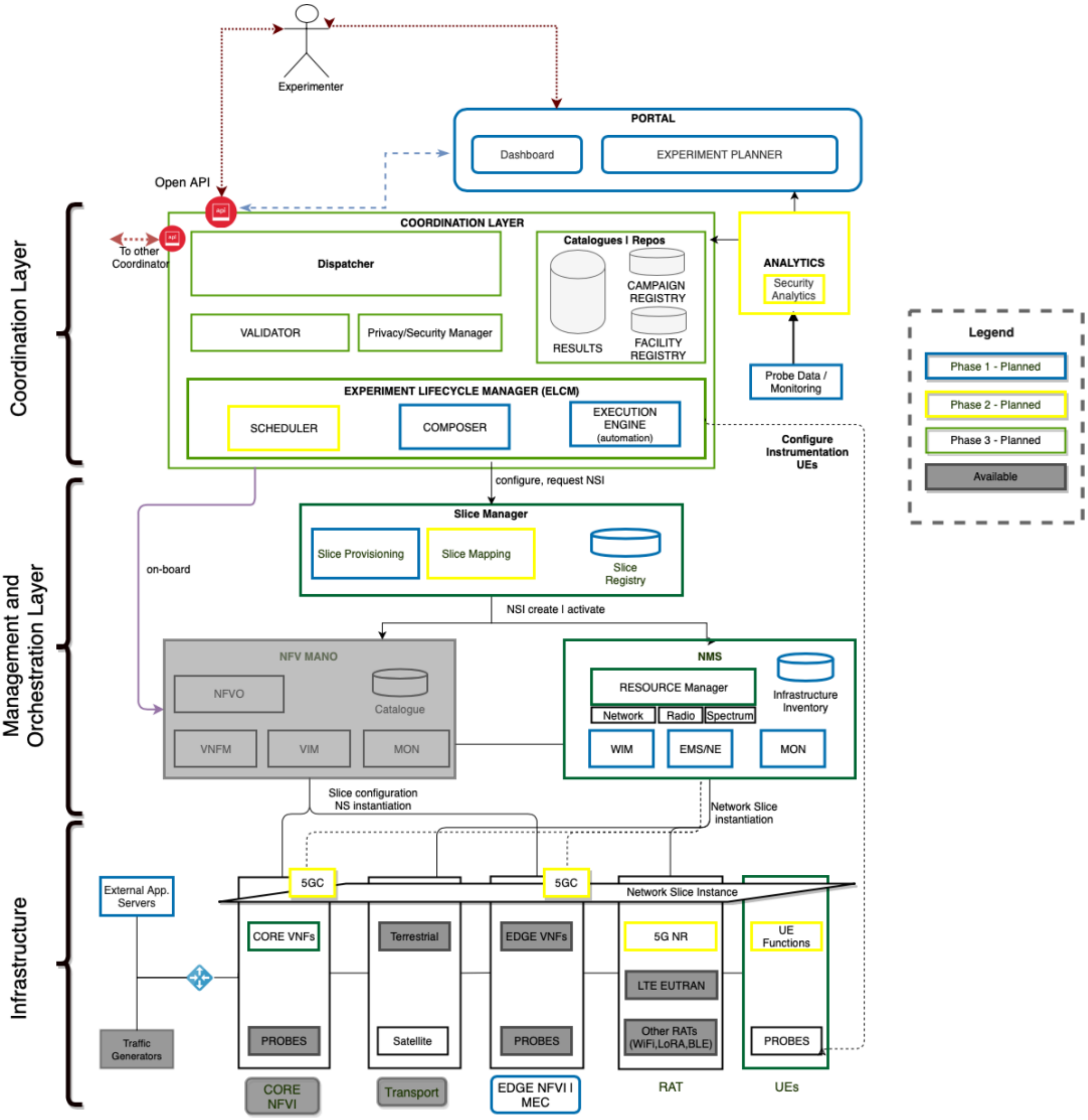


Figure 32: 5GENESIS platform Architecture

3.4 NITOS

UTH is operating since 2007 the Network Implementation Testbed using Open Source platforms (NITOS), which has evolved over the years to a compact solution for evaluating bleeding-edge ideas on the forefront of networking related research. The NITOS testbed is one of the largest single-site open experimental facilities in Europe, allowing users from around the globe to take advantage of highly programmable equipment. The testbed is an integral part of larger federations of resources, such as OneLab (Fdida, Friedman, & Parmentelat, 2010) and Fed4FIRE (Wauters, et al., 2014), enabling experiments with more heterogeneous resources. NITOS has an established user base of over 4000 users in the past years, with over 20 researchers using the infrastructure in a daily basis. In short, the current offering of the testbed is the following:

- Over 100 nodes equipped with IEEE 802.11 a/b/g/e/n/ac compatible equipment, and using open source drivers. The nodes are compatible also with the IEEE 802.11s (Hiertz, et al., 2010) protocol for the creation of wireless mesh networks. The nodes feature multiple wireless interfaces, and are high-end computers, with quad-core Intel Core i5 and Core i7 processing capabilities, 4/8 GBs of RAM and SSD disks.
- Commercial off-the-shelf (COTS) LTE testbed, consisting of a highly programmable LTE macrocell, multiple femtocells, an experimenter configurable EPC network and multiple User Equipment (UE), such as USB dongles and Android Smartphones (Makris, et al., 2015).
- Open Source LTE equipment, running over commodity Software Defined Radio (SDR) equipment, by the adoption of the OpenAirInterface (www.openairinterface.org) platform (Nikaein, et al., 2014). The platform is allowing multiple configurations for creating highly customizable beyond 4G networks.
- COTS WiMAX testbed, based on a highly programmable WiMAX base station in standalone mode (no ASN-GW component), along with several open source WiMAX clients.
- A Software Defined Radio (SDR) 5G testbed, consisting of 10 USRPs N210, 14 USRPs B210, 4 USRPs X310, 2 USRPs N310 and 4 ExMIMO2 FPGA boards. MAC and PHY algorithms are able to be executed over the SDR platforms, with very high accuracy.
- A millimeter wave testbed, operating in the V-band (60GHz), based on six nodes. The platforms support high data-rate point-to-point setups, with beam steering capabilities of up to 90 degrees with a step of 7.5 degrees.
- The nodes are interconnected with each other via 5 OpenFlow hardware switches, sliced using the FlowVisor framework.
- A Cloud Computing testbed, consisting of 96 Cores, 286 GB RAM and 10 TBs of hardware storage. For the provisioning of the cloud, OpenStack is used.
- Multiple WSN clusters, supporting the IEEE 802.15.4, 802.11 and LoRaWAN protocols, gathering measurements such as temperature, luminosity, air quality, radiation emission, etc.

The equipment is distributed across three different testbed locations in the city of Volos and can be combined with each other for creating a very rich experimentation environment. The nodes are running any major UNIX based distributions.

Figure 33: NITOS testbed deployments; indoor/outdoor testbeds and COTS LTE macro-cell

A typical example of a NITOS node is given in the Figure below. Each node is equipped with two Ethernet interfaces (1 Gbps each), one used for controlling the node (e.g. establishing a secure shell connection onto it) and one of them is terminated in a hardware OpenFlow switch that is user controlled and is free of addressing so that each experimenter sets up their own settings for communication. The node is also equipped with two WiFi cards, one compliant with the 802.11ac standard and one with the 802.11n, that can be used by the experimenter in any possible mode (Access Point, Managed, Ad-Hoc, Monitor, Mesh). Some of the nodes are equipped with LTE USB dongles, that can be used to connect to the NITOS provided network, by using AT commands. The dongles are using NITOS specific SIM cards for connecting to the LTE network. Finally, some nodes are equipped with USB3.0 based USRP devices (USRP B210). These can be used to execute software-based base stations, using tools such as srsLTE (Software Radio Systems, 2019) and OpenAirInterface, which are widely used for prototyping 5G radio technologies. Each node has access to the mmWave infrastructure of the testbed, through the experimental Ethernet interface.

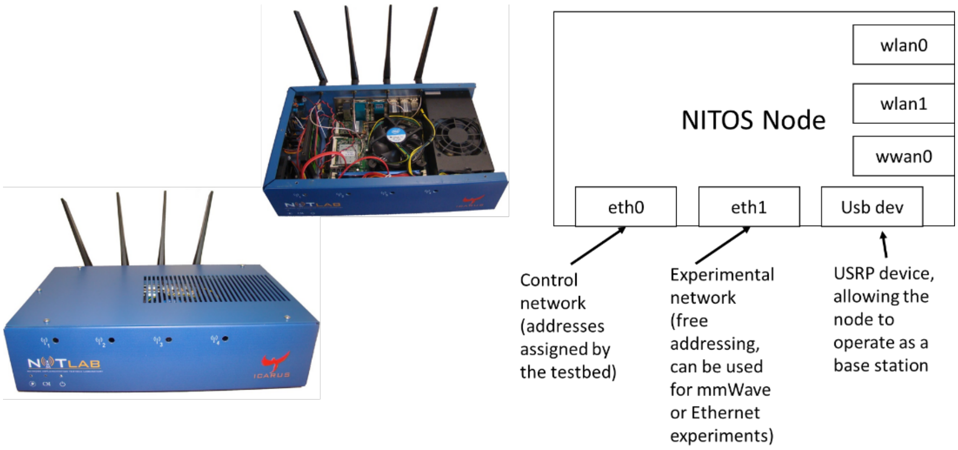


Figure 34: NITOS node architecture

Tools for providing remote access

Over the years, the UTH team has established several tools that allow the remote control of the testbed's resources. Experimenters are able to take advantage of these tools in order to remotely reserve "slices" of the infrastructure, prepare their experimentation environment and post-process their experimentation results (control plane) or tools regarding the actual experimentation process, like generating traffic, invoking handovers, etc (data plane). A brief description of the available tools is provided in the subsections below.

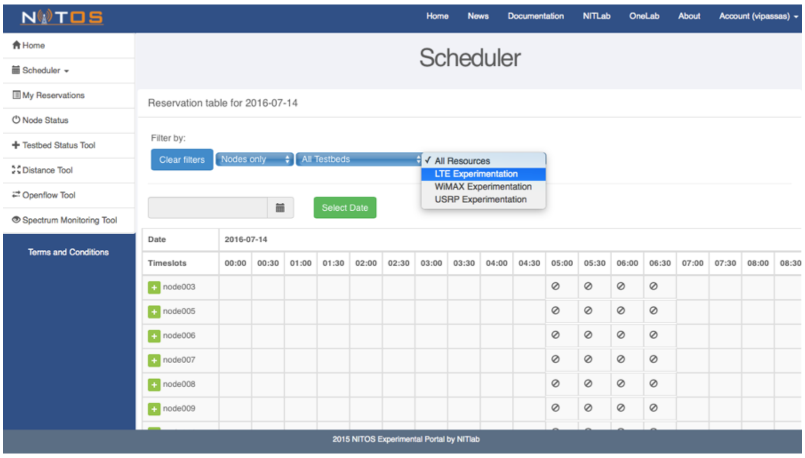
3.4.1 NITOS Broker and Portal

The NITOS broker (Stavropoulos, et al., 2015) is a testbed service based on the OMF6 framework that is directly pluggable to any NITOS-like testbed, and provides essential functionality for the provisioning of the testbeds remotely. Particularly, it provides the following functionality:

- Discovery of the available equipment in each testbed
- Configuration of this equipment tailored to each experimeter's needs (e.g. using a NITOS base station with a 3rd party EPC network using only the Internet connection). This functionality is based on the Resource Specifications (RSpecs) that are used for describing the testbed resources.
- Intercommunication among the different components across federated testbeds for handling resource reservation and access through federation frameworks.
- Setting up and maintaining the proper user accounts for accessing the testbed's nodes.
- Configuring the appropriate access rules on each testbed for isolating concurrent experiments among different users.

The broker entity is interfacing the scheduler of each testbed and based on the resources creates the appropriate Resource Specifications (RSpecs) for advertising the testbed components. It is also featuring multiple APIs for interfacing the SFA API that it provides. The supported APIs are three; 1) an SFA client based, using for example applications like SFI,

2) a REST based and 3) an FRCP based. The NITOS portal is a service that is assisting in coordinating the usage of the testbed's resources among multiple users. It is available in the following address http://nitos.inf.uth.gr. The portal's user interface is responsible for guiding the user through the reservation process, making sure that he/she does not make a reservation conflicting with reservations made by other users. The server component manages the resources, namely wireless nodes and spectrum channels, by associating at the beginning of each time slot (30 min) each resource to the corresponding slice, according to the information communicated by the NITOS broker.



Date	2016-07-14																	
Timeslots	00:00	00:30	01:00	01:30	02:00	02:30	03:00	03:30	04:00	04:30	05:00	05:30	06:00	06:30	07:00	07:30	08:00	08:30
+ node003											⊘	⊘	⊘	⊘				
+ node005											⊘	⊘	⊘	⊘				
+ node006											⊘	⊘	⊘	⊘				
+ node007											⊘	⊘	⊘	⊘				
+ node008											⊘	⊘	⊘	⊘				
+ node009											⊘	⊘	⊘	⊘				

Figure 35: NITOS Portal UI for listing the available resources within a timeslot

3.4.2 *Data Plane tools*

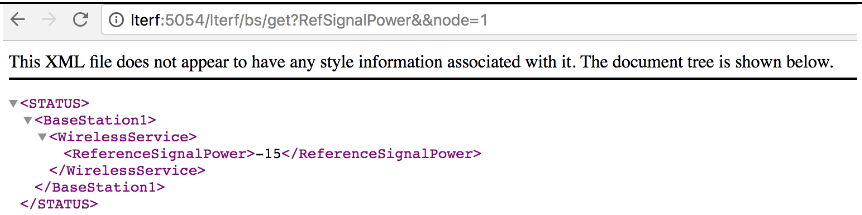
Data plane tools refer to the actual tools that are offered for experimentation with the testbed's nodes. There are two main tools used for controlling the NITOS resources: 1) the bscontrol framework and 2) the OMF framework.

3.4.2.1 BScontrol framework

The bscontrol framework (formerly known as LTErf (Makris, et al., 2015)) has been developed and is maintained by NITOS as a means to control all the different instances of base stations in the testbed, regardless of their RF technology. Through a common REST based interface, experimenters can control similar parameters across the different networks (e.g. transmission power, antenna configuration, modulation and coding schemes, etc.). The tool is currently supporting configuration of the following equipment:

- COTS LTE base stations in NITOS
- OAI base stations in NITOS
- COTS EPC in NITOS
- OAI Core Network
- COTS WiMAX Base Station in NITOS
- WiFi Access Points
- Programmable attenuators for emulated mobility experiments

Control of other types of cells is also supported, such as OpenBTS cells for 2G/3G experimentation, and other Core Networks, that are not currently hosted in NITOS. The tool is also handling the "datapath" configuration of the cells, meaning the routing of traffic beyond the base station/core network. The experimenter is able to relay the traffic that is generated within the network to either the public Internet through a NAT connection, or to a dedicated network connection to another testbed over a GEANT virtual circuit.


lterf:5054/lterf/bs/get?RefSignalPower&&node=1

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<STATUS>
  ▼<BaseStation1>
    ▼<WirelessService>
       <ReferenceSignalPower>-15</ReferenceSignalPower>
     </WirelessService>
   </BaseStation1>
 </STATUS>
```

Figure 36: Browser query to the service for retrieving the power level value of the femtocell

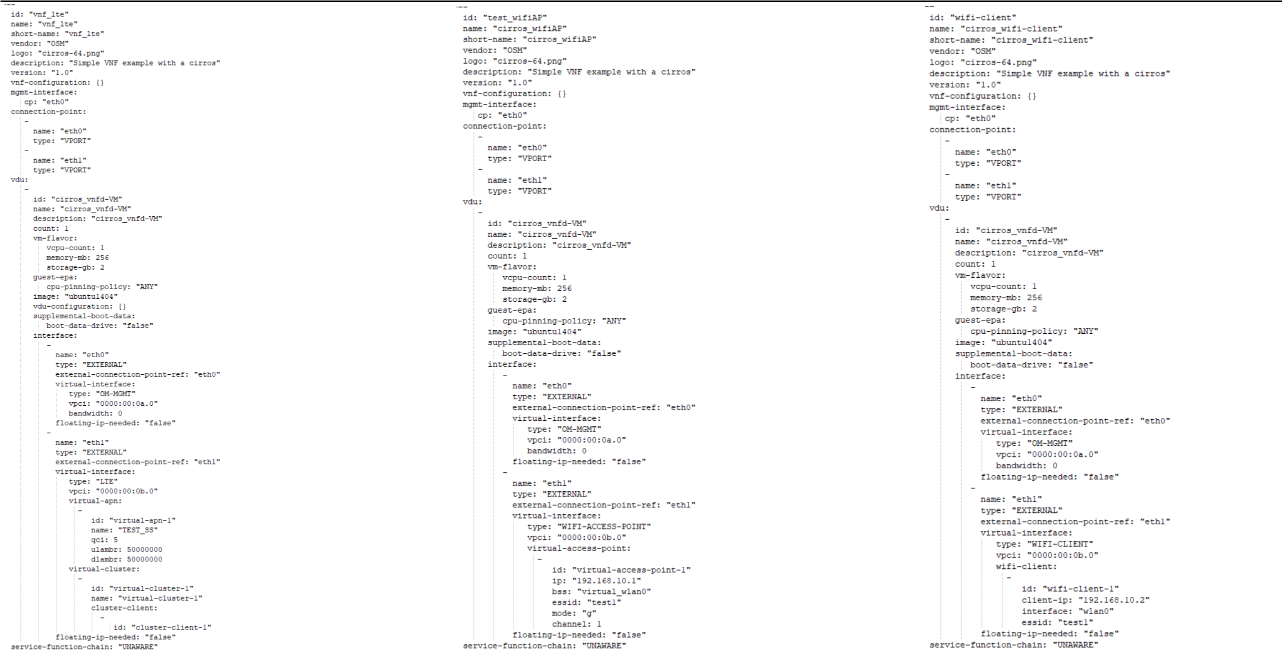
3.4.2.2 OMF framework

OMF is a framework developed exclusively in Ruby language and based on the exchange of messages with a publish-subscribe scheme for controlling and managing networking devices. The messaging scheme can be either realized through the XMPP (for all the OMF versions) and the AMQP protocols (only in version 6). OMF is an orchestrating framework for handling the entire experiment lifecycle, and thus depends on a suite of software tools and components for providing the experimentation services (such as dynamic addressing, remote OS loading, data archiving, etc.). OMF is handling the testbed nodes based on the Metal as a Service paradigm. With OMF, the experimenters of the testbed are able to design their experiment with a high-level approach, in the OMF Experiment Description Language (OEDL), with an experiment description resembling a simulation. Once this description is submitted to the testbed, the OMF tools orchestrate the experiment execution on top of the testbed's machines. They get full root access on the testbed nodes, and thus can configure complex experiments that may use experimenter defined hardware drivers for the wireless infrastructure, etc.

3.4.3 MANO tools

NITOS testbed is hosting an OpenSourceMANO installation for managing the testbed components, using the nodes of the testbed as the compute nodes. OpenSourceMANO is an NFV-MANO compliant orchestrator. The NFV-MANO architecture is providing the necessary abstractions of the underlying hardware equipment, and concentrates only on the orchestration, provisioning and cross- interaction of the deployed functions, taking care of all the low-level configurations for setting up end-to-end paths between the functions. Nevertheless, NFV-MANO is mainly addressing datacenter resources, with the networking being programmed through the SDN concept, whereas services are deployed using either virtual machines or light-weight containers. This approach can extend to generic networking devices (e.g. like the ones that are present in NITOS testbed) as long as they are organized in a distributed fashion, which includes other technologies (such as wireless) that are not currently addressed by SDN through production grade software.

Nevertheless, the heterogeneous wireless technologies that exist in the testbed (e.g. mmWave, LTE, WiFi) are not inherently supported by OSM. To this aim, the VIM instance that is installed in NITOS testbed is using an updated syntax for the VDU components of the VNFs, supporting the configuration of the wireless interfaces of the nodes. These interfaces are subsequently bridged with the actual interfaces that are used in the provisioned VNFs. Using this approach, the experimenters are able to access the testbed by using only OSM and depicting the interconnection of the VNFs, even in the wireless domain. To this aim, the NITOS VIM instance is able to consume VNFDs that are augmented with configuration parameters for wireless networks as well. The following figures present the VNFDs supporting configuration of the WiFi and LTE networks of the testbed.



```
---
id: "vnf_lte"
name: "vnf_lte"
short-name: "vnf_lte"
vendor: "OSM"
logo: "cirros-64.png"
description: "Simple VNF example with a cirros"
version: "1.0"
vnf-configuration: {}
mgmt-interface:
   cp: "eth0"
connection-point:
   -
     name: "eth0"
     type: "VPORT"
   -
     name: "eth1"
     type: "VPORT"
vdu:
   -
     id: "cirros_vnfd-VM"
     name: "cirros_vnfd-VM"
     description: "cirros_vnfd-VM"
     count: 1
     vm-flavor:
        vcpu-count: 1
        memory-mb: 256
        storage-gb: 2
     guest-epa:
        cpu-pinning-policy: "ANY"
     image: "ubuntu1404"
     vdu-configuration: {}
     supplemental-boot-data:
        boot-data-drive: "false"
     interface:
        -
          name: "eth0"
          type: "EXTERNAL"
          external-connection-point-ref: "eth0"
          virtual-interface:
             type: "OM-MGMT"
             vpci: "0000:00:0a.0"
             bandwidth: 0
          floating-ip-needed: "false"
        -
          name: "eth1"
          type: "EXTERNAL"
          external-connection-point-ref: "eth1"
          virtual-interface:
             type: "LTE"
             vpci: "0000:00:0b.0"
             virtual-apn:
                -
                   id: "virtual-apn-1"
                   name: "TEST_SS"
                   qci: 5
                   ulambr: 50000000
                   dlambr: 50000000
             virtual-cluster:
                -
                   id: "virtual-cluster-1"
                   name: "virtual-cluster-1"
                   cluster-client:
                     -
                        id: "cluster-client-1"
          floating-ip-needed: "false"
service-function-chain: "UNAWARE"
```

```
---
id: "test_wifiAP"
name: "cirros_wifiAP"
short-name: "cirros_wifiAP"
vendor: "OSM"
logo: "cirros-64.png"
description: "Simple VNF example with a cirros"
version: "1.0"
vnf-configuration: {}
mgmt-interface:
   cp: "eth0"
connection-point:
   -
     name: "eth0"
     type: "VPORT"
   -
     name: "eth1"
     type: "VPORT"
vdu:
   -
     id: "cirros_vnfd-VM"
     name: "cirros_vnfd-VM"
     description: "cirros_vnfd-VM"
     count: 1
     vm-flavor:
        vcpu-count: 1
        memory-mb: 256
        storage-gb: 2
     guest-epa:
        cpu-pinning-policy: "ANY"
     image: "ubuntu1404"
     supplemental-boot-data:
        boot-data-drive: "false"
     interface:
        -
          name: "eth0"
          type: "EXTERNAL"
          external-connection-point-ref: "eth0"
          virtual-interface:
             type: "OM-MGMT"
             vpci: "0000:00:0a.0"
             bandwidth: 0
          floating-ip-needed: "false"
        -
          name: "eth1"
          type: "EXTERNAL"
          external-connection-point-ref: "eth1"
          virtual-interface:
             type: "WIFI-ACCESS-POINT"
             vpci: "0000:00:0b.0"
             virtual-access-point:
                -
                   id: "virtual-access-point-1"
                   ip: "192.168.10.1"
                   bss: "virtual_wlan0"
                   essid: "test1"
                   mode: "g"
                   channel: 1
          floating-ip-needed: "false"
service-function-chain: "UNAWARE"
```

```
--
id: "wifi-client"
name: "cirros_wifi-client"
short-name: "cirros_wifi-client"
vendor: "OSM"
logo: "cirros-64.png"
description: "Simple VNF example with a cirros"
version: "1.0"
vnf-configuration: {}
mgmt-interface:
   cp: "eth0"
connection-point:
   -
     name: "eth0"
     type: "VPORT"
   -
     name: "eth1"
     type: "VPORT"
vdu:
   -
     id: "cirros_vnfd-VM"
     name: "cirros_vnfd-VM"
     description: "cirros_vnfd-VM"
     count: 1
     vm-flavor:
        vcpu-count: 1
        memory-mb: 256
        storage-gb: 2
     guest-epa:
        cpu-pinning-policy: "ANY"
     image: "ubuntu1404"
     supplemental-boot-data:
        boot-data-drive: "false"
     interface:
        -
          name: "eth0"
          type: "EXTERNAL"
          external-connection-point-ref: "eth0"
          virtual-interface:
             type: "OM-MGMT"
             vpci: "0000:00:0a.0"
             bandwidth: 0
          floating-ip-needed: "false"
        -
          name: "eth1"
          type: "EXTERNAL"
          external-connection-point-ref: "eth1"
          virtual-interface:
             type: "WIFI-CLIENT"
             vpci: "0000:00:0b.0"
             wifi-client:
                -
                   id: "wifi-client-1"
                   client-ip: "192.168.10.2"
                   interface: "wlan0"
                   essid: "test1"
          floating-ip-needed: "false"
service-function-chain: "UNAWARE"
```

Figure 37: NITOS VNFDs for empty VNFs with an experimental LTE link (left), WiFi in AP configuration (middle) and WiFi station mode (right)

3.5 R2LAB

R2lab (www.r2lab.inria.fr) is an open tested put in place by INRIA Sophia Antipolis in France. The testbed radios are located in an anechoic chamber and its framework allows for reproducible research in wireless WiFi and 4G/5G networks. R2lab is part of the French FIT federation, an open large-scale testing infrastructure for systems and applications on wireless and sensor communications.

R2lab proposes thirty seven customisable commercial off-the-shelf wireless devices, together with USRP nodes and commercial LTE phones, fit to create rich experimental setups. The testbed also features advanced software like leverage GnuRadio and OpenAirterface, as well as efficient software tools, to support easy experimentation.

These tools allow to book the whole testbed, to remotely control the wireless devices, to easily deploy various scenarios and to collect results. The facility can be access via a portal by experimenters either to be used as a sandbox or by preconfiguring configurations to be tested automatically. As described in Section 3.1.1, the R2Lab facility is used remotely to test the OAI codebase in the context of OAI's continuous delivery procedures.

3.6 COSMOS

The COSMOS testbed is intended to *enable several new classes of wireless experiments not currently supported by testbeds available to the research community*. The "sweet spot" for experimenters is ultra-high access bandwidth coupled with low latency mobile networks and edge cloud services, an attractive target because it enables an entirely new class of interactive rich media applications associated with scenarios such as AR/VR for mobile users or cloud-assisted connected cars. There is a great deal of exploratory research still to be done at various layers of the system. To enable this research the platform offers the following major hardware components deployed outdoors in West Manhattan near the Columbia University campus: (1) SDR Radio Nodes – 100 Small (both mobile and fixed), 20 Medium (pole/building mount) and 9 Large (roof-top installation); (2) 27 Ultra-High-Speed WDM Optical Switches for front-haul interconnect; (3) 15 Edge Cloud Servers co-located with large radio nodes; (4) 10 SDN switches for the back-haul network with NYSERNet high-speed connectivity; (5) 4 Datacenter capacity Cloud Computing Racks for hosting centralized testbed services (including user portal, network control and experiment management), and for network and higher layer protocols and applications in testbed experiments; (7) COSMOS sandboxes for small-scale experiments and testing, each consisting of 4-6 nodes and servers, Ethernet and RF matrix interconnect; (6) IP routers forming a private network for wide-area connectivity between the NYC deployment site, the NYU PoP at 32 Avenue of Americas, the Rutgers campus network, and gateway to other PAWR testbeds (I2/GENI) and the Internet.

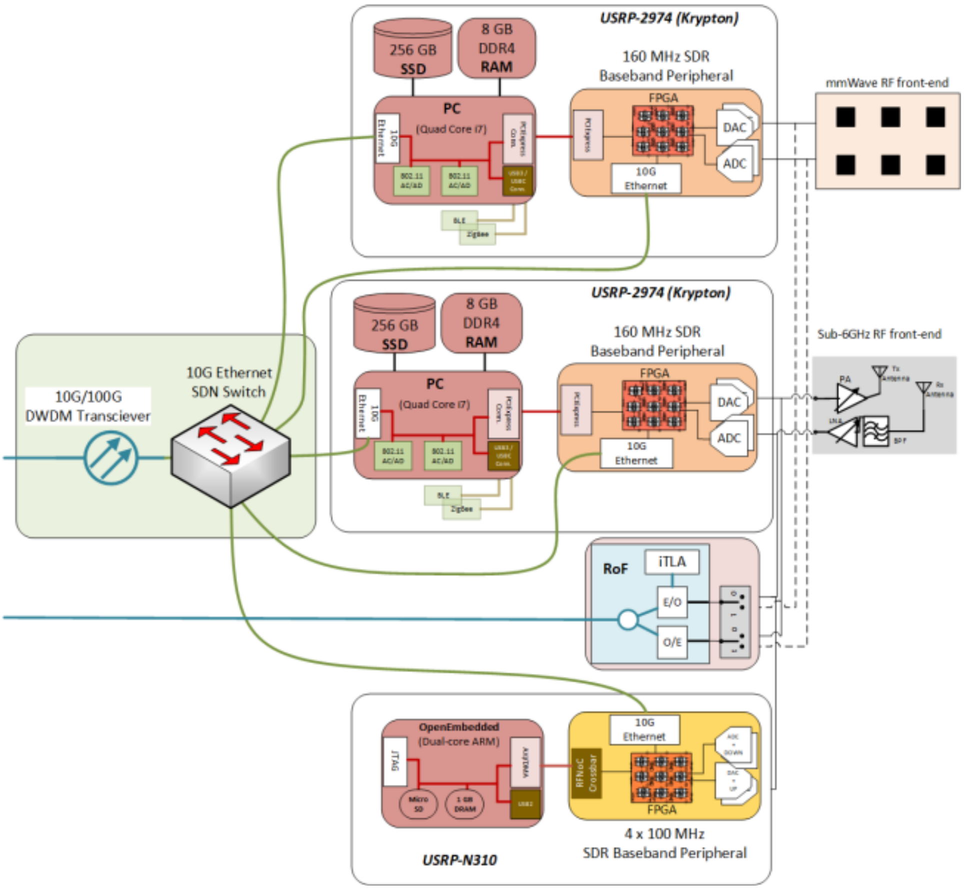


Figure 38: COSMOS Node Design

The component level equipment list available from COSMOS

Component Sub-systems	Quantity
320 x 320 Space switch	2
ROADM	27
CU Fiber	1
USRP 2974	44
USRP N310	37
Antenna case	37

Power amplifier (ZVE-8G+)	64
Large standard AC case	29
Compute server with GPU	60
PCIe FPGA Add-on Card	60
Large 100G Switch	5
10G Tunable DWDM tranceiver	20
16 x 16 Space switch	5
10G Tunable sfp	40
Medium street-furniture case	20
Heat-pipes and cooling	9
USRP E312	50
USRP B205mini	50
USRP B210	10
RFSoC	20
10G RoF Link	10
Infrastructure server	4
Cluster Rack Kit, 42U, Deep Rack	4
Directional antenna case and radome cover	0
Omnidirectional antenna case and radome cover	0

3.7 POWDER

To realize the vision of city-scale experimentation, the POWDER/RENEW team will place dozens of fixed base stations over an area of fourteen square kilometers, with sixty mobile devices traveling on couriers and seventy sensor nodes, either fixed or mobile, deployed throughout the area. This contiguous space covers three distinct environments: a high-rise urban downtown, a moderate density residential area, and a limited-scale radio network on the University of Utah campus. Also included in the deployment will be a custom set of massive full-duplex MIMO devices provided by RENEW, which can be potentially scaled to 160 antennas and operational over a wide range of frequency bands covering 470-700 MHz UHF, 2.4-2.7 GHz and 3.5-3.8 GHz.

The standard hardware package will provide a common experimental infrastructure and control system across our entire testbed.

Massive MIMO Base station: The RENEW base station (Figure 39) is based on Argos base station but with important updates. It consists of daisy chains of Iris modules as depicted, using two specialized bus connectors on each side of Iris modules. The bus connectors carry reference clock and power as well as 4 GPIO lines and 13.2 Gbps serial link routed to the FPGA fabric, thereby eliminating the need for cabling between Iris modules while locking all daisy-chained Iris modules in frequency and enabling phase-coherent MIMO operation. A Hub board like the Xilinx VCU118 development board combines multiple daisy chains coherently through a customized backplane that provides high-precision clocking, power, and bi-directional connectivity. The Hub also aggregates/ distributes streams of samples from/to all radios.

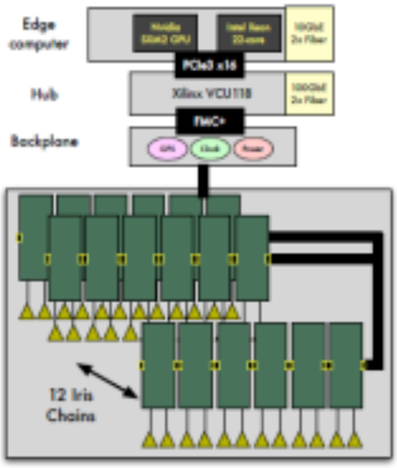


Figure 39: RENEW Basestation

RF Base Unit: The RF base unit will be contained in a weather-resistant enclosure. Enclosures will be placed in backpacks, mounted in courier vehicles, or installed at base stations. Inside, we integrate infrastructure for centralized control, experimental hardware, and telemetry. The controller is a single-board computer that fulfills three roles. First, it provides power control for experimental hardware. Second, it verifies spectrum licensing compliance by monitoring RF transmissions.

Third, the controller performs experimental setup and teardown; it re-images the disks and resets any hardware configuration for all experimental hardware. When part of a base station, the control network will be a wired backhaul connection. For mobile endpoints, low-bandwidth control will be done with commodity LTE networking while high-bandwidth control will use commodity Wi-Fi during times when it is available.

The experimental hardware consists of general purpose single-board computers, software defined radios, and standard LTE UE equipment. We will have versions of the base unit with different numbers of SDR and compute elements. Each SDR is attached to and controlled by a small form factor PC to which experimenters have full and exclusive access. The standard LTE UE device is an Android smartphone, to be controlled by experimenters via a proxied ADB interface and console.

Regular Base Station: These base stations include a 3-SDR version of the RF base unit, power supply, directional antennas, and an RF matrix. The RF matrix uses wideband circulators and programmable switching and filter elements to allow any piece of hardware to be connected to any antenna. Due to space, power, and cooling considerations, these base stations will be broken into two pieces: the rooftop RF component and a data closet compute component. The compute nodes and network switches will reside in a data closet, typically in the building’s basement.

Dense Deployment Base Station: All dense base stations will include two of POWDER’s standard wireless base station units. They will target the 5.8 GHz ISM band, which simplifies antenna requirements versus the wide spectrum coverage offered by POWDER’s other base stations. The dense deployment will use small omnidirectional antennas to allay concerns related to physical load and visibility. The University’s facilities management group has worked with us to determine the best options for power and network provisioning for about 8 dense deployment sites.

Vehicle-based Mobile Endpoint: Installations mounted inside vehicles will provide POWDER’s main mobile couriers. Each vehicle will include the standard base unit control components and between one and three wireless SDR plus small form factor PC pairs. Each SDR+PC pair will have its own antenna, with a tunable RF filter and circulator (for TX/RX sharing) inline. Vehicle-based mobile endpoints will be deployed in campus shuttles (40 endpoints) and other campus or city-operated vehicles (30 endpoints). Each vehicle-based mobile endpoint has a combined power conditioner to avoid surges, and UPS system to allow orderly shutdown of systems when the vehicle turns off.

Portable Mobile Endpoint: Portable mobile endpoints encompass a variety of use-cases, including operation mounted in a motorized courier or simply carried by an experimenter. To allow their use for several hours without power, we will include a large capacity rechargeable battery to power the RF base unit and any other peripherals. In order to extend the lifetime of the battery, especially when used outside, each portable mobile endpoint will also be provided with a solar charging array. Portable endpoints will have fewer SDRs, but each SDR will be provided with a separate antenna for transmission and reception.

Edge Compute: The need to reduce latency between the wireless access medium and accessible compute will be addressed by POWDER's edge compute resources. Compute resources at each edge compute site will consist of servers with at least 64 GB of RAM and dual socket, 20+ core processors. Paired with these compute resources, there will be at least 24 TB of iSCSI-exported storage, which can be flexibly allocated or shared among the edge compute nodes at a site. Edge compute sites will have better-provisioned backhaul links to the metro compute clusters, typically 40 Gb/s or better depending on the site.

Sensor Units POWDER will include flexible sensor units that can either be paired with existing mobile unit courier deployments, or operate standalone in locations throughout the proposed coverage area. These sensor units will have broad capabilities for sensor pairing as well as for IoT networking protocols. The core of the unit will be the same single-board computer used in the base wireless units. We plan to include environmental sensors available via the in-kind contributions from PAWR consortium members. A flexible sensor radio element will be included, capable of communication over three key IoT radio protocols: LoRa, Sigfox, and LTE-M. An RTL-SDR software defined radio will also be present. This inexpensive SDR will provide a means to independently measure and monitor the IoT RF environment.

Network POWDER base stations and edge compute resources will be connected by a high-speed wired backhaul network. The network consists of two primary aggregation points–one on campus and one downtown–which form a backbone connected via 2 x 100GbE. Rooftop base stations will be connected directly to one of the two aggregation points with 100GbE. Massive MIMO sites on campus will be co-located with base stations and share their 100GbE uplink. The campus dense deployment sites (not shown) will be connected to the campus aggregation site with 10GbE.

Logically there are two distinct networks, the control and management network (the control plane) and the experiment network (the data plane). The former is used by the control framework for managing and monitoring switches and nodes as well as providing Internet access to devices for user access. Both are carried over the same fiber infrastructure out to the base station and edge sites.

Compute In addition to single-board, small form factor computers (e.g., an Intel NUC) there will be rack mount servers supporting faster processors and increased RAM and disk capacity. While not finalized, it is expected that many of the rack mount servers will contain a GPU or FPGA for specialized computation.

The two major network aggregation points, the Fort and Downtown datacenters will each host 16 additional rack mount servers for general purpose computing as well as iSCSI-based shared and persistent storage provided by a rack mount server with 20-50TB of available disk space. The Fort datacenter will also host three even more powerful servers intended to be coupled with the three mMIMO base-stations to provide for their computing needs. These two near edge computing facilities should provide significant additional computing and storage capacity with less than 15ms round-trip latency to the base stations.

The metro computing facilities are located at the Downtown datacenter and the Merrill Engineering Building.

Cloud Compute The Utah CloudLab and Apt facilities at the Downtown datacenter feature over 750 allocatable servers within a 30ms round-trip of the downtown base stations. The Emulab cluster in MEB has over 500 servers within a 30ms round-trip of the campus base-stations.

3.8 Colloseum

At its core, Colosseum is a specialized data center with significant amount of hardware. It has 900 TB of Network Attached Storage (NAS), 171 high-performance servers, 256 USRP X310s (128 as communications devices, 128 as part of the channel emulator), 18 10G switches, 19 clock distribution systems, and hundreds of high-speed optical connections. It requires a 800 sqft dedicated facility to store 21 racks, 40 ton HVAC, and 65 kW, 208/120 VAC 3 phase power. All these features come with a sophisticated experiment control framework, traffic generation system, real world scenario creation and channel reconfiguration.

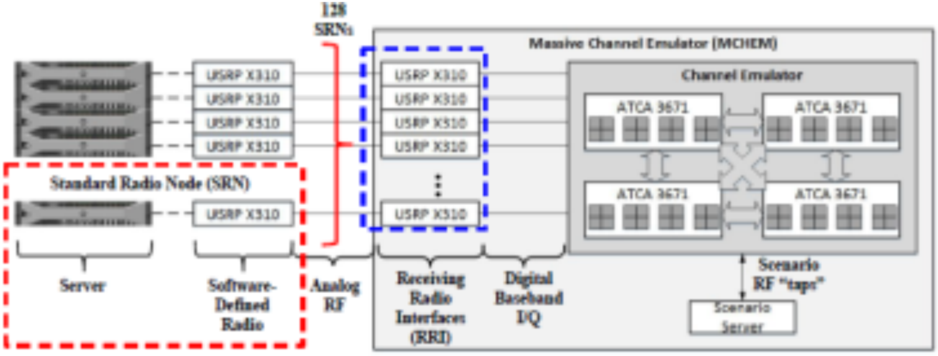


Figure 40: High Level Overview of Colosseum Architecture

3.9 AERPAW

AERPAW (https://aerpaw.org), the USA's first aerial wireless experimentation platform spanning 5G technologies and beyond, will enable cutting-edge research. It has the potential to create transformative wireless advances for aerial systems. On the AERPAW platform, drones and 5G are integrated to be mutually beneficial. Drones are supporting 5G by providing increased coverage and connectivity; and 5G is supporting drones by providing improved signals and location data.

On today's networks, fixed nodes enable 4G signals to connect to wireless devices. On the AERPAW platform, nodes will be mobile, with the ability to transmit and receive radio waves from user devices while moving on demand. For example, in the aftermath of a natural disaster such as a hurricane, existing cellular networks may be damaged. As a result, aerial base stations can position themselves to provide the best wireless coverage to victims and first responders who would otherwise have no cellular connectivity.

Drones are not the only mobile nodes. Researchers will also be putting 5G equipment on cars, buses, golf carts and rovers for vehicle-to-vehicle communications, which will support autonomous driving and accident reduction.

4. Conclusion

We provided a survey of software tools, deployment and testing methodologies and computing platforms used in major current EU and US platforms. This furthermore constitutes the initial EMPOWER software catalogue, high-level documentation and collaboration methods and objectives for transatlantic mutualisation of components and systems for wireless experimentation. In particular, we provided an overview of open-source Core Network, Radio-Access Network and Edge Computing software packages which can be used to build proof-of-concept experimental networks. We also provided an overview of existing facilities in the EU and USA which offer remote-access to experimental infrastructure which can be used to prototype 3GPP radio and core network technologies.

5. References

5GENESIS. (2019). 5th Generation End-to-end Network, Experimentation, System Integration, and Showcasing. https://5g-ppp.eu/5genesis/.

5G-EVE. (2019). 5G European Validation platform for Extensive trials. https://5g-ppp.eu/5g-eve/.

5G-VINNI. (2019). 5G-VINNI: 5G Verticals INNovation Infrastructure. https://5g-ppp.eu/5g-vinni/.

COSMOS. (2019). http://cosmos-lab.org/.

Fdida, S., Friedman, T., & Parmentelat, T. (2010). OneLab: An open federated facility for experimentally driven future internet research. In *New Network Architectures* (pp. 141-152). Springer Berlin Heidelberg.

Forum, S. C. (2017). *NFAPI (SCF082.09.05).* Small Cell Forum.

GNU Radio. (2019). www.gnuradio.org.

Hamano, J., & Torvalds, L. (n.d.). git. https://git-scm.com/.

Hiertz, G. R., Denteneer, D., Max, S., Taori, R., Cardona, J., Berlemann, L., & Walke, B. (2010). IEEE 802.11s: the WLAN mesh standard. *IEEE Wireless Communications, 17*(1), 104-111.

Kawaguchi, K. (2019). Jenkins. www.jenkins.io.

Lehmann, L. (2018). “The position of ITU-T Study Group 13 in global 5G Standardization: Softwarization, Fix Mobile Convergence and Data Aware Networking”. *Sixth Regional Workshop for Africa on "Standardization of future networks: What opportunities for Africa?".* Abidjan, Cote D'Ivoire.

Linux Foundation. (019). *Open Networking Summit Europe.* https://events19.linuxfoundation.org/events/open-networking-summit-europe-2019/.

Linux Foundation. (2019). *KubeCon North America.* https://events19.linuxfoundation.org/events/kubecon-cloudnativecon-north-america-2019/.

Linux Foundation. (2019). *Open Network Automation Platofmr.* www.onap.org.

Linux Foundation. (2019). *Open Networking Summit.* https://events19.linuxfoundation.org/events/open-networking-summit-north-america-2019/.

Linux Foundation. (2019). *Open Platform for Network Function Vittualization.* www.opnfv.org.

Linux Foundation. (2019). Operator Defined Next Generation RAN and Interfaces. www.o-ran.org.

Linux Foundation. (2019). *Virctual Central Office 3.0.* https://wiki.opnfv.org/display/OSDD/VCO+Demo+3.0+Home.

Linux Foundation. (2020). *KubeCon Europe.* https://events19.linuxfoundation.org/events/kubecon-cloudnativecon-europe-2020/.

Makris, N., Zarafetas, C., Kechagias, S., Korakis, T., Seskar, I., & Tassiulas, L. (2015). Enabling open access to LTE network components; the NITOS testbed paradigm. *1st IEEE Conference on Network Softwarization (NetSoft).* London.

Nikaein, N., Mahesh, K. M., Saravana, M., Alex, D., Raymond, K., & Christian, B. (2014). OpenAirInterface: A Flexible Platform for 5G Research. *ACM SIGCOMM Computer Communication Review, 44*(5), 33-38.

OpenAirInterface. (2019). www.openairinterface.org.

openLTE. (2018). https://github.com/mgp25/OpenLTE.

POWDER-RENEW. (2019). Platform for Open Wireless Data-driven Experimental Research. https://powderwireless.net.

Redhat. (019). OpenShift 4.2. https://docs.openshift.com/container-platform/4.2/welcome/index.html.

Slack. (2019). www.slack.com.

Software Radio Systems. (2019). srsLTE. https://www.softwareradiosystems.com/tag/srslte/.

Stavropoulos, D., Dadoukis, A., Rakotoarivelo, T., Ott, M., Korakis, T., & Tassiulas, L. (2015). Design, architecture and implementation of a resource discovery, reservation and provisioning framework for testbeds. *13th*

International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt). Mumbai.

Wauters, T., Vermeulen, B., Vandenberghe, W., Demeester, P., Taylor, S., Baron, L., . . . others. (2014). Federation of Internet experimentation facilities: architecture and implementation. *IEEE European Conference on Networks and Communications (EuCNC).* Bologna.

Zdarsky, F. (2019). OPENAIRINTERFACE/openair-k8s. https://github.com/OPENAIRINTERFACE/openair-k8s.