DISTRIBUTED SERVICES ORCHESTRATION FOR 5G AND BEYOND 5G NETWORKS

BÁRBARA VALERA MUROS

Tesis Doctoral
Programa de Doctorado en Tecnologías Informáticas
Escuela Técnica Superior de Ingeniería Informática

Universidad de Málaga

Málaga, 2025

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

AUTORA: Bárbara Valera Muros

http://orcid.org/0000-0003-4239-0388

EDITA: Publicaciones y Divulgación Científica. Universidad de Málaga

Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional:
Cualquier parte de esta obra se puede reproducir sin autorización
pero con el reconocimiento y atribución de los autores.
No se puede hacer uso comercial de la obra y no se puede alterar, transformar o hacer obras derivadas.
http://creativecommons.org/licenses/by-nc-nd/4.0/legalcode

Esta Tesis Doctoral está depositada en el Repositorio Institucional de la Universidad de Málaga (RIUMA): riuma.uma.es

UNIVERSIDAD
DE MÁLAGA

Bárbara Valera Muros: *Distributed Services Orchestration for 5G and Beyond 5G Networks,* Doctor of Philosophy Thesis, © Málaga 2025

UNIVERSIDAD DE MÁLAGA

DISTRIBUTED SERVICES ORCHESTRATION FOR 5G AND BEYOND 5G NETWORKS

BÁRBARA VALERA MUROS

Doctor of Philosophy Thesis
Supervised by Pedro Merino Gómez
and Laura Panizo Jaime

Department of Computer Science
University of Malaga

Málaga, 2025

UNIVERSIDAD DE MÁLAGA

Bárbara Valera Muros: *Distributed Services Orchestration for 5G and Beyond 5G Networks,* Doctor of Philosophy Thesis, © Málaga 2025

UNIVERSIDAD DE MÁLAGA

AUTHORSHIP STATEMENT

DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD DE LA TESIS PRESENTADA PARA OBTENER EL TÍTULO DE DOCTOR

Dña. *Bárbara Valera Muros*
Estudiante del programa de doctorado en *Tecnologías Informáticas* de la Universidad de Málaga, autora de la tesis, presentada para la obtención del título de doctor por la Universidad de Málaga, titulada:
Distributed Services Orchestration for 5G and Beyond 5G Networks

Realizada bajo la tutorización y dirección de *Pedro Merino Gómez* y *Laura Panizo Jaime*.

DECLARO QUE:

La tesis presentada es una obra original que no infringe los derechos de propiedad intelectual ni los derechos de propiedad industrial u otros, conforme al ordenamiento jurídico vigente (Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia), modificado por la Ley 2/2019, de 1 de marzo.

Igualmente asumo, ante a la Universidad de Málaga y ante cualquier otra instancia, la responsabilidad que pudiera derivarse en caso de plagio de contenidos en la tesis presentada, conforme al ordenamiento jurídico vigente.

En Málaga, 26 de mayo de 2025

Doctoranda	Tutor

Directores

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

SUPERVISORS AUTHORIZATION

Por la presente, Dr. Pedro Merino Gómez y Dra. Laura Panizo Jaime, profesores del Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga, CERTIFICAN:

Que Bárbara Valera Muros ha realizado en el Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga bajo su dirección, el trabajo de investigación correspondiente a su TESIS DOCTORAL titulada:

Distributed Services Orchestration for 5G and Beyond 5G Networks

En dicho trabajo se han propuesto aportaciones originales en el campo de la orquestación de servicios distribuidos para redes móviles de última generación. Los resultados expuestos han dado lugar a las siguientes publicaciones en revistas y aportaciones a congresos:

1. A. Rios, B. Valera-Muros, P. Merino-Gomez, and J. Sobieski, "Expanding GÉANT Testbeds Service to Support Pan-European 5G Network Slices for Research in the EuWireless Project," in Mobile Information Systems, Vol. 2019, Article ID 6249247, doi: 10.1155/2019/6249247

2. B. Valera-Muros and P. Merino-Gomez, "Is GÉANT Testbeds Service compliant with ETSI MANO?," in Proceedings of the 2019 IEEE 2nd 5G World Forum (5GWF), Dresden, Germany, 2019, pp. 502-507, doi: 10.1109/5GWF.2019.8911622

3. I. Harjula, L. Panizo, B. Valera-Muros, J. Pinola, M. Hoppari, and A. Flizikowski, "Dynamic Spectrum Management for Euro-pean-Wide Research Network," in Proceedings of the 2020 IEEE 91st Vehicular Technology Conference (VTC2020), Antwerp, Belgium, 2020, pp. 1-6, doi: 10.1109/VTC2020-Spring48590.2020. 9129017

4. B. Valera-Muros, L. Panizo, A. Rios, and P. Merino-Gomez, "An Architecture for Creating Slices to Experiment on Wireless Networks," in Journal of Network and Systems Management 29, 1 (2021), doi: 10.1007/s10922-020-09571-8

5. G. Margetis, B. Valera-Muros, K. C. Apostolakis, A. Díaz Zayas, L. Panizo, P. Tomás, "Validation of NFV management and orchestration on Kubernetes-based 5G testbed environment," 2022 IEEE Globecom Workshops (GC Wkshps), Rio de Janeiro, Brazil, 2022, pp. 844-849, doi: 10.1109/GCWkshps56602.2022. 10008690

UNIVERSIDAD DE MÁLAGA

6. K. C. Apostolakis, B. Valera-Muros, N. di Pietro, P. Garrido, D. del Teso, M. Kamarianakis, P. Tomás, H. Khalili, L. Panizo, A. Díaz Zayas, A. Protopsaltis, G. Margetis, J. Mangues-Bafalluy, M. Requena-Esteso, A. Gomes, L. Cordeiro, G. Papagiannakis, and C. Stephanidis, "A network application approach towards 5G and beyond critical communications use cases," in Frontiers in Communications and Networks, Vol. 5 - 2024, doi: 10.3389/frcmn.2024.1286660

Estas publicaciones en coautoría que avalan la tesis no han sido utilizadas en tesis anteriores. Por todo ello, consideran que esta Tesis es apta para su presentación al Tribunal que ha de juzgarla. Y para que conste a efectos de lo establecido, AUTORIZAN la presentación de esta Tesis en la Universidad de Málaga.

En Málaga, 26 de mayo de 2025

Dr. Pedro Merino Gómez	Dra. Laura Panizo Jaime

UNIVERSIDAD DE MÁLAGA

ACKNOWLEDGEMENTS

The work described in this thesis has been partially funded by the Spanish Ministry of Science, Innovation and Universities project with reference RTI2018-099777-B-I00, and by the EU Commission under the H2020 research and innovation program under Grant Agreements No. 777517 (EuWireless project) and No. 101016521 (5G-EPICENTRE project).

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

ABSTRACT

Cloud computing technologies success in the last decade has extended its application to the field of cellular networks. As a consequence, new generations of cellular networks base their design and architecture on virtualization technologies fundamentals. One major virtual network characteristic is the presence of an entity in charge of the resource management and orchestration. In the context of virtualized cellular networks, this element becomes a crucial component to be standardized and implemented.

Furthermore, cellular network research has been historically limited due to the demanding requirements to perform realistic mobile experiments. Several initiatives have addressed this limitation by creating experimentation platforms to offer the research community the infrastructure and resources to deploy mobile networks at scale for testing. Nevertheless, these platforms rely on a centralized orchestration system.

Bearing this in mind, this thesis proposal consists of the architectural design of an experimentation platform based on virtualization technologies for 5G and Beyond 5G networks. This thesis follows an incremental approach, so that once the research problem is defined, it is followed by a research on the state of the art of the cellular networks orchestration challenges. Due to the observed constraints of the centralized orchestration model, the proposed architecture distributes its orchestration system. Among the challenges of distributing the orchestration identified, this thesis is focused on the ones related to the resource allocation and sharing, and the dynamic network slices creation and management.

To demonstrate the feasibility of the proposed architectural design, the solution is implemented through the deployment of different experimentation platforms that integrate the design and adapt it to various virtualization and distribution techniques, and the validation of those platforms by deploying temporary experimentation networks on top of them. Specifically, the first platform deployed includes the distributed orchestration and abstracts the resources and services in the form of virtual machines; the second one centralizes the orchestration and follows a container-based approach in which the services provided increase their granularity by dividing themselves into microservices chains; and the third platform combines the distribution of both the orchestration of the resources and the microservices.

The experimentation on the different platforms by means of deploying services oriented to critical communications, which is a characteristic vertical sector of 5G networks, demonstrates the feasibility of

UNIVERSIDAD DE MÁLAGA

the proposed architecture, and the benefits of relying on a container-based distributed deployment for the underlying infrastructure. In conclusion, the distribution required in cellular networks deployment based on virtualization technologies has proven to reside not only in the orchestration systems but in the services provided by the network, since increasing their granularity by creating microservices improves the network performance and adaptability to novel technologies and architectures, making it future-proof.

UNIVERSIDAD DE MÁLAGA

RESUMEN

El éxito de las tecnologías de computación en la nube en los últimos años y su aplicación en sectores como las redes móviles, supone que las nuevas generaciones de redes móviles se fundamenten en los principios de virtualización y la arquitectura típica de este tipo de sistemas. En este contexto, la entidad encargada de la gestión y orquestación de los recursos de la red móvil se convierte en un componente crucial a estandarizar e implementar.

Por otra parte, la experimentación en redes celulares se ha visto limitada históricamente debido a las restricciones para realizar experimentos móviles realistas. Distintas iniciativas han abordado esta limitación, creando plataformas de experimentación para ofrecer a la comunidad investigadora la infraestructura y los recursos necesarios para desplegar redes móviles de prueba a escala. Sin embargo, los sistemas que gestionan los recursos de dichas plataformas se han abordado con una visión centralizada.

Así, en esta tesis se propone el diseño de la arquitectura de una plataforma de experimentación en redes 5G y B5G basada en tecnologías de virtualización. Además, debido a las limitaciones observadas en los modelos de orquestación centralizados, se propone que dicha plataforma distribuya su orquestación. Esta tesis sigue una aproximación incremental, de manera que una vez se ha definido el problema de investigación, se ha realizado un estudio de los retos que supone la orquestación de las redes móviles. Debido a las limitaciones observadas en los modelos de orquestación centralizados, la propuesta de arquitectura distribuye el sistema de orquestación. De entre los retos identificados en los sistemas de orquestación distribuidos, en esta tesis se propone una arquitectura para abordar los relacionados con la asignación de recursos, la compartición de los mismos y la creación y gestión dinámica de *slices* de red.

Tras el diseño de la arquitectura, se procede a la implementación de la solución mediante el despliegue de varias plataformas de experimentación reales que integran el diseño original adaptándolo a diferentes técnicas de virtualización y distribución, así como la validación de las plataformas realizando una serie de despliegues de redes experimentales temporales sobre las mismas. Concretamente, se despliega una primera plataforma con orquestación distribuida que utiliza una abstracción de sus recursos y servicios en forma de máquinas virtuales; una segunda plataforma que centraliza la orquestación y sigue un enfoque en contenedores en el que los servicios desplegados aumentan su granularidad dividiéndose en cadenas de microservicios;

UNIVERSIDAD DE MÁLAGA

y una tercera plataforma que combina la distribución de la orquestación de los recursos con la de los microservicios.

La experimentación sobre las distintas plataformas mediante el despliegue de servicios orientados a las comunicaciones críticas, identificado como sector vertical característico de las redes 5G, demuestra tanto la viabilidad de la arquitectura propuesta como los beneficios del despliegue distribuido basado en contenedores. De esta manera, se puede concluir que la distribución necesaria en el despliegue de redes celulares basadas en tecnologías de virtualización no solo reside en la orquestación, sino en los servicios que la propia red soporta, de modo que aumentando su granularidad mediante la creación de microservicios se mejora el rendimiento de la red y su capacidad de adaptarse a nuevas tecnologías y arquitecturas de cara a diseñar redes duraderas en el tiempo.

UNIVERSIDAD DE MÁLAGA

PUBLICATIONS

Some ideas and figures have appeared previously in the following publications:

- A. Rios, B. Valera-Muros, P. Merino-Gomez, and J. Sobieski, "Expanding GÉANT Testbeds Service to Support Pan-European 5G Network Slices for Research in the EuWireless Project," in Mobile Information Systems, Vol. 2019, Article ID 6249247, doi: 10.1155/2019/6249247 [134]
- B. Valera-Muros and P. Merino-Gomez, "Is GÉANT Testbeds Service compliant with ETSI MANO?," in Proceedings of the 2019 IEEE 2nd 5G World Forum (5GWF), Dresden, Germany, 2019, pp. 502-507, doi: 10.1109/5GWF.2019.8911622 [168]
- I. Harjula, L. Panizo, B. Valera-Muros, J. Pinola, M. Hoppari, and A. Flizikowski, "Dynamic Spectrum Management for European-Wide Research Network," in Proceedings of the 2020 IEEE 91st Vehicular Technology Conference (VTC2020), Antwerp, Belgium, 2020, pp. 1-6, doi: 10.1109/VTC2020-Spring48590.2020.9129017 [83]
- B. Valera-Muros, L. Panizo, A. Rios, and P. Merino-Gomez, "An Architecture for Creating Slices to Experiment on Wireless Networks," in Journal of Network and Systems Management 29, 1 (2021), doi: 10.1007/s10922-020-09571-8 [169]
- G. Margetis, B. Valera-Muros, K. C. Apostolakis, A. Díaz Zayas, L. Panizo, P. Tomás, "Validation of NFV management and orchestration on Kubernetes-based 5G testbed environment," 2022 IEEE Globecom Workshops (GC Wkshps), Rio de Janeiro, Brazil, 2022, pp. 844-849, doi: 10.1109/GCWkshps56602.2022.10008690 [113]
- K. C. Apostolakis, B. Valera-Muros, N. di Pietro, P. Garrido, D. del Teso, M. Kamarianakis, P. Tomás, H. Khalili, L. Panizo, A. Díaz Zayas, A. Protopsaltis, G. Margetis, J. Mangues-Bafalluy, M. Requena-Esteso, A. Gomes, L. Cordeiro, G. Papagiannakis, and C. Stephanidis, "A network application approach towards 5G and beyond critical communications use cases," in Frontiers in Communications and Networks, Vol. 5 - 2024, doi: 10.3389/frcmn.2024.1286660 [23]

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

CONTENTS

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD DE MÁLAGA

LIST OF FIGURES

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD DE MÁLAGA

LIST OF TABLES

UNIVERSIDAD DE MÁLAGA

ACRONYMS

3GPP 3rd Generation Partnership Project

5GC 5G Core

5GPPP 5th Generation Public and Private Partnership

5G NR 5G New Radio

5QI 5G QoS Identifier

AAA Authentication, Authorization and Accounting

AF Application Function

AI Artificial Intelligence

AMF Access and Mobility Management Function

ANM Autonomous Network Management

API Application Programming Interface

APN Access Point Name

AR Augmented Reality

AuC Authentication Center

AUSF Authentication Server Function

AWS Amazon Web Services

B5G Beyond 5G

BBU Base-Band function Unit

BSC Base Station Controller

BSS Base Station Subsystem

BTS Base Transceiver Station

C-RAN Cloud-RAN

CapEx Capital Expenditure

CCAM Connected, Cooperative, and Automated Mobility

CCC Command and Control Center

CI/CD Continuous Integration/Continuous Delivery

UNIVERSIDAD DE MÁLAGA

CIS Container Infrastructure Service

CISM CIS Management

CIR Container Image Registry

CMaaS Connectivity Management as a Service

CMS Configuration Management System

CNCF Cloud-native Computing Foundation

CNF Cloud-native Network Function

CNFVI Cloud-native NFV Infrastructure

CPU Central Processing Unit

CSF Central Server Facility

CU Centralized Unit

DDoS Distributed Denial-of-Service

DevOps Development and Operations

DNS Domain Name System

DoS Denial of Service

DSAF Dynamic Slice Allocation Framework

DSL Domain Specific Language

DU Distributed Unit

EAP Enhanced Authentication Profile

EIR Equipment Identity Register

eMBB enhanced Mobile Broadband

EMS Element Management System

eNB Evolved NodeB

EPC Evolved Packet Core

ES3A Efficient, Secure network-Sliced and Service-oriented Authentication

ETSI European Telecom Standards Institute

GGSN Gateway GPRS Support Node

GMS Group Management System

UNIVERSIDAD DE MÁLAGA

gNB gNodeB

GPRS General Packet Radio Service

GPU Graphical Processing Unit

GSM Global System for Mobile Communications

GSMA GSM Association

GTS GÉANT Testbed Service

GUI Graphical User Interface

GVM Generic Virtualization Model

HetNet Heterogeneous Networks

HLR Home Location Register

HMD Head-Mounted Display

HSDPA High-Speed Downlink Packet Access

HSPA High-Speed Packet Access

HSS Home Subscriber Server

HSUPA High-Speed Uplink Packet Access

IaaS Infrastructure as a Service

ICT Information and Communications Technology

IdMS Identity Management System

ILP Integer Linear Programming

ILS Iterated Local Search

IMSI International Mobile Subscriber Identities

IoT Internet of Things

IP Internet Protocol

K8s Kubernetes

KMS Key Management System

KPI Key Performance Indicator

KPI 2 End-to-End MCPTT Access Time

LDAP Lightweight Directory Access Protocol

LSA Licensed Shared Access

UNIVERSIDAD DE MÁLAGA

LTE Long-Term Evolution

LTE-A LTE-Advanced

MAC Medium Access Control

MANO Management and Orchestration

MCC Mobile Country Code

MCData Mission Critical Data

MCPTT Mission Critical Push-to-Talk

MCVideo Mission Critical Video

MCX Mission Critical Everything

ME Mobile Equipment

MEC Mobile Edge Computing

MILP Mixed-Integer Linear Programming

MIMO Multiple-input Multiple-output

MMaaS Mobility Management as a Service

MME Mobility Management Entity

mMTC massive Machine Type Communication

MNC Mobile Network Code

MNO Mobile Network Operator

MSC Mobile Switching Center

MVNO Mobile Virtual Network Operator

NaaS Network as a Service

NAT Network Address Translation

NetApp Network Application

NEF Network Exposure Function

NF Network Function

NFV Network Functions Virtualization

NFV ISG NFV Industry Specification Group

NFVI NFV Infrastructure

NFVO NFV Orchestrator

UNIVERSIDAD DE MÁLAGA

NMS Network Management System

NRF NF Repository Function

NS Network Service

NSF National Science Foundation

NSI Network Service Interface

NSS Network and Switching Subsystem

NSSF Network Slice Selection Function

NVS Network Virtualization Substrate

OAI OpenAirInterface

OFDM Orthogonal Frequency Division Multiplexing

OpEx Operational Expenditure

OSM Open Source MANO

OSS Operational Support System

OSS/BSS Operational and Business Support Systems

P2P Peer-to-Peer

PCF Policy Control Function

PCRF Policy and Charging Rules Function

PDN Public Data Network

PGW Packet Data Node Gateway

PKI Public-Key Infrastructure

PLMN Public Land Mobile Network

PNF Physical Network Function

PoC Proof-of-Concept

PoDs Points of Distribution

PoP Point of Presence

PPDR Public Protection and Disaster Relief

PSTN Public Switched Telephone Network

QoE Quality of Experience

QoS Quality of Service

UNIVERSIDAD DE MÁLAGA

RA Resource Agent

RBAC Role-Based Access Control

RCA Resource Control Agent

RAN Radio Access Network

RLL Reliable Low Latency

RNC Radio Network Controller

RO Resource Orchestrator

RRH Remote Radio Head

RTT Round-Trip-Time

RU Radio Unit

SA Stand Alone

SD Spectrum Database

SD-RAN Software Defined RAN

SDN Software Defined Networking

SECaaS Security as a Service

SGW Serving Gateway

SGSN Serving GPRS Support Node

SIM Subscriber Identity Module

SLA Service Level Agreement

SM Spectrum Manager

SMF Session Management Function

SO Service Orchestrator

SOC Service-Oriented Computing

SR Spectrum Repository

TaaS Testbed as a Service

TAC Tracking Area Code

TCP Transmission Control Protocol

TDD Time Division Multiplexing

TLS Transport Layer Security

UNIVERSIDAD DE MÁLAGA

TOSCA Topology and Orchestration Specification for Cloud Applications

TURN Traversal Using Relay NAT

UDM Unified Data Management

UDP User Datagram Protocol

UDR Unified Data Repository

UE User Equipment

UMTS Universal Mobile Telecommunications System

UPF User Plane Function

uRLLC ultra-Reliable Low-Latency Communication

UTRAN UMTS Terrestrial Radio Access Network

V2X Vehicle-to-Everything

VA Vertical Application

VANET Vehicular Ad Hoc Network

VCA VNF Configuration and Abstraction

VFC Vehicular Fog Computing

VIM Virtual Infrastructure Manager

VLR Visitor Location Register

VM Virtual Machine

VMM Virtual Machine Manager

vMNO virtual MNO

VNE Virtual Network Embedding

VNF Virtualized Network Function

VNFM VNF Manager

VPN Virtual Private Network

VS Vertical System

xRAN extensible RAN

UNIVERSIDAD DE MÁLAGA

Part I

PRELIMINARIES

This part includes the introduction to the thesis work, detailing the motivation behind the research, and providing an overview of the research outline and document structure.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

1 INTRODUCTION

1.1 MOTIVATION

The great success of cloud computing technologies during the last years has reflected directly on mobile networks [34], which are now adopting the foundations of virtualization in the design of the 5G and Beyond 5G (B5G) networks core architecture.

Moreover, the industrial sectors and application areas that will benefit from adopting 5G technologies have been identified as key verticals in this context, and are categorized according to their distinctive requirements. Among these verticals, mission-critical services reflect a representative case since their criticality hinders their transition to broadband.

To ensure a successful integration of these novel concepts in legacy networks, a proper environment to test and develop new designs and services is required. However, experimentation in wireless networks has been historically limited due to the need for specific equipment and licensed radio spectrum to perform realistic tests outside fixed laboratories. This situation has led to the creation of several experimentation platforms across Europe and the US, which offer the research community the infrastructure and resources required to deploy mobile networks at scale.

Regarding the major experimentation platforms already existing, the European Commission and the US National Science Foundation have sponsored several initiatives to provide the research community with access to commercial equipment. Some of these programs are federation-based, such as the EU's Future Internet Research and Experimentation (FIRE) [53] and the US' Global Environment for Networking Innovation (GENI) project [33][32].

While FIRE has different technology laboratories distributed across Europe and equipped with state-of-the-art components, these laboratories are stationary. As a consequence, there are limitations to research on wireless networks due to the restricted mobility, which has to be simulated [74][73]. The advanced version of this environment is the project 5GinFIRE [152], which is focused on 5G vertical industries and follows the European Telecom Standards Institute (ETSI) Network Functions Virtualization (NFV) architectural reference.

The GENI virtual laboratory counts with Long-Term Evolution (LTE) base stations operating in the licensed Educational Broadband spectrum to deploy mobile edge testbeds on campus [77]. However, it

UNIVERSIDAD DE MÁLAGA

focuses mainly on computer networks and distributed systems research.

The 5th Generation Public and Private Partnership (5GPPP) [131] is another European initiative that collaborates with the Information and Communications Technology (ICT) industry. In 2017, the 5GPPP's Phase 2 program funded 21 projects oriented towards 5G technologies validation. Between these projects, 5Gtango [186] focuses on deploying and validating network applications and services, with Open Source MANO (OSM) as centralized orchestrator. The MATILDA project aims at designing, developing, and orchestrating network services and applications by using a multi-site NFV Orchestrator (NFVO), a multi-site virtualized infrastructure manager, and a multi-site service conductor [161]. On top of them, there is a Global Orchestrator [16] to keep the orchestration centralized. The project SliceNet provides an end-to-end slice management framework for virtualized multi-domain and multi-tenant 5G networks, differentiating service, slice, and resource planes and offering a Cross-Plane Orchestration system [172].

In 2018, the 5GPPP started the Phase 3 program with three infrastructure projects oriented to create 5G end-to-end facilities to research the 5G Key Performance Indicator (KPI) of different vertical industries. The 5G End-to-end Network, Experimentation, System Integration, and Showcasing (5GENESIS) project [98] consists of five testing platforms with the same reference architecture deployed across Europe to interoperate and expose APIs to verticals. The 5G Verticals INNovation Infrastructure (5G-VINNI) project [111] supports two kinds of experimentation facilities, the ones stated on their Service Level Agreement (SLA) to support other projects, and the others to feedback the own project implementation. The 5G European Validation platform for Extensive trials (5G EVE) project [81] comprises four facilities distributed across Europe for experimentation and validation of 5G pilots.

In the US, there are two projects oriented to wireless network experimentation and funded under the umbrella of the PAWR program. The project Cloud Enhanced Open Software Defined Mobile Wireless Testbed for City-Scale Deployment (COSMOS) [184] aims at ultra-high bandwidth and low latency wireless communication to support city-scale real-world experiments. It relies on an edge cloud computing-based architecture and the ORBIT Management Framework (OMF) [132] for the control, measurement, and management tasks. On the other hand, the Platform for Open Wireless Data-driven Experimental Research Reconfigurable Ecosystem for Next-gen End-to-end Wireless (Powder-RENEW) project [55] provides the infrastructure to deploy 5G radio testbeds in a scaled, real-world environment, with the Emulab [64] Control entity at the center of the control layer.

UNIVERSIDAD DE MÁLAGA

In 2019, the US National Science Foundation (NSF) funded the FABRIC research infrastructure to connect PAWR networks. FABRIC is not an isolated testbed, but a combination of Layer 1 core and edge components integrating wireless resources and linked to several facilities to enable beyond 5G networking at-scale experimentation, among others [29].

A summary of the major related experimentation platforms hereby introduced is presented in Table 1, extracted and updated from [169].

In this context, this thesis envisions the creation of a research environment to deploy dedicated 5G networks that span geographically across several locations, and whose resources are managed by an orchestration system distributed across the different points of presence that compose the research environment. The motivation behind this approach is not only the need for a proper infrastructure to test new applications, technologies, and services, but also the similarities found between the concept of creating customized testbeds for experimentation and the network slicing technology, which is a key enabler in 5G networks. Furthermore, distributing the architecture and its orchestration constitutes an alternative to the current centralized approaches available, addressing the principal challenges of centralized systems to facilitate the creation of dedicated networks suitable for the novel 5G key verticals' use cases.

1.2 RESEARCH OUTLINE

In recent years, novel cellular networks' design has advanced towards the integration of cloud computing and virtualization technologies, mainly due to the benefits of automation and resource optimization. However, new challenges have arisen from inheriting virtualization in the mobile communications environment. Additionally, research on cellular networks is limited due to the equipment required to perform realistic experiments. This motivates the creation of different experimentation platforms that address these limitations to offer the research community the resources to deploy mobile experiments at scale.

In this context, the main objective of this thesis is the architectural design of a cellular network for experimentation based on virtualization technologies. Moreover, due to the limitations observed in the already existing experimentation platforms which follow a centralized orchestration model for their resources, this thesis seeks at justifying the need for distributed orchestration systems.

In this section, we present the research questions that provide a framework for the investigation, the contributions resulting from the study, and the methodology followed.

UNIVERSIDAD DE MÁLAGA

Project	Coverage	Mobility	Spectrum	Extensible	Orchestration
FIRE	Fixed labs(Europe)	Per testbed	Per testbed	Per testbed	Independent per testbed
GENI	US Universities	Per project	Educational broadband	Per project	Independent per project
COSMOS	US NY City Sector	V2X	Commercial 5G	Yes	Centralized OMF
Powder	US Salt Lake City Sector	V2X	Commercial 5G	Yes	Centralized Emulab
5Gtango	No	No	No	Yes	Centralized OSM
MATILDA	No	No	No	Yes	Multi-site, centralized
SliceNet	No	No	No	Yes	Multi-domain, cross-plane
5GENESIS	EU City Sectors	Yes	Commercial 4G & 5G	Yes	Centralized, per platform
5G-VINNI	EU City Sectors	Yes	Commercial 4G & 5G	Yes	Centralized, per platform
5G EVE	EU City Sectors	Yes	Commercial 4G & 5G	Yes	Centralized, per platform
FABRIC	US City Sectors	Yes	Commercial 5G	Yes	Centralized, per testbed

Table 1: Comparison of related experimentation platforms

UNIVERSIDAD DE MÁLAGA

1.2.1 *Research questions*

This section provides an overview on the research questions that drive this thesis:

- **Research Question 1 (RQ1):** Which are the main enabling technologies required in an end-to-end experimentation platform to support realistic cellular tests?
- **Research Question 2 (RQ2):** Are centralized functions ready to support the creation of customized and temporary networks as the key enabler to provide service in novel cellular networks?
- **Research Question 3 (RQ3):** Are traditional virtualization technologies (such as virtual machines) enough to meet the requirements of novel cellular networks?
- **Research Question 4 (RQ4):** Which are the main challenges of orchestration systems and what are the benefits and particular challenges of distributing the orchestration?

1.2.2 *Thesis objectives and contributions*

This thesis aims at demonstrating the feasibility of a distributed architecture for 5G networks that integrates distributed services and orchestration. Thus, the objectives that can be outlined are the following:

i The definition of the distributed architecture, considering in the design its components, interfaces, protocols, orchestration, and service provisioning.

ii The evaluation of the architecture designed by using simulations and emulations with real software.

iii The creation of an experimentation platform with the functionality defined and its demonstration by use cases application.

The preliminary hypothesis of this research resides in the improvements that are consequence of distributing the orchestration of resources and services in the context of 5G networks. Thus, the following main contributions are expected:

i The identification of the benefits inherited from virtualizing and distributing the services.

ii The identification of deficiencies in centralized orchestration systems.

UNIVERSIDAD DE MÁLAGA

iii The proposal of a wireless network architecture that integrates a distributed orchestration system to provide distributed 5G services, including the details on the system architecture, and the communication among the entities composing the architecture.

iv The practical evaluation of the proposal feasibility to support 5G networks experimentation by means of the testbed implementation.

1.2.3 *Methodology*

Bearing in mind the objectives and the thesis contributions previously exposed, this research is divided in four phases:

i The first phase consists of the study of the orchestration problem, including the specific challenges that affect the cloud computing platforms, as well as the cellular networks. Additionally, this study includes the state-of-the-art on distributed orchestration applied to mobile networks. This phase aims at identifying the strengths and weaknesses of centralized and distributed orchestration.

ii The second phase includes the proposal to address the challenges identified at centralizing the orchestration of the network and services. This proposal contains the design of an architecture to create distributed mobile networks.

iii To evaluate the design, the third phase includes the prototyping and implementation of the architecture in an experimental environment.

iv The last phase includes the evaluation of the solution implemented by means of applying different use cases and performing experiments identified in the state of the art and the proposal definition.

1.3 DOCUMENT STRUCTURE

This document is organized into five main parts that include seven Chapters. This structure seeks to offer the reader a clear understanding of the research.

i **Preliminaries.** This part includes the introduction to the thesis work.

- **Chapter 1 - Introduction.** This chapter details the motivation behind the research, and provides an overview of the research outline and document structure.

ii **Fundamentals.** This part contains the study of the related literature to establish the thesis contributions' fundamentals.

UNIVERSIDAD DE MÁLAGA

- Chapter 2 - **Background.** This chapter includes a high-level background on mobile networks and virtualization technologies.
- Chapter 3 - **State of the art on orchestration systems.** This chapter presents an overview of the orchestration challenges and current solutions to address them.

iii **Proposal and evaluation.** This part presents the main contributions of the thesis, with the design of a wireless network architecture for distributed services, and the development of the testbeds to evaluate the feasibility of the design proposed.

- Chapter 4 - **Wireless networks architecture for distributed services.** This chapter presents the architectural principles and resulting distributed infrastructure to support the creation of customized mobile networks.
- Chapter 5 - **Proposal evaluation.** This chapter describes the design and development of three different experimentation platforms based on the proposed architecture in order to evaluate the feasibility of the proposal.

iv **Application.** This part contains the research on microservices and network applications to increase the granularity of the services and their distribution, and improve the architecture proposal.

- Chapter 6 - **Increasing distribution with Microservices.** This chapter presents the microservices approach to show how it unlocks the benefits of 5G technology for the verticals operators and end-users.

v **Final remarks.** This part summarizes the thesis, outlines directions for future research, and includes the list of publications and projects associated to this thesis.

- Chapter 7 - **Conclusions and future work.** This chapter comprises the conclusions obtained during the development of the thesis, the open challenges for further investigation, and the publications and projects related to this thesis.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Part II

FUNDAMENTALS

This part contains the high-level background on mobile networks and virtualization technologies, and the state of the art on orchestration systems, to establish the thesis contributions' fundamentals. The published work supporting this part includes *"Is GÉANT Testbed Service compliant with ETSI MANO?"* published in *IEEE 5G World Forum* in 2019 [168].

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

2 BACKGROUND

2.1 BACKGROUND ON MOBILE NETWORKS

When LTE, commonly referred to as the Fourth Generation of mobile networks, or 4G, was presented in 2008 as the evolution of the Universal Mobile Telecommunications System (UMTS) technology [1], not only did it bring an increased bandwidth and data rate but a profound architectural change in the operators' infrastructure. The new standard was based on data packet switching and followed an *all-Internet Protocol (IP)* approach for the communication between all the network's entities, instead of the circuit-based nature of previous generations, in addition to adopting Orthogonal Frequency Division Multiplexing (OFDM) as dominant data transmission technique.

To better understand the architectural evolution, it is required some high-level background on LTE's precedents. Starting with the Global System for Mobile Communications (GSM), it is based on the creation of open interfaces to interconnect network components from different operators integrated into the same network.

GSM's architecture [10] is divided into subsystems that decentralize the management of the network, specifically the User Equipment (UE), the Base Station Subsystem (BSS), the Network and Switching Subsystem (NSS), and the Network Management System (NMS)/Operational Support System (OSS), as depicted in Figure 1.

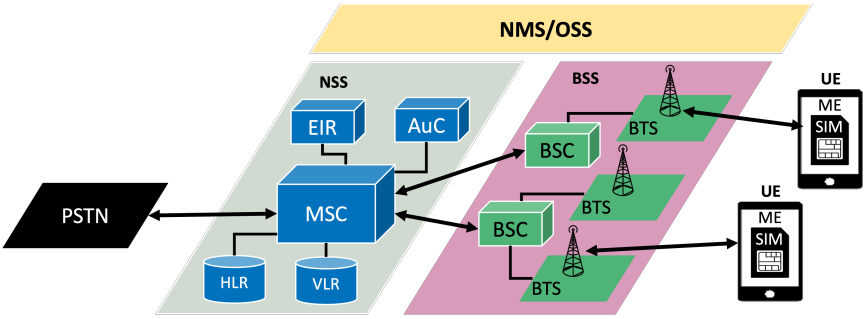

Figure 1: GSM high-level architecture

The UE is the equipment used by the end-user to connect and communicate with the operator's network. It has two main components: the actual physical device (i.e., a phone or a modem) or Mobile Equipment (ME), and one element to identify the user to the network, usually stored in a Subscriber Identity Module (SIM) card provided by the operator.

UNIVERSIDAD DE MÁLAGA

The BSS composes the access network, and sends to the UE information on signaling and processing through the Base Transceiver Station (BTS), which is controlled by the Base Station Controller (BSC). Each BSC controls one or more BTS to handle the frequency distribution among the network users.

The NSS composes the core network and links the end-users with the Public Switched Telephone Network (PSTN). Its main component is the Mobile Switching Center (MSC), which communicates with the Visitor Location Register (VLR), the Home Location Register (HLR), the Authentication Center (AuC), and the Equipment Identity Register (EIR) for authentication, security, billing, and mobility management tasks.

Finally, the NMS/OSS performs the operation and maintenance tasks for the Quality of Service (QoS) management by communicating with the different network elements through each operator's proprietary interface.

To include new multimedia services, GSM evolved to General Packet Radio Service (GPRS) by introducing packet switching through IP. To this end, the Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN) modules are integrated to communicate the BSC with the data network [2], whereas the same radio infrastructure from GSM is maintained, as shown in Figure 2.

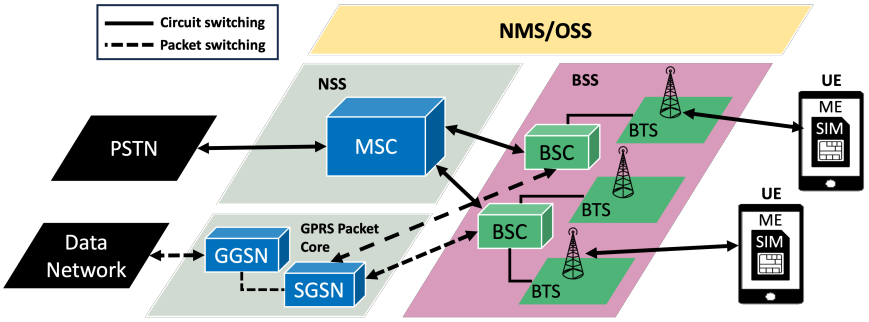

Figure 2: GPRS high-level architecture

The GPRS standard was later replaced by UMTS, presented in Figure 3, which includes a novel radio access network known as UMTS Terrestrial Radio Access Network (UTRAN) to support the users' increasing traffic demands. UMTS, characterized by the convergence of voice and data traffic, improved multimedia features and network bandwidth, being considered the 4G networks closest precedent.

Regarding the architecture, the *NodeB* stations substitute the previous BTS deployment, and connect to the Radio Network Controller (RNC) substituting the BSC, whereas the core network maintains the same structure as the previous generation.

Concerning the technology, the most significant evolution in this standard is the introduction of the High-Speed Packet Access (HSPA), which combines the High-Speed Downlink Packet Access (HSDPA)

UNIVERSIDAD DE MÁLAGA

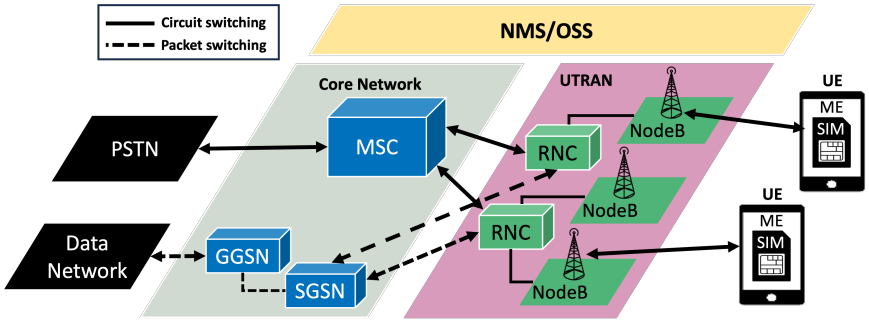

Figure 3: UMTS high-level architecture

and High-Speed Uplink Packet Access (HSUPA) protocols for new modulation schemes and hybrid retransmission mechanisms [137]. The main objective of UMTS was addressing the increase of network bandwidth and data rate.

Bearing these precedents in mind, the 4G architecture also presents three separate domains: the UE [3] used to connect to the network, the radio devices (antennas and base stations) [4] that provide the physical link for the UE to connect to, and the operator's internal packet systems [5] that manage the traffic between the UE and other UEs or external data networks, such as the Internet. The changes introduced in this architecture compared with previous generations apply only to the latest two domains. Figure 4 provides a high-level view of such architecture.

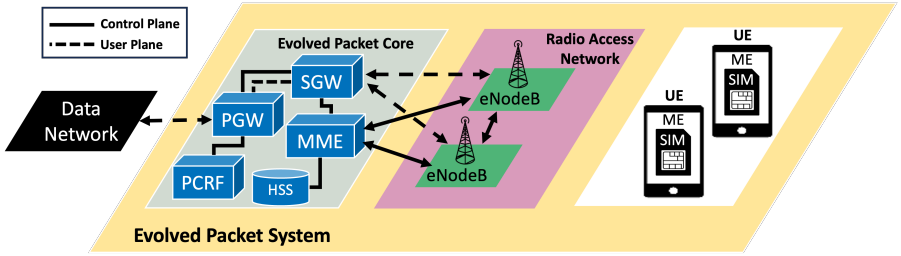

Figure 4: 4G network high-level architecture

The Radio Access Network (RAN) consists of a mesh of base stations, the so-called Evolved NodeB (eNB), which connect the users with the operator's core network. In this environment, the network's intelligence is decentralized, instead of relying on controllers, to ease the connection establishment and reduce handover latencies. The eNBs are interconnected through a dedicated interface, whereas the connection with the core network is divided into two interfaces to differentiate the control plane and data plane traffic, reducing the time needed by the UE to jump between eNBs, and providing a nearly seamless communication when the user moves.

Finally, the Evolved Packet Core (EPC) constitutes the core of the 4G network, and provides much of the intelligence needed to manage it.

UNIVERSIDAD DE MÁLAGA

Aiming at the communications convergence, the EPC relies on the IP protocol, and differentiates the signalling or control plane from the user plane to facilitate network dimensioning.

Instead of using a monolithic architecture, an EPC is composed of several entities tasked with different functions and services. This distribution of tasks maximizes the performance of each component and enables their replication if the need arises. The primary entities inside the EPC are the following:

- The Mobility Management Entity (MME) that controls the connection and the user's mobility between base stations, and maintains a single session for a particular UE between all the entities.
- The Home Subscriber Server (HSS) that ensures the security of the connections maintaining the users' database.
- One or more Serving Gateway (SGW) and Packet Data Node Gateway (PGW) that connect the RAN with the EPC and the EPC with external networks, known as Public Data Network (PDN), respectively.

There are also entities, for instance, to control charging and billing (the Policy and Charging Rules Function (PCRF)), connect to older mobile networks, measure and optimize the handover between base stations, etc.

The Fifth Generation of mobile networks, the so-called 5G networks [7], is an evolution of the 4G architecture and doubles down on the effort of separating functionality into different entities, both at the core and the RAN. It is designed to support a massive number of users with heterogeneous devices, meeting not only a high traffic demand but also QoS requirements of new and complex services.

The 5G network includes three main subsystems similar to LTE networks, which correspond to the evolution of the UE (5G UE), the 5G New Radio (5G NR) composed of new base stations called gNodeB (gNB), and the 5G core network, that comprises the 5G Network Function (NF)s.

A large part of the performance improvement in 5G is achieved by decoupling, even more, the functionality of the operator's core network into new entities. For example, the MME of 4G is split into two different network functions: the Access and Mobility Management Function (AMF), which is responsible for handling connection and mobility tasks, and the Session Management Function (SMF) to coordinate session synchronization between other NFs.

The same happens with the HSS of 4G, as it is now divided into separate functions for user data management (the Unified Data Management (UDM)), authentication procedures (the Authentication Server Function (AUSF)), and a database with user profiles and encryption key management (the Unified Data Repository (UDR)). This decoupling allows each entity to be used by new services independently

UNIVERSIDAD DE MÁLAGA

without the need to parse the complete messages and protocols of the previous architecture.

The complete architecture description and 5G system features are included in the following technical specifications of the 3rd Generation Partnership Project (3GPP): TS 23.501 [7], TS 23.502 [8], and TS 23.503[9]. Besides the previously mentioned entities, the reference service-based architecture, as presented in Figure 5, includes the following entities:

- The Network Slice Selection Function (NSSF) that selects the network slice and the AMF associated to serve a UE.
- The Network Exposure Function (NEF) to safely communicate with with applications externals to the 3GPP network, translating external and internal information and capabilities.
- The NF Repository Function (NRF) for the service discovery, and to maintain the available NF's instances and profiles.
- The Policy Control Function (PCF) that unifies the network policies, containing part of the PCRF's functionality.
- The Application Function (AF) to provide the application services to the UE by handling service-specific aspects and the application-related policies.
- The User Plane Function (UPF) that routes the traffic coming from the access network to the data network to maintain the QoS expected, combining functionality from the SGW and the PGW.

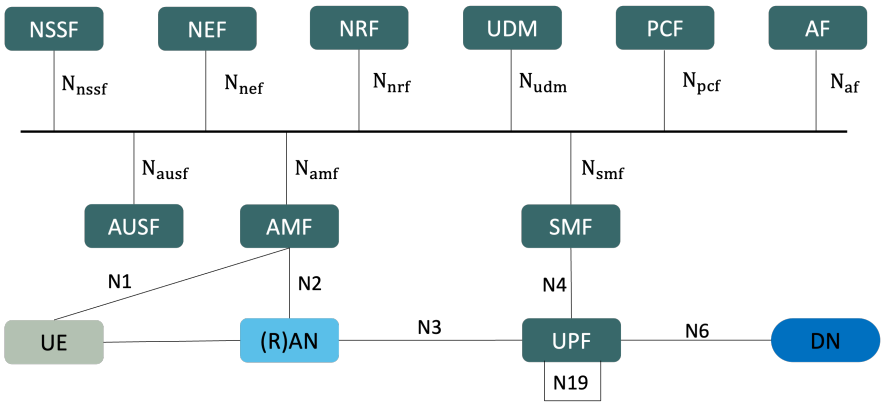

Figure 5: 5G network logical entities

In this context, new architectural design paradigms arise to introduce changes from a network management perspective. Specifically, to support the massive number of connections expected in this mobile generation and meet the new QoS requirements, 5G networks introduce the concept of network slices, which are private and isolated virtual networks on top of a common shared physical infrastructure.

UNIVERSIDAD DE MÁLAGA

The implementation of network slices relies mainly on three technologies: NFV, which allows the virtualization of core and network functions, Software Defined Networking (SDN), which supports the programmability of the network to separate data and control planes into distinct network topologies, and Mobile Edge Computing (MEC), which increases the flexibility of the network by moving part of the core close to the end user. Figure 6 shows the high-level architecture of a 5G network, where the underlying infrastructure integrates the NFV, SDN, and MEC technologies to support the core and transport networks. Thus, a 5G network slice [34, 71] includes exactly the network functions required by a service to achieve the expected performance, using a common infrastructure that can be dynamically configured to put some NF near the end user.

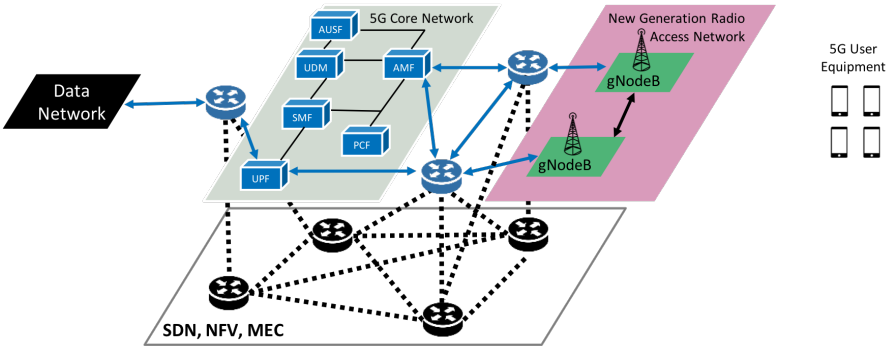

Figure 6: 5G deployment over a SDN architecture

2.1.1 *Network slicing*

The convergence between traditional wired networks and novel mobile ones is a goal stated in the 5G standard, as it is a key enabler to meet a broad variety of 5G use cases, known as verticals, considering each vertical has its particular set of requirements. Since this diversity in the verticals' requirements can lead to an inefficient use of the network, the need for an architectural design able to satisfy them arises.

From a functional perspective, a reasonable approach is operating one dedicated network per vertical, which allows a tailor made implementation and an specific network operation. A more efficient approach relies on a unique platform operating multiple dedicated logical networks. Thus, the sought convergence is known as network slicing, and consists on the adoption of virtualized network paradigms and new schemes of resource sharing so different logical networks operate virtually independently on a shared physical infrastructure efficiently to provide the different verticals' services to the end users.

The network slicing technology aggregates virtual and physical resources, combining radio spectrum, processing capacity, and diverse

UNIVERSIDAD DE MÁLAGA

network capabilities transparently to the service providers and end users, optimizing the network use by customizing each slice according to the specific use case requirements to meet the corresponding SLA. The key concept behind this technology is the complete resource isolation at a logical level, whereas they are being used concurrently at a physical one by all the end users, differentiated by their priority.

From an operator perspective, the slices represent independent end-to-end logical networks running on top of a shared physical infrastructure, that can be provided as a service. Hence, an operator can guarantee certain KPI or QoS by creating and reserving core network instances for a specific group of users sharing the same traffic profile. Those KPI and QoS, as stated by the GSM Association (GSMA) in [71], are defined for each use case based on the vertical specification.

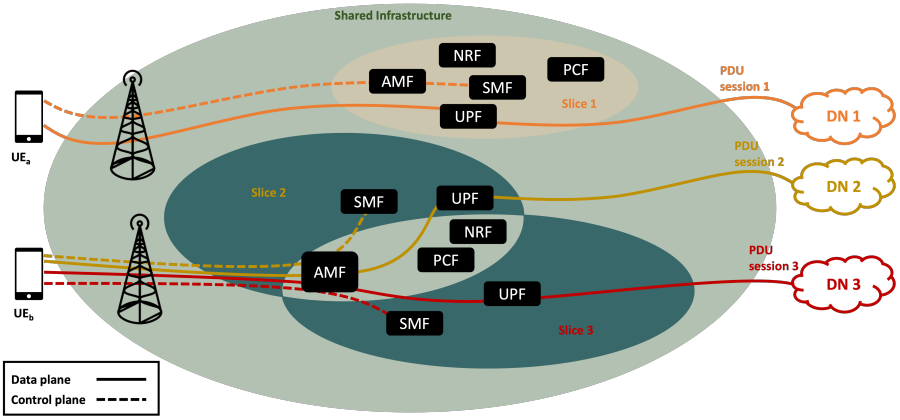

Figure 7: Network slicing in a 5G network

Figure 7 depicts a 5G network deployment relying on the network slicing technology. One of the network slices, numbered as slice 1, represents a dedicated deployment in which every NF supports the same network slice. On the lower part of the figure, UE_b receives service from network slices 2 and 3. Since there is only one access control per UE and one instance for the mobility management for all the services, these two slices share some common NFs, such as the AMF, the PCF, and the NRF, whereas the user plane NFs handling the data services are differentiated into the two slices, so that the slice 2 provides data services for the data network 2, and the slice 3 for the data network 3.

Hence, slices 2 and 3 share the common access and the mobility control, which are applied to all the services provided to the UE. Other than that, both slices are completely independent and isolated, and the same applies to the data services they provide. This allows a fine grained customization of the slices to the services and their requirements. Additionally, separating the data and control plane, and dedicating a tailor made slice per data session, the user experience is similar to having a dedicated physical network.

UNIVERSIDAD DE MÁLAGA

Bearing this in mind, a network operator could, by means of network slicing technology, share their resources with different operators to provide a specific service. Compared to a traditional approach in which each operator has its own resources, and different networks' instantiation involve the replication of those resources to ensure isolation, network slicing enables isolation in a seamless way without the need of duplicating the resources for each use case. Furthermore, the virtualization of these network components facilitates the creation of a large number of dedicated networks to support the multiple novel services arising with the 5G networks establishment.

2.2 BACKGROUND ON VIRTUALIZATION TECHNOLOGIES

The virtualization concept encompasses a great variety of technologies for resource management, providing an abstraction layer between the software and the physical hardware that turns the physical resources into logical or virtual ones. By means of this virtualization, users, applications, and management software are able to operate directly on the abstraction layer, making use of these logical resources regardless of the physical specification of the underlying available resources.

Formally, the virtualization is defined in [158] as *the combination of technologies delivering an abstraction layer between hardware and software to emulate or simulate in software a hardware resource, such as computing, storage or network devices*.

Through virtualization, the costs associated to deploy and maintain a system, called the Capital Expenditure (CapEx) and Operational Expenditure (OpEx), respectively, experiment a notable decrease. This is due to the increase in hardware utilization, the decoupling of functionality from infrastructure, the flexibility in the resources management, and the simplicity of service migration. Focusing on network virtualization, it aims at adapting the already existing networks to the structural changes proposed by new network architectures. Hence, multiple virtual wireless networks can be operated by different service providers that share dynamically the physical substrate of the wireless networks operated by the Mobile Network Operator (MNO). Additionally, the easiness to integrate new services accelerates the migration to new products and technologies, as well as the interoperability among technologies in heterogeneous networks that require convergent management mechanisms [95] [175].

Besides improving network capabilities with higher data rates, lower latencies, and high efficient resource's use, 5G technologies have triggered a full transformation towards customizing the network to address the specific needs of new verticals and services. However, 5G and beyond technologies, as any other advance in broadband communications, can only improve the services provided to the end users if

UNIVERSIDAD DE MÁLAGA

those services are able to evolve and take advantage of the underlying technology.

This leads to the use of virtualization to enable dynamic deployments tailor made to meet each vertical's requirements. In this context, cloud-native solutions and services containerization provides the abstraction from the infrastructure required to easily adapt to new technologies while maintaining compatibility with legacy solutions.

2.2.1 *Virtualized Network Functions*

In [158], NFV is defined as the network function virtualization by means of their software implementation and execution on top of virtual machines. It entails a significative shift in the network design and deployment, since functions such as the *firewall*, Network Address Translation (NAT), Domain Name System (DNS), and *caching*, are decoupled from the proprietary hardware equipment and executed as software distributed across different Virtual Machine (VM)s. The Virtual Machine Manager (VMM), commonly known as the hypervisor, is the virtualization software solution located between the hardware and the VM to deal with the resources, enabling the coexistence of different VMs on top of only one host sharing the resources.

In traditional networks, the network functions act as *black boxes*, which are proprietary platforms in which hardware cannot be shared, so if the system is not working at maximum capacity, the hardware keeps itself idle. In the NFV paradigm, network elements are independent applications deployed flexibly on top of an unified platform that can be composed of one or more servers, switches, routers, or storage devices. In this case, the decoupling between hardware and software allows each application to increase or decrease its capacity adding or releasing virtual resources depending on its needs.

In 2012, the ETSI defined in the white paper [57] the NFV concept, describing the specifications and reference architecture for any NFV platform. This standardization aims at an industrial consensus on the technical and business-related requirements of the NFV technology to establish a common unique infrastructure.

As a result, the NFV Industry Specification Group (NFV ISG) was created including different service providers, technology companies, network components manufacturers, and several telecommunications operators, such as AT&T, BT, Deutsche Telekom, Orange, Telecom Italia, Telefónica, and Verizon. In 2013, the NFV ISG published in [58] the first specifications set, addressing the technical challenges introduced in the previous white paper, and documenting the NFV use cases and architectural framework, among others.

From that point onward, NFV is presented as the solution to satisfy the current and future demands of suppliers, companies, and users of communication networks. The provision of the Infrastructure as

UNIVERSIDAD DE MÁLAGA

a Service (IaaS) and the Network as a Service (NaaS), are reported in [118] as the first use cases of NFV technology, where these Cloud Computing Service Models are mapped as elements within the NFV Infrastructure (NFVI). The following list, extracted from [168], outlines the main requirements of the NFV technology:

- **Portability:** Load, execute and migrate software functions across multi-vendor environments. Optimization of location, reservation, and allocation of resources within the infrastructure. Support and integration of a variety of ecosystems.
- **Performance:** Achieve high performance regardless of the underlying hypervisors or hardware vendors. Describe the infrastructure requirements, instantiate and configure the Virtualized Network Function (VNF) to meet the performance, and data collection related to that performance.
- **Elasticity:** Scale up and down to work concurrently and meet the needs of each function in terms of its traffic demands. Automation of decisions based on parameters and external inputs. Change the location of components while ensuring service continuity.
- **Resiliency:** Assure network stability avoiding single points of failure and returning to normal operation after a failure. VNFs classified into resiliency levels to address the requirements specified in the SLA.
- **Security:** Include countermeasures to reduce the vulnerabilities due to virtualization, network sharing, interconnectivity and isolation. There are different layers to apply security and privileges to control the hierarchy of actors in the system.
- **Service continuity:** Meet the continuity described in the SLA regardless of migration of components and anomalies in the system. Ensuring migration of instances with no impact on the communication among them, even if the instance is not aware of the change in its location.
- **Operation and management:** The VNFs' lifecycle is key. Automation of functions and management of instances based on description models to maintain integrity with the infrastructure and its available resources, and monitorization to prevent misconfiguration failures.
- **Energy efficiency:** Provide on-demand VNFs from a pool of shared resources. Place and move VNFs to distribute the instances through the available working resources instead of available resources in sleep mode to reduce energy consumption.

UNIVERSIDAD DE MÁLAGA

- **Migration and co-existence:** Transition from today's environment to the fully virtualized environment. Co-existence with bespoke hardware-based network platforms to migrate and reuse the network operators' Operational and Business Support Systems (OSS/BSS). Work in hybrid environments with physical and virtual elements.

In the NFV context, the MNO is able to determine which applications and network nodes compose each service, as well as distribute them depending on the system needs. Based on the NFV concept, the VNFs are defined as the blocks used to create end-to-end network services following three main principles:

- **Service chaining:** The VNFs are modular components that provide a limited functionality by itself. Thus, to have a determined traffic flow for an application, the service provider must route that traffic through multiple VNFs to meet the expected network function.
- **Management and Orchestration (MANO):** It refers to the deployment and lifecycle management of the NFV instances, including their creation, service chaining, supervision, redistribution, removal, and billing. The MANO also refers to the management of the elements composing the NFVI.
- **Distributed architecture:** One VNF can be also a composition of different VNF components, each of them implementing a subset of functionality of that VNF. Moreover, each component can be deployed in one or more instances, and these, in turn, can be deployed in different distributed terminals to provide service scalability and redundancy.

All of these characteristics and requirements lay the foundations for any NFV deployment or NFV functional block implementation. Thus, the ETSI describes in [117] the NFV system architecture, including the functional blocks and reference points between such blocks, focusing on the changes to be made in the current network architectures in order to adapt them to the NFV technology. The use of Topology and Orchestration Specification for Cloud Applications (TOSCA) as a declarative method of indicating network service and function requirements is also proposed.

The main functional blocks included in the reference architectural framework, depicted in Figure 8, are the following:

- The NFVI including the storage, network, and computing resources. It is composed of three layers: the underlying hardware components, the virtualization layer that abstracts those components, and the virtual resources provided by that abstraction that are required to support the VNFs that run on top of the infrastructure.

UNIVERSIDAD DE MÁLAGA

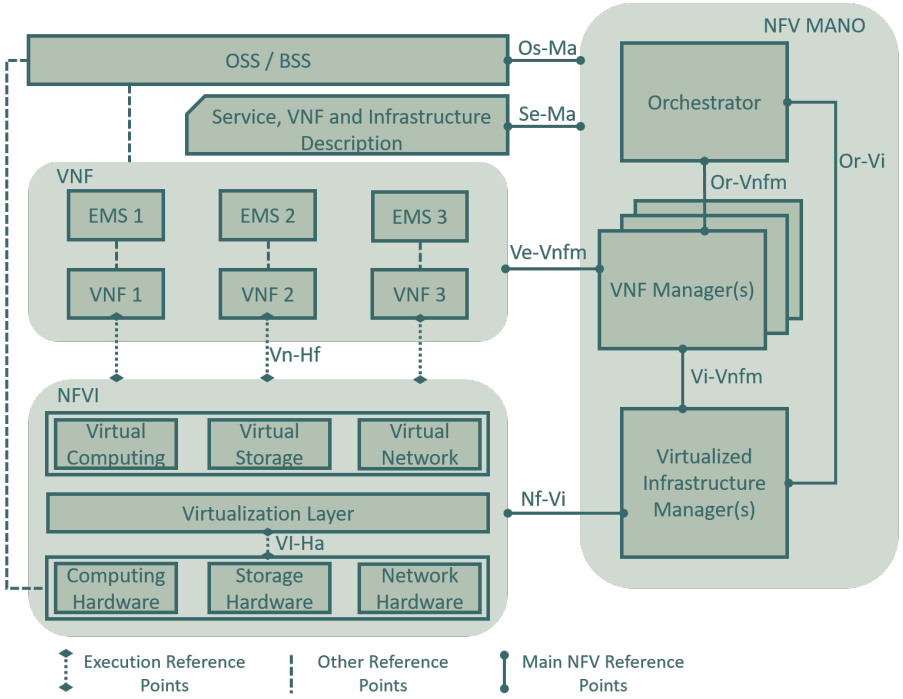

Figure 8: ETSI NFV Architectural Framework. Extracted from [168]

- The VNF and Element Management System (EMS) that run over the NFVI, where the VNF represents the software implementation of a network function, and the EMS handles the VNF's management functionality.
- The NFV MANO is responsible for the management and orchestration of all the NFV environment's resources, VNFs, and their lifecycle. It is divided into three blocks:
  - The Orchestrator is responsible for installing and configuring network services and VNFs, the services lifecycle management, the global resources management, and the validation and authorization of the resources requests coming from the NFVI.
  - The VNF Manager (VNFM) is responsible for the VNF instances lifecycle management.
  - The Virtual Infrastructure Manager (VIM) responsible for controlling and managing the interaction between one VNF and the resources (computing, network, and storage) virtualized and under its control.
- The Service, VNF, and Infrastructure Description provides the MANO information on templates, services, and models.
- The OSS/BSS, which is implemented by the service provider or operator, interfaces with the MANO and enables NFV integration into a multi-vendor environment.

UNIVERSIDAD DE MÁLAGA

Bearing all of this in mind, the NFV characteristics are adopted in the design of new telecommunication networks, and thus, the new network architectures include the NFV architectural framework components. As stated previously, one of these components is the MANO, which logically leads to the need for orchestration of future networks.

However, the reference architecture represents the main constraint in the design of a generalized network orchestrator, since its interfaces must be uniform in any deployment of an NFV system, and each vendor relies on its own interfaces and protocols [136].

In this context, different implementations of the ETSI NFV MANO specification have been developed, but several technical challenges must be overcome to offer the potential benefits identified along with the definition of NFV.

2.2.1.1 *OpenStack as ETSI VIM*

OpenStack[1] is an open source cloud computing platform. It started in 2010 combining NASA's Nebula platform and Rackspace's Cloud Files platform. Due to its success, it has become the de facto standard for open source cloud software deployments. OpenStack is mainly used to deploy IaaS. It is composed of a set projects, each of them handling a service. These projects interact through an Application Programming Interface (API) to integrate and manage the computing, network, and storage resources available in order to ease the on demand provisioning of virtual resources to the users.

The OpenStack openness makes it a cost-effective alternative to proprietary virtualization solutions, such as VMware[2]. It provides not only the management platform, but a fully functional cloud platform that resembles the behaviour of public clouds.

Its modular architecture is based in the following six main projects:

- **Keystone:** To handle identity services. It is responsible for authentication, and management of access points, user's accounts, and roles.
- **Nova:** To handle computing services. It manages the VM-related operations, such as their creation and destruction, or the selection of the computing node to run a specific VM.
- **Neutron:** To handle network services. It provides network connectivity to other OpenStack projects; for example, the project Nova relies on the Neutron API to connect a VM to a determined network segment. Neutron allows the creation of different networks, subnets, routers, firewalls, load balancers, and Virtual

1 https://www.openstack.org
2 https://www.vmware.com

UNIVERSIDAD DE MÁLAGA

Private Network (VPN) by means of an underlying SDN technology that integrates multiple solutions from different external network providers.

- **Glance:** To handle image services. It manages, uploads, and retrieves disk images for the VMs running on the infrastructure.
- **Cinder:** To handle block storage services. It provides permanent storage by provisioning and managing block devices that can be attached to the VMs.
- **Swift:** To handle object storage services. It manages and stores unstructured data as objects accessible through an HTTP-based API. It is highly fault-tolerant due to the combination of an scalable architecture with effective replication mechanisms.

Additionally, there are different components developed for telemetry (Ceilometer project), for dashboard purposes (Horizon project), for shared filesystems (Manila project), for alarming services (Aodh project), and for orchestration purposes (Heat project).

The VMs on OpenStack are called instances, since they are instances of an image created on demand and configured on launching. In traditional virtualization solutions, the state of these instances is considered persistent, whereas in an OpenStack environment both persistent and ephemeral state are possible, which is the main difference with previous virtualization models.

In a persistent state, an instance is created from a volume of persistent storage, such as a device, a file, a block device, a partition, or any other persistent storage form, on top of a computing node. When the instance terminates, any changes performed on the session is kept for future instances.

In contrast, in an ephemeral mode, instances are created from the image service by copying that image into the execution area, so that when the copy is complete the instance starts executing. Instance size and connectivity features are defined in the moment of creation, and when the instance is deleted the original image remains intact, whereas the state of the instance is deleted too, which is useful in situations where a system requires scalation without impacting its users.

2.2.1.2 *ETSI NFV MANO implementations*

There are several open source industry projects that provide an ETSI compliant implementation of the NFV MANO. Among them, OSM and Open Baton are the most extended options, mainly due to the support received from the community.

OSM[3] is the open source implementation of the MANO stack provided by the ETSI in 2014, following the publication of the standard.

3 https://www.osm.etsi.org

UNIVERSIDAD DE MÁLAGA

Its architecture follows an approach based on layering, abstraction, modularity, and simplicity.

Behind the foundation of the OSM project, as stated in [138], there is an operator-led community, which entails an approach focused on meeting the requirements of commercial NFV networks. As a direct consequence, the OSM implementation has not only an evident support from the ETSI and commercial operators community, but also a significant interoperability with different VIMs, which makes it stand out from other available MANO implementations.

The architecture of the OSM solution [86], as depicted in Figure 9, includes a Graphical User Interface (GUI) for the users to access the system. The GUI interfaces with the Service Orchestrator (SO), which receives and processes the packages that contain the Network Service (NS) or VNF description. These packages are available as part of the descriptors catalog at the GUI for the users to employ. The SO is the entity that manages the service lifecycle, and handles the element's control to the corresponding component.

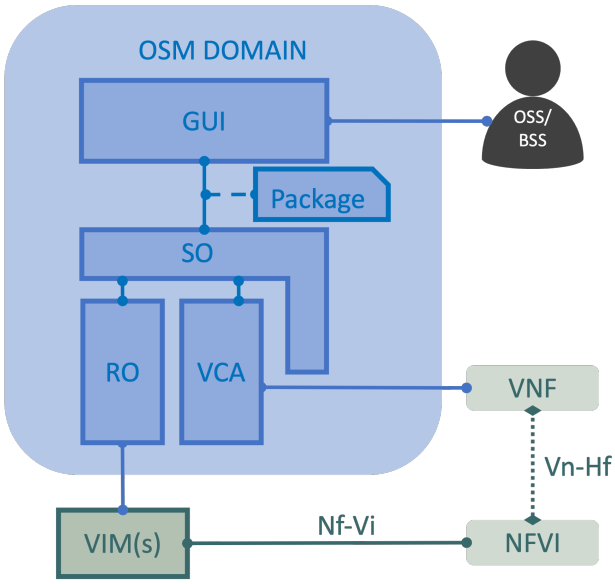

Figure 9: OpenSource MANO Architecture

The requests coming from the SO are processed by the Resource Orchestrator (RO), which operates on the VIM to allocate the resources required and to create the instances expected to meet the descriptors' requirements. On the other hand, the VNF Configuration and Abstraction (VCA) receives the configuration of the VNF descriptors and acts aligned with the VNFM from the ETSI standard, processing any request from the SO once the VNFs are fully functional.

As an alternative, Open Baton[4] is a framework developed by Fraunhofer FOKUS and TU Berlin in 2015, based on the ETSI MANO specification. Similarly to the OSM, the entry point to the system for the

4 https://openbaton.github.io/

UNIVERSIDAD DE MÁLAGA

users is the GUI. Interfacing with the GUI is the NFVO, which manages the lifecycle, QoS, and faults in the system for automatic runtime management of the VNFs, including an OSS component in the block [40].

Open Baton relies on different drivers to interact with different VIMs, and includes a generic VNFM to request the allocation of resources. As in the previous implementation, users describe the NS or VNF via the GUI, and those descriptions are processed by the NFVO, which creates the packages with the description and queues them over a RabbitMQ queue, as shown in Figure 10.

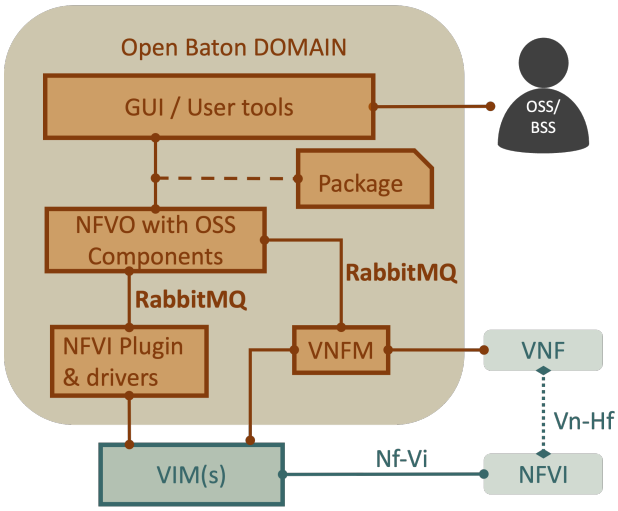

Figure 10: Open Baton Architecture

In [168], we identify the key comparison criteria to evaluate the compliance of any network orchestrator with regards to the ETSI MANO standard. Following our previous work, another orchestrator is proposed to be evaluated as a valid implementation of the ETSI MANO: the GÉANT Testbed Service (GTS).

The GTS was created by GÉANT in 2013, envisioned as a production service to offer Testbed as a Service (TaaS) to the research community. The service provides a tool to create virtual testbeds within a shared infrastructure, relying on dedicated APIs to define with a simple groovy-based language, the Domain Specific Language (DSL), realistic scenarios for experimentation using resources available in the GEÁNT core infrastructure, as well as resources from an external domain [157] [176] [156].

Once the testbed is defined, the allocation of resources is automated by the system and experimenters are given control of the slice created, which runs over the real infrastructure in an isolated manner that prevents collateral effects between experiments, which makes the service a candidate for an ETSI compliant implementation of the MANO. However, since the service is oriented to the research and academic

UNIVERSIDAD DE MÁLAGA

community, and its implementation started prior to ETSI MANO standardization, it is not considered a MANO system per se.

Regarding its architecture, it comprises two main components: the Central Server Facility (CSF), in charge of the service's overall management and operation, and the Points of Distribution (PoDs), which are located across Europe and include the hardware resources ready to be used by the researchers. The service relies on a Generic Virtualization Model (GVM) architecture that manages the resources' lifecycle and hides the internal allocation and provisioning process by means of using Resource Control Agent (RCA).

The first step we considered to determine if GTS is compliant with the ETSI MANO is analyzing its compliance with the NFV requirements defined by the ETSI. The results from the analysis conducted, extracted from [168], are the following:

- **Portability:** The GVM abstracts resources from GTS' own infrastructure and from external domains, to query the resources, reinitialize in case of failure, and optimize their reservation and allocation. This abstraction allows the co-existence of multi-vendor environments.

- **Performance:** Within the GVM, virtualization refers to the definition of the object's abstract behavior so it does not imply a limited performance but actually allows a more efficient use of hardware and the creation of scenarios with native hardware performance.

- **Elasticity:** The GVM is highly scalable and extensible. However, the active modification of a testbed while maintaining consistent topology information is still pending in the service and, at this moment, location cannot be changed while ensuring service continuity.

- **Resiliency, Security, and Service continuity:** GTS is a production service, and thus, it is operational 24/7/365. The isolation of testbeds is assured, users are created with different roles and privileges, and the GVM enables a secure network sharing for all the testbeds in operation.

- **Operation and management:** Resources are scheduled for a specific time window, during which they are available. This allows the provider to bind the availability of the resources to concrete policies to automate management and optimize the resource allocation process.

- **Energy efficiency:** The virtualization model allows the service to share the underlying infrastructure among an ample variety of users and testbeds, scheduling the resources in the efficiently to prevent resources and energy waste.

UNIVERSIDAD DE MÁLAGA

- **Migration and co-existence:** Third party resources are abstracted as external domains. The GVM is extensible and abstracts extra resources by creating new RCAs to manage external infrastructure, allowing the existence of hybrid environments.

The second step to determine the compliance is comparing the similarities in architectural terms of GTS and the ETSI MANO, taking into account that even if the implementation of GTS is prior to the ETSI MANO standardization, both are based on the same virtualization principles.

Considering the whole ETSI reference architecture in comparison with the GTS service, the equivalent to the NFVI, which incorporate the system's computing and network resources, is the combination of GTS' PoDs, containing at least a data plane router with the OpenFlow fabric to provide virtual circuits, the compute nodes to provide VMs, and a switch to connect all of these resources.

Regarding the VIM, GTS is deployed on top of an OpenStack platform, containing among others a Cinder component to host the block storage service for the VIM with the experiment's configuration and data, and a Neutron component combined with an OpenStack controller deployed on the CSF for the OpenStack networking tasks.

Concerning the VNFs, GTS includes a wide variety of resources virtualized available for experimentation in its catalog. The catalog can be extended, as proposed in [134], including new RCAs for VNFs, 5G entities, and resources for wireless experimentation. Thus, compared to the reference architecture, the resources virtualized are the VNFs, and the RCAs act as the EMS, controlling the lifecycle of those resources through the control primitives.

The reference points within the infrastructure and the OpenStack deployment remain the same as in the NFV reference. The GVM allows the communication between the service entities via API. The interface Os-Ma is equivalent to the API that connects the users through the GUI with the rest of the service. There is a specific API between each type of resource and its corresponding RCA.

The MANO functional block, which is the focus of this comparison, includes the descriptors, the orchestrator, and the VNFM, as shown in Figure 11. In the GTS system, the CSF component includes the core, which acts as the system orchestrator, the RCAs to control the resources, and the GUI to access the system.

Mapping this access to the architectural reference, the experimenter in GTS acts as the MNO or service provider that intends to deploy a network slice. To control the RCAs, the core service hosts an RCA Manager, and to handle the virtual circuits, the CSF includes a network component that exchanges information with the data plane router of the PoDs composing the facilities.

The GTS equivalent to the "Service, VNF and Infrastructure Description" is the DSL used by the experimenters to define the testbeds on the GUI. This DSL contains the resources' specific characteristics, and

UNIVERSIDAD DE MÁLAGA

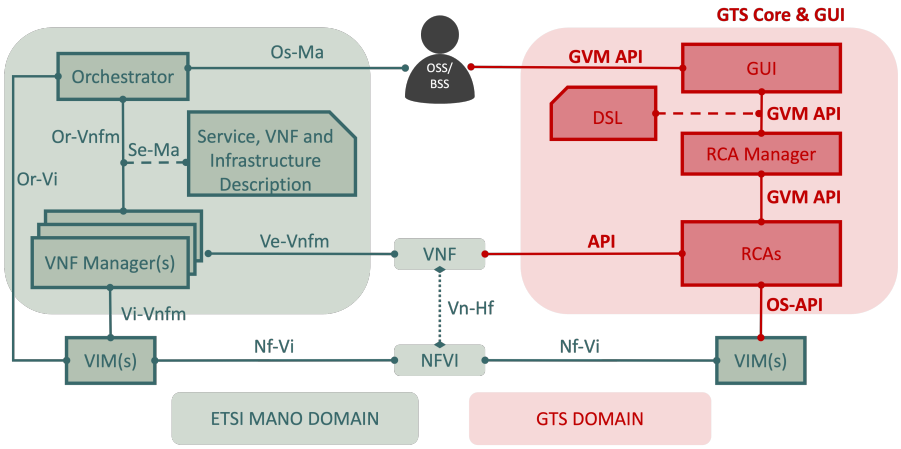

Figure 11: Architectural mapping between ETSI MANO and GTS [168]

the scenario's topology, and the description is used by the RCA Manager to proceed with the creation of the RCAs required, that will then act as VNFM.

Bearing this mapping in mind, GTS or any other orchestrator implementation can be evaluated in the context of the ETSI MANO standard by proving its compliance with the ETSI NFV requirements, and by following the standard in its architectural definition. In the case of GTS, which runs on top of an OpenStack infrastructure, it has proven to be compliant.

Moreover, the system's virtualization model allows the integration of external domains, which could be extensible to the deployment of architectures with multi-domain orchestration, a crucial feature in virtualized 5G networks.

2.2.2 *Containerized Network Functions*

When providers started to substitute the physical network elements, the VNF created for that purpose included every function of the original device, which resulted in the creation of heavy VMs running inefficient virtual applications, that were not optimized and thus, costly in terms of management and maintenance. Hence, taking the VNFs as a first step to a fully virtualized environment, there are limitations in this approach due to that 1:1 VNF to Physical Network Function (PNF) mapping presumption.

Furthermore, the complexity of those VNFs inherited made them unsuitable for cloud environments. The adoption of a common NFVI cloud platform allowed the improvement of these initial function implementation to simplify the deployments, and adopt the NFV technology in the design of edge computing and 5G networks. As a result, a lightweight virtualization was developed, where the VNF evolves to a Cloud-native Network Function (CNF).

UNIVERSIDAD DE MÁLAGA

A CNF is designed and implemented to be executed packaged inside containers instead of VMs, which allows the CNF to access the host's resources and operative system. Hence, containers are defined as units of software application that include the code and dependencies required to run individually and be transferred reliably between different environments and clouds without losing functionality, presenting a faster deployment and lower resource consumption [60].

Moreover, the CNF adoption entails not only the functions' containerization, but also a redesign of said functions, in which their functionality is divided into smaller functions that act as building blocks loosely coupled to easily scale, which reduces complexity in development, implementation, maintenance, and operation.

The evolution of traditional components with the adoption of the different virtualization technologies is reflected in Figure 12, extracted from the official Kubernetes (K8s) documentation[5]. As depicted, in traditional deployments the applications used to run on top of physical servers without any resource boundary on the server, which led to resource allocation issues, and the inability to scale, since separating the applications into different physical servers resulted in costly deployments with resource under-utilization.

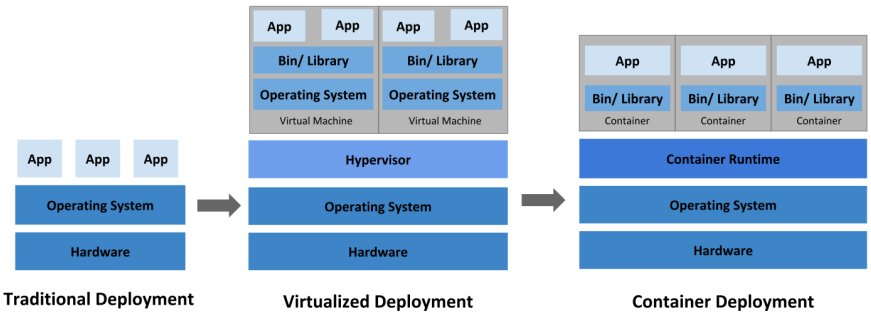

Figure 12: Evolution of deployments with regards to virtualization

In this context, virtualization arises as a solution, where multiple VMs run isolated on top of the same physical server. This technology improves the resource utilization and scalability, reducing hardware costs. However, each VM acts as a whole machine that runs even its own operative system, which results in a heavyweight form of abstraction. In contrast, containers share the operating system with relaxed isolation properties, higher resource utilization efficiency, and increased agility in terms of creation, development, deployment, and integration.

Therefore, containers are presented as the evolution of the VMs, a lightweight virtualization that allows the division of the monolithic stacks into independent services that are deployed and managed dynamically, boosting the level of abstraction from running an operating

5 https://kubernetes.io/docs

UNIVERSIDAD DE MÁLAGA

system on virtual hardware to running an application on an operative system using logical resources.

In this new paradigm, an application or service is composed of several containers that are deployed, scaled, and managed by a container-specific orchestrator that assumes the role of the ETSI MANO in a cloud-native scenario. In 2014, Google[6] open sourced the project Kubernetes, or K8s, which is now hosted by the Cloud-native Computing Foundation (CNCF)[7], and has become the de facto standard for container's orchestration with a rapidly growing ecosystem.

2.2.2.1 *Kubernetes as container orchestration engine*

In [61], the ETSI presented the specification of the NFV architecture adaptation to follow the cloud-native design principles, and rely on container technologies. Particularly, the ETSI mapped the VIM and VNFM architecture and requirements into a container framework to enhance the management and orchestration of CNFs.

However, in [61] the mapping was generic for a containerized environment, which limited the specification exposure. As the K8s framework is recognized as the standard for telecommunications network equipment providers, Amazon Web Services (AWS) in its white paper [13] presents the translation of this abstraction for containerized platforms into K8s terminology.

Summarizing the K8s architecture, a K8s platform includes a cluster composed of a set of nodes, that are the worker machines and a control plane. The worker nodes host the *Pods*, which are the smallest deployable computing unit, and run the containerized applications.

Each *Pod* corresponds to a containerized micro-service, and comprises one or more containers. The control plane manages the *Pods* and nodes, making global decisions about the cluster, and detecting and responding to cluster events. To expose the application running on a set of *Pods*, K8s defines the service objects as an abstraction layer that defines the policy to access that application. These services, as the *Pods*, are controlled as API objects by a set of building primitives.

As TOSCA, K8s also relies on *YAML* representation. However, in contrast to ETSI MANO approach, K8s defines the applications regardless of their function, but as declarative objects understood as deployments. Hence, once an application is defined with regard to the objects that compose it, K8s is responsible for its components interaction and proper running.

The mapping of ETSI MANO into K8s architecture is the following: the VIM would be in charge of placing and scheduling the *Pods* on the worker nodes, that can be bare-metal machines or virtualization platforms themselves; whereas the management of the *Pods*' lifecycle and scaling, which are the main VNFM tasks, are performed by changing

6 https://about.google/
7 https://www.cncf.io

UNIVERSIDAD DE MÁLAGA

the deployment specification to add some constraints in the configuration such as the desired number of replicas for a *Pod*, which K8s will guarantee. The *ConfigMap* type of resource provides the definition for the operation and management of the infrastructure and platform.

The main difference between the ETSI MANO reference and K8s is the northbound exposure, since the VNFM maintains a detailed view of its VNFs, and exposes it northbound to the NFVO, while K8s does not expose its internal workings and placement to the upper layers. In a K8s environment, the operations are controlled through object definitions and primitives that include labels, tags, and selectors.

Thus, instead of using a lifecycle operation granting, as the ETSI MANO, in which the VNFM communicates with the NFVO to ask for permission, in K8s operation the NFVO only specifies the desired final state of operation in declarative ways by defining artifacts, APIs, and manifests, such as deployment and service *YAML* files. Then, the scheduler, along with the constructs, ensure the operation within the specified range, placing the *Pods* and scaling up or down, when required, to efficiently manage the demand on resources.

Regarding the configuration of the VNFs after instantiation, in the ETSI MANO reference it is performed by the VNFM through the EMS. As K8s lacks a built-in mechanism for application configuration, it relies on the *lifecycle hooks*, *init containers*, *ConfigMaps*, and *Operators* to configure the CNFs during and after instantiation.

UNIVERSIDAD DE MÁLAGA

3

STATE OF THE ART ON ORCHESTRATION SYSTEMS

A common feature observed among the experimentation platforms available that motivate this thesis, presented in Chapter 1, is the centralized orchestration of their architecture. However, centralized orchestration imposes diverse constraints for networks, such as scalability issues and limitations in the automation and resiliency of the network.

As stated by Augé and Enguehard [28], this centralization leads to the need for mapping from the users' requests to the configurations. Thus, the need for the orchestrator to have full knowledge and intervene in every operation. They propose to adopt a network protocol to distribute orchestration based on intent-based forwarding. Frick et al. [69] emphasize the issue of having a single point of failure if the centralized NFV orchestrator defined by the ETSI is used.

Distributing the orchestration will optimize the network traffic and lighten the orchestrator's and network links' workload, besides preventing the system from a single point of failure. Nevertheless, the complexity of managing a distributed orchestrator is high, and the intra-orchestration traffic required increases. The basic requirements to distribute the orchestration of a system include a decentralized architecture, a light-weight module for communication and synchronization among distributed orchestrators, the possibility of adding nodes that also include their orchestrator to the topology, as well as the possibility of destroying them in case of failure without compromising the scenario, the continuous resource awareness, and an enhanced security.

Additionally, in mobile networks, there is also a need for wireless resources virtualization and awareness, which complicates the orchestration of the system. Simoens et al. [154] propose a two-layer framework for orchestration of composite services in which the components are distributed across several service nodes. The framework, which was developed within the FUSION[1] project, also handles the instance selection, because the entities from a domain might compose one or more services, and therefore limitations related to the availability of specific resources might show up. FUSION architecture is further detailed in [78].

In this chapter, the challenges of orchestration are presented, classified as challenges on centralized environments, on distributed ones, and common to both architectures as a consequence of the cellular aspect of the resources orchestrated. The researches that compose the

1 http://www.fusion-project.eu

UNIVERSIDAD DE MÁLAGA

state-of-the-art on solutions to address these orchestration challenges are presented in the subsections corresponding to the challenge they focus on, even if some solutions are applicable to more than one challenge.

3.1 CHALLENGES OF CENTRALIZED ORCHESTRATION

Centralized orchestration systems present different challenges, specially when the orchestrated network is distributed. This is the case of mobile networks that span across large territories. The main challenges of centralized orchestration are:

- The **scalability** of the network is fundamental to ensure its survivability when system load is temporally and spatially changing. As a solution, different approaches focus on dynamic slicing and placing of resources, such as [18] and [37]. However, a common strategy in the literature leans towards distribution to face the scalability challenge [41][166][42][167].
- The **automation** of the slice creation is crucial to minimize the human interaction in environments where the number of users and devices is continuously growing, increasing the complexity of the network. To face this automation, different approaches are presented, based mainly in implementing data driven models for intelligent orchestration [178] [163] [44]; machine learning [187] [27] [153]; or the distribution of the orchestration for a complexity reduction [123] [160].
- **Resiliency** is required to deploy highly reliable and available systems, such as 5G networks, where a VNF failure can have a major impact in the whole network. To this end, different mechanisms are proposed, such as the restoration mechanisms in [162], and [14] and [15] not to avoid failure, but to predict and restore the service with the minimal downtime possible; the federated and cooperative orchestration in [46] and [165]; or the algorithms to improve the network survival in [102] and [170].

3.1.1 *Scalability*

Since the traffic in novel and complex networks travels through it while competing for resources during transmission, buffering, and computing, reliability is not sufficient to ensure the requirements from the 5G verticals are met. Thus, in the context of high-rate critical services, scalability arises as a challenge to address in network design and operation.

In [18], Akguel et al. aim at maintaining service guarantees and continuity on a sustainable sharing platform by means of dynamic

UNIVERSIDAD DE MÁLAGA

slicing and trading. They propose a framework that automates network slice adjustment to meet the expected quality per service, and scales rapidly the resources provisioned according to the traffic, conditions, and expectations. The dynamic network slice scaling relies on the tenants trading the unused resources in their slices to reduce expenditure without collateral effects on their working slices.

Buyakar et al. propose in [37] adapting the scaling of a network slice to the type of slice in question, identifying enhanced Mobile Broadband (eMBB) slices when the main requirements are high bandwidth and sustained high capacity, massive Machine Type Communication (mMTC) slices when there is high connection density, and ultra-Reliable Low-Latency Communication (uRLLC) slices when the main constrains are related to latency, reliability, and availability. Thus, the authors include two new components in the orchestrator, the network slicing profiler, and the network slice scaling function, to better predict the slices demands and scale accordingly.

Castellano et al. identify in [41] the edge computing as the enabler to provide the network with flexibility and scalability, and present a distributed assignment and orchestration algorithm for sharing resources from a common edge infrastructure. The authors consider that a centralized orchestrator is unable to address the needs in highly dynamic environments as the edge of the network, and thus, propose distributing the deployments across the infrastructure to better scale the shared resources among the applications running at the edge.

In [166], Toczé and Nadjm-Tehrani also propose a distributed orchestration framework relying on mobile edge devices to address the scalability challenges and provide a high QoS under temporally and spatially changing load. Their framework, called ORCH, aims at serving delay-sensitive traffic in a flexible way when local sudden surges in load appear.

Chekired et al. focus in [42] on the scalability of the SDN core network architecture to provide uRLLC slices for the autonomous driving service in the Vehicle-to-Everything (V2X) communications vertical. Hence, the authors consider distributing the architecture hierarchically across the fog, edge, and cloud, adding four SDN controllers to efficiently modify the forwarding rules for each flow of traffic, maintaining the priority of the critical flows and ensuring scalability and QoS requirements for autonomous driving. Distributing the control of the network over its architecture minimizes the network congestion and message-control overhead.

Tsai et al. also propose in [167] splitting the management functions into different SDN controllers to increase the scalability, stability, and quality of the network. The authors present a cross-domain network slicing system with a multi-controller load balancing mechanism that improves network survivability.

UNIVERSIDAD DE MÁLAGA

3.1.2 *Automation*

With the growing complexity of cellular networks, automation is essential to guarantee the new users and devices are properly served whereas manual interaction is reduced. The softwarization of the network has impacted greatly in its automation, pointing out the need of intelligent models to reduce the complexity of slice creation and service provisioning.

Yang et al. present in [178] the implementation of a full-stack orchestrator called StackV, targeting an end-to-end automation solution for large distributed infrastructures. The implementation is based on a model driven intelligent orchestration approach built to support the full stack of service integration, orchestration, abstraction, and intent and policy representation. Their orchestrator relies on a computation model with pluggable elements to create work-flows to solve an ample set of end-to-end computation, co-scheduling, and automation problems.

In [123], the authors propose an architecture for creating operative network slices following a set of stages to accomplish great agility, flexibility, and full-automation. To this end, the authors propose decomposing the monolithic NFVO into a network slice manager and a network slice orchestrator, so that they differentiate between the resource and network service orchestration blocks. Thus, the authors propose distributing the orchestration to have different abstraction levels across functional blocks placed at different layers to ensure full-automation in the creation of network slices.

There is a considerable number of proposals to address the automation challenge based on Artificial Intelligence (AI), deep learning, and machine learning.

Thantharate et al. present in [163] a model called DeepSlice to make smart decisions selecting the most appropriate network slice. The model is based on a neural network to manage network availability, load balancing, and slice failure by means of deep learning and prediction. The model was trained using the KPIs to analyze the incoming traffic.

AUTODEEPSLICE [44] relies on AI powered data-driven based decisions to implement automatic deep learning and ensure optimal models for automatic slice creation considering load balancing, device KPIs, and failure prediction.

Zhou et al. introduce in [187] a machine learning-based framework for automatic network slice creation oriented to provide service to the Internet of Things (IoT) vertical. The authors propose following a deep reinforcement leaning paradigm to analyze the need for new slices generation, the coordination of resources, and the scaling the already existing slices, so that the network intelligence improves, and slices are created automatically guaranteeing performance and

UNIVERSIDAD DE MÁLAGA

robustness in delay-sensitive and emergency IoT services in smart cities.

Machine learning is also identified by Arzo et al. in [27] as the most promising approach for network automation in the context of NFV and SDN. The authors also consider cloudification of the NFs and containerization as network automation enablers that allow better integration in distributed architectures.

In [153], Simiscuka et al. present a machine learning-based multipath solution called FRADIS to handle the increasing number of devices and users in the network. Their framework focuses on the lower network layers to control the traffic and better adapt the content delivery at the application layer, aiming at maintaining a high service quality in dynamic environments.

In [160], the authors propose distributing the network intelligence to ease the automation of the end-to-end slice creation, so that a cooperative learning approach is applied to model the network. The authors consider that in a heterogeneous multidomain environment, keeping the intelligence centralized complicates end-to-end management and automation. Thus, distributing the orchestration tasks and introducing a broker to model the components in between domains improves connectivity provisioning and reconfiguration of the network when QoS degradation is anticipated.

3.1.3 *Resiliency*

Resiliency is crucial to ensure high reliability and availability in 5G networks. In a cloud environment that runs the 5G NFs virtualized, a VNF failure in the application layer can have a major impact in the network performance, resulting even in network blackout. In [85], Hutchison et al. present their research on resilient networking systems, highlighting the need for dynamically verifiable software defined systems that remain trustworthy despite increasingly autonomous operation.

Taleb et al. propose in [162] a framework with efficient and proactive restoration mechanisms to ensure service resilience in carrier cloud, focusing on the restoration of the VNFs and its impact on the UEs instead of mechanisms to solve the actual VNF failure. They also propose different network overload control mechanisms to minimize the NF restoration impact. Specifically, they focus on the MME restoration, since its failure might result not only in service disruption but also on a storm of signaling messages that could overload the network.

Abhishek et al. identify in [14] redundancy as the approach to provide network resilience at minimal cost. With that objective, the authors propose a self organizing ad hoc network among eNBs from different providers, so when the aggregation network fails, the eNBs

UNIVERSIDAD DE MÁLAGA

form a proactive network with their neighboring eNBs for restoration of the service. In [15], the authors continue the development of this framework by relying on NFV with multiple providers, the use of unlicensed spectrum band and non-terrestrial network, and the self organizing ad hoc network with gNBs. In this paper, the authors identify network virtualization as crucial to ensure resilience in 5G networks. This extension of their work includes a more complete network resilience environment, demonstrating that the use of unlicensed spectrum band and non-terrestrial network adds resilience to the network.

In [102], Lemamou et al. use the mean time between failures to measure the reliability of the network and study its survivability. The authors propose a hybrid Iterated Local Search (ILS) to solve the resiliency issues of LTE wireless networks by planning the set of potential location of network devices to cover a given area ensuring its resiliency. The results of their experiments confirm that an ILS algorithm is able to compute quasi-optimal solutions, improving the results of different Integer Linear Programming (ILP) algorithms also tested.

The need for architectures for end-to-end service orchestration in multi-operator environments is identified by Chung et al. in [46], where the authors recognize a federated distributed orchestration system as the approach to build resilient infrastructures, even thought its adoption by cellular network operators will be slow. Thus, the authors propose a reference architecture for service orchestration called Cell-Orch that leverages software defined infrastructure for agile programmability of cellular 5G networks. Cell-Orch differentiates itself from other orchestration solutions by providing configuration verification mechanisms.

Tipper et al. analyze in [165] the challenges network operators face in terms of backhaul resilience, discussing the impact of moving to commodity hardware, disaggregation of the radio access network, edge computing, densification of the network, and the increased electric power requirements on resilience, together with research directions to overcome this challenges, such as cooperative operator techniques and extending resilient overlays to the wireless edge. Due to its economic and legal implications, the authors identify the approach of extending overlay network techniques to the wireless edge as the most promising solution.

In [170], Vittal et al. build a self-resilient 5G Core (5GC) as a combination of large ILP-based generation and deep learning in a closed loop automation with a self organizing network paradigm. The column generation technique improves the scheduling and serving of control plane user requests in situations where failures of service instances or overload are present. According to their study, this tech-

UNIVERSIDAD DE MÁLAGA

nique combined with AI improves the efficiency in scaling and reconfiguring the slices, and the resiliency and high availability of the 5GC.

3.2 CHALLENGES OF DISTRIBUTED ORCHESTRATION

Distributing the orchestration is the solution proposed to address the challenges of centralized orchestration. Even though this distribution presents several advantages compared to centralized systems, it also entails different challenges that must be addressed:

- Fast and dynamic mechanisms to perform **resource allocation** are needed. Distributed resource allocation is addressed by applying game theory [56]; with mobile traffic forecasting [149]; with a formulation based on ILP [66][54][143][67]; by Virtual Network Embedding (VNE) [31]; using a queuing-based system model [17]; and by auction-based models [87][105][171].

- **Dynamic slice creation and management mechanisms** are required to maximize the number of different services the network can fit. Some of the approaches are the SLA-driven [21]; the creation of protocols to orchestrate the resources dynamically across domains [109]; the data-centric approach [112]; the based on cloud infrastructure management [88]; TOSCA-based approaches [38] [173]; or the creation of a master-agent task scheduler [94].

- Dynamic **resource sharing** optimizes the network usage. A scheduling mechanism is needed to properly allocate the resources shared among slices. Usually, this generates new issues to other challenges such as security and isolation. In the case of mobile networks, the shared resources are not just computational but also radio, which is particularly challenging. There are different mechanisms addressing radio sharing, such as the Network Virtualization Substrate (NVS) [95][110][80][104]. Other approaches aim at acting as Network Slice Brokers for 5G services, such as the blockchain-based [121], or the one proposed by Samdanis et al. [140][75][39]; and others that focus on collaborative cloud computing platforms [151].

- In multi-domain deployments, security concerns are even more complex. Security mechanisms and coordination among domains are required, which was not considered by previous generations' architectures [25]. **Security in distributed scenarios** has been addressed similarly to multi-cloud infrastructures, as proposed in SafeLib [114], in the Security as a Service (SECaaS) of multi-cloud environments [92], and as in the domain-trust models [130]. Roaming scenarios from previous generations has been also studied [108].

UNIVERSIDAD DE MÁLAGA

3.2.1 *Resource allocation algorithms*

With the arrival of 5G networks, new verticals and services have emerged, which need fast and dynamic mechanisms to perform resource allocation and network slice instantiation.

Yigitoglu et al. [182] identify the need for resource-aware allocation models that perform the scheduling and allocation dynamically in large-scale distributed systems. Besides, they consider the scheduling should be workload-aware to provision several services concurrently.

Distributed resource allocation is addressed by D'Oro et al. [56]. Their approach applies game theory to the interaction among servers and users as the mechanism to find a unique solution that is a trade-off between the effectivity and the complexity of a distributed system. Besides exploiting game theory, they propose a reinforcing learning procedure that maintains privacy and distribution. The authors prove the feasibility of applying game theory to allocate and manage resources in this kind of virtualized and distributed environment.

The work of Sciancalepore et al. [149] focuses on mobile traffic forecasting by implementing a model based on the multi-armed bandit problem. The authors aim at the increment of the system's resource utilization thanks to the allocation of resources according to predictions of slice's load.

Fendt et al. [66] formalize the mobile slice embedding in a shared infrastructure as an ILP. In their approach, they analyse the SLAs to add constraints since the slice will have fixed requirements, and sets of slices might not be compatible to share a common infrastructure. Dietrich et al. [54] also propose a Linear Programming formulation to take into account optimality and time complexity in the process of VNF placement in LTE cellular core networks. This approach proves the improvement in terms of load balancing in LTE networks, which also entails better resource utilization.

Following the work of [54], Sattar et al. [143] address the challenge of optimal resource allocation in 5G core networks by extending the linear model to consider intra-slice isolation. This way, it is possible to add reliability and security, and also guarantee end-to-end delay to support real-time services. The formulation used in this case is the Mixed-Integer Linear Programming (MILP), taking the network model and variables from [54].

Ford et al. [67] address the challenge of optimal VNF placing in SDN-based 5G mobile-edge clouds. They also formulate this issue as a MILP problem. However, to simplify the problem for large-scale networks, they propose an optimization algorithm based on the distribution of the VNFs among data centres. They retrieve topology, data traffic and handover statistics as input to several simulations. The results show that their algorithm reduces 75% of redundant capacity while maintaining the same resilience.

UNIVERSIDAD DE MÁLAGA

Baumgartner et al. [31] place VNFs for the mobile virtual core taking into account the cost of placement and the physical network constraints for storage, processing, and switching. The main objective is to minimize the number of occupied links and the use of node resources. Their approach combines linear programming and VNE optimization.

VNF placement and Central Processing Unit (CPU) allocation in 5G networks are addressed by Agarwal et al. [17] by a queuing-based system model that takes latency as the primary KPI to formulate the optimization problem. The allocation process considers the computational capabilities of the physical hosts shared by the VNFs, the requirements of the VNFs of each vertical, and the additional system capabilities available that could be used by the VNFs, resulting altogether in a flexible CPU allocation.

Jiang et al. [87] propose an auction-based model that allocates resources providing an increased revenue while satisfying the requirements of each slice. The model satisfies the resource requirements of the 5G slices optimally, and determines different prices of network chunks to apply the auction mechanism, obtaining the maximum revenue for the network considering the demand for resources.

Liang et al. [105] also propose an auction-based resource allocation for service-oriented slicing. To do so, they create a method to tailor the bidding information to users, and then they address the allocation challenge as an online winner determination problem. The concepts considered in this approach are the allocation of customized resources to maximize the user Quality of Experience (QoE) for each service, the use of dynamic pricing to maximize the economic efficiency, and the order in which the users requested the resources to provide them successively.

Wang et al. [171] suggest the idea of resource pricing to improve the revenue of the network and minimize the demand. As a result, they maximize the resource efficiency of the system. Based on this idea, they model the slice, dimensioning in networks where several slices coexist, and formulate a low-complexity distributed algorithm to address this issue.

3.2.2 *Dynamic slice creation and management*

The slice creation includes the optimal allocation of resources to maximize the number of different services the network can fit. These slices should be able to dynamically scale up and down to guarantee the SLAs and operational constraints of each service regardless of the service load. Moreover, the slice might need partial permissions to manage some resources itself in case the service quality decreases.

Antonescu et al. [21] introduce an architecture to dynamically orchestrate distributed cloud-based software while guaranteeing the

UNIVERSIDAD DE MÁLAGA

SLAs and operational constraints of each application. Their approach consists of scaling the number of VMs according to the service load and the SLA. This SLA-driven model simplifies the validation of rules to map applications to the infrastructure, enhancing the level of automation of the infrastructure management, which includes provisioning, deployment, monitoring, and problem specification and resolution, taking into account the constraints of each resource.

Liu and Han [109] address the distributed cross-domain resource orchestration for dynamic network slicing in cellular edge computing by implementing the protocol so-called DIRECT that also maintains performance and isolation. DIRECT includes a learning-assisted optimization algorithm and it has been validated in a small-scale system prototype based on OpenAirInterface (OAI) LTE and network simulations.

Liu et al. [107] describe a data-centric approach to cloud orchestration by modelling the resources as data structures, which are then queried and updated using transactional semantics: views, constraints, actions, stored procedures and transactions.

Ranjan et al. [133] differentiate the resource orchestration tasks to be performed at design and runtime, such as selecting resources and deploying them, and only at runtime, such as monitoring and controlling those resources. As Liu et. al, they advocate for declarative programming languages to ensure orchestration in environments with high-level of concurrency and network traffic.

In [38], Caballer et al. introduce the INDIGO-DataCloud project, which focuses on the orchestration of heterogeneous clouds by using the TOSCA standard to intercommunicate different cloud management platforms. TOSCA exploits the use of service templates to describe cloud application architectures, with their components and relations among them, providing portability and automating the management irrespective of the underlying infrastructure. This approach is similar to the proposal of the GSMA of using a Generic Network Slicing template [72].

In line with [38], Wettinger et al. propose in [173] a modular deployment methodology oriented to middleware components. The authors aim at an abstraction of the underlying infrastructure to create generic deployment plans and automate application deployment across platforms. Thus, the middleware components can be hosted and reused in any infrastructure, making the cloud environment modular and extensible, and decreasing the number of deployment plans required when deploying an application.

Juve and Deelman [88] present the Wrangler system for cloud infrastructure management and automatic deployment of distributed applications. The system has an similar to ours, further explained in , with agents and plugins deployed on the nodes and a coordinator acting as a broker between the clients and the agents. However, Wran-

UNIVERSIDAD DE MÁLAGA

gler only focuses on the initial distributed application deployment, and it does not consider the impact and provisioning of network resources. In contrast, our proposal considers also dynamic scaling of virtual resources as a response to adaptation triggers.

Kukkalli et al. [100] focus on run-time modifications and resource scaling to provide dynamic end-to-end slices for medical emergency multi-operator scenarios. To address the lack of cooperation among operators, the authors propose vendor independent RAN APIs, and an open source orchestrator to enable the use of multi-operator and multi-vendor resources. Additionally, they identify the fixed configurations for the networks as the main constraint to dynamically create and manage 5G slices since resources tend to be under- or over-provisioned. Thus, their approach includes real-time allocation and deallocation of resources on demand, either creating the slice if it does not exist when the request is received, or updating the existing slice to meet the new requirements of the network.

Kim et al. [94] describe a master-agent task scheduler based on the CometCloud management system, able to provision and size virtual infrastructures composed of VMs in hybrid infrastructure environments. Their system is able to autonomously operate and recover from node failures. However, they do not address SLA guarantees, or how the network is treated as a managed resource.

Sattar and Matrawy [145] use their previous work on optimization-based allocation [143] to dynamically allocate slices and to defend against the malicious co-residency. Hence, the authors proposed the Dynamic Slice Allocation Framework (DSAF) to dynamically allocate slices in real-time, which also provides on-demand intra-slice isolation as a proactive defense against malicious co-residency. In DSAF, only the slice allocation request requires only user interaction, whereas the rest is automated. In addition, it implements their optimization model to fulfill the requirements of the 5G mobile core network.

3.2.3 *Resource sharing*

The main benefit of dynamic resource sharing is the optimization of the network usage. However, the allocation of resources among the slices requires a specific scheduling mechanism, and yields issues to other orchestration challenges, mainly related to isolation and security. In addition to the computational resources, the radio resources are also shared in the context of mobile networks. Some of the scheduling mechanisms introduced in this section focus on radio sharing, whereas others spotlight the network slice brokers for 5G services.

Starting with the radio sharing mechanisms, Kokku et al. [95] present the design and implementation of a NVS to virtualize wireless re-

UNIVERSIDAD DE MÁLAGA

sources in cellular networks. This NVS is oriented to the cellular base station equipment, and it is the first detailed design, implementation, and evaluation of flow-level virtualization of wireless resources on base stations. The solution was tested on a *WiMAX* network with satisfactory results. The authors claim that the NVScan be adapted to cellular technologies with similar characteristics, such as LTE.

Continuing with that research, Mahindra et al. present in [110] the design and implementation of a network-wide radio resource management framework called *NetShare*, that provides effective RAN sharing and targets the network operators interested in RAN sharing to improve their coverage and capacity at an effective cost. *NetShare* includes a scheduler divided into two-levels between the mobile gateway and the cellular base stations to allocate and manage wireless resources shared among different entities while maintaining isolation. *NetShare* was simulated in a LTE-based system, demonstrating spectrum sharing keeps both isolation and efficient distribution of the shared resources among the entities.

In [80], Guo and Arnott introduce a partial resource reservation scheme for LTE networks as a flexible active RAN sharing technique. In this approach, each operator is guaranteed a minimum share of resources whilst also having access to shared common resources on a *first-come-first-served* basis. The partial resource reservation is in charge of both scheduler and admission control aspects. Compared to complete sharing schemes, partial sharing ensures a minimum guaranteed performance for each operator, whereas compared to the full reservation schemes, it allocates the resources based on the actual traffic loads and priorities, which increases the spectrum utilization.

Regarding the network slice brokers, in [121], Nour et al. include a blockchain-based broker in the network slicing process. The broker secures and ensures anonymous transactions between the vertical service provider and the resource provider. The authors aim at building end-to-end network slices securely by using resources from different stakeholders. To do that, the slice provider publishes in the blockchain a request for resources and evaluates the offers received in terms of cost and capability to meet the expected performance, understanding the resource allocation as a series of small contracts with unique identifiers.

Samdanis et al. [140] also analyze the network slice broker to enable the 5G system's actors to request and lease resources from infrastructure providers dynamically via signaling means. In their proposal, the broker is logically centralized, and it monitors and controls incoming requests and resource assignments by means of an enhancement of the 3GPP network sharing management architecture interfaces and service exposure capability function in the context of on-demand multi-tenant networks.

UNIVERSIDAD DE MÁLAGA

Capone et al. [39] present an scenario in which network resources, both at the wireless access and the core network, are traded dynamically on a real-time market, where virtual operators or tenants compete to obtain the resources to serve their users. In their approach, they include real-time resource negotiation, the possibility of service differentiation, the capability of handling service heterogeneity, and enablers for future network expansion.

Shen and Liu introduce in [151] a collaborative cloud computing platform called *Harmony*, which integrates multi-faceted resource/reputation management, multi-QoS-oriented resource selection, and price-assisted resource/reputation control. The collaborative cloud computing platforms interconnect physical resources to enable resource sharing between clouds, so the resource and reputation management is crucial. Thus, the authors identify and address three main tasks: efficiently locating the required (and trustworthy) resources, choosing the resources, and fully utilizing them while avoiding node overload. Their experimental results on Harmony also show a high scalability, balanced load distribution, locality-awareness, and dynamism-resilience in the large-scale and dynamic collaborative cloud computing environment.

3.2.4 *Security in distributed scenarios*

The challenges of security in 5G networks are consequence of the dynamic environment and criticality of the verticals' services. Also, security architectures for previous generations do not consider multi-tenancy operation, so they have no procedure to verify the relation among tenants.

In [25], Arfaoui et al. analyze the security of the architectures designed for previous mobile network generations, together with the 5G networks' use cases. Their objective is to determine the design objectives of a 5G security architecture, describe the components that compose their architecture, and finally demonstrate its applicability with a smart city IoT use case. To better differentiate security countermeasures, authors divide the architecture in security realms.

In 5G networks, where the underlying infrastructure is virtualized and shared among slices, an attack against the resources of one service might affect other services that share those same resources. The work presented in [144] relies on slice isolation to mitigate Distributed Denial-of-Service (DDoS) attacks. The authors analyze inter-slice and intra-slice isolation, aiming at achieving efficient use of system resources while guaranteeing the isolation and end-to-end delay.

The 4G network standards lay the ground for a standardized 5G security architecture. Consequently, vulnerabilities found in LTE networks must be overcome in the design of an architecture supporting 5G network slicing. In mobile networks, the Enhanced Authentication

UNIVERSIDAD DE MÁLAGA

Profile (EAP) framework is responsible for authentication between the user and the core network. However, user privacy is not guaranteed, and the approach of replacing the user identity with an anonymous tag would prevent identity authentication.

In traditional mobile networks, this issue is addressed in roaming scenarios by using anonymous authentication protocols, as stated in [108]. Roaming scenarios are the starting point in the design of a distributed orchestrator since each platform with its orchestrator could be considered a single operator with its administrative domain, and the request of resources from other platforms could mirror a roaming user requesting resources from another operator.

Besides the security threats inherited from 4G networks, we must consider the greediness of 5G users themselves when using distributed resources to ensure legitimate access to services when fog nodes are present. Additionally, fog nodes and IoT service providers tend to behave unfaithfully in terms of user privacy. In [119], the authors present an Efficient, Secure network-Sliced and Service-oriented Authentication (ES3A) framework to ensure private slice selection and anonymous service-oriented authentication. As a result, subscribers will be able to access IoT services anonymously through a group signature, and the types of slices or services are not exposed to the nodes during the slice selection.

SECaaS in multi-cloud environments is the main concept presented in [92]. Khettab et al. addressed the security challenges of network slicing through NFV and SDN technologies. SECaaS is provided with an auto-scaling algorithm based on intelligent security. After the algorithm's performance evaluation, the authors conclude that a VNF's malfunction could jeopardize the security of the whole system. On the other hand, benefits of using SDN to deploy inter-slice isolation and intra-slice traffic control are proved.

Pustchi et al. [130] propose a cross-cloud domain-trust model to share resources across domains from different clouds. Their approach consists of an extension of the OpenStack identity and federation services since OpenStack is the "de facto" platform for offering IaaS.

The design of SafeLib is introduced by Marku et al. in [114]. SafeLib is a middlebox platform to achieve high performance while protecting user traffic, VNF code, policy input, and state. The motivation of SafeLib relies on the cooperation among network operators to provide slices that meet the requirements of 5G verticals, which should be transparent to the final user in a way that the user requests a service from a single operator, and this operator will be responsible for outsourcing the VNFs if needed.

UNIVERSIDAD DE MÁLAGA

3.3 CHALLENGES OF MOBILE NETWORKS ORCHESTRATION

The following challenges derive from the mobile aspect and, thus, are mutual for both centralized and distributed orchestration of networks [103]:

- Most of the work on virtualized networks focuses on the core network. Nevertheless, wireless links vary over time and might suffer interferences, which prevents them from being virtualized by the methods normally applied to wired resources, and thus require specific **wireless resource virtualization**. There are several proposals to support the abstraction of wireless resources, such as the implementation and design of the NVS [95] [110][80][104]; the FlexRAN protocol based on OAI [68][99]; the SoftRAN control plane [79]; the OpenRAN architecture [91]; the extensible RAN (xRAN) initiative [189]; the Connectivity Management as a Service (CMaaS) architecture [179]; and the Cloud-RAN (C-RAN) approach [174].

- **Isolation** in network slicing is key to guarantee security, otherwise any attack or failure affecting one slice could impact on the whole network. Mobile networks add the challenge of resource isolation in RAN. Among the solutions to provide isolation, we can distinguish RAN isolation [120]; the use of private infrastructure [148]; hardware isolation [180]; protocols for securing isolation [142]; control-plane isolation [30]; and methods to perform network slicing providing isolation [90], quantifying the level of isolation achieved [96], and provide isolation reducing the resource utilization inefficiency it usually involves [183].

- The automotive industry is one of the primary verticals of 5G and beyond networks, which entails the need for **mobility support and management**. Also, real-time services require fast and seamless mobility handover. These challenges have been addressed from a centralized point of view [150] [124], and from a distributed one [76] [19] [155]. Additionally, there is research focused on vehicular networks and fog computing to address the mobility support challenge [84] [185] [52].

- Network slicing is a key enabler in 5G and beyond networks and thus, **security in network slicing** orchestration is a major concern, regardless of distribution. Research on security in network slicing, can be classified in security in the physical layer [177]; slice isolation [144][36]; intra-slice communication [142]; and micro-slicing [36][128][35].

UNIVERSIDAD DE MÁLAGA

3.3.1 *Wireless resources virtualization*

Due to the nature of the resources, the mechanisms traditionally applied to virtualize wired networks are not suitable for wireless spectrum components. This leads to the design of specific mechanisms that allow RAN slicing in 5G networks. These mechanisms have to fulfill the capacity and QoS requirements of each tenant, usually specified in the SLAs. In addition, the coordination of these mechanisms among cells must ensure that these requirements are met across the whole RAN, in spite of the non-homogeneously spatial distribution of the traffic [139]. In this context, the Software Defined RAN (SD-RAN) abstracts the underlying RAN resources to allow the Service Orchestrator to dynamically manage the resources that compose a network slice. There are several proposals to support the abstraction of wireless resources. An introduction to the most relevant solutions identified is presented below.

Kokku et al. [95] introduced the concept of NVS to flexibly allocating shared resources. To do so, the NVS modifies the Medium Access Control (MAC) scheduler according to the SLA and the traffic needs of the Mobile Virtual Network Operator (MVNO). The design and implementation of this NVS take into account the requirements in terms of the level of isolation and the slice provisioning. *NetShare* [110] is an updated implementation of the NVS. Guo and Arnott proposed in [80] the extension of the NVS to perform partial resource reservation. The NVS solution might also be considered to address the challenge of radio resource sharing. In the survey on wireless network virtualization [104], Liang and Yu consider the NVS as a viable solution to virtualize cellular networks, and not only WiMax networks as proposed in the evaluation of the initial implementation.

Based on the OAI, the FlexRAN protocol implements another SD-RAN architecture solution. FlexRAN [68] provides abstraction through an open API. Besides, it has a southbound API that maps the third parties' instructions to the corresponding instruction of the OAI eNB. The implementation includes a master controller and a set of agents that receive requests from the master and perform the time-critical control functions. In [99], the authors propose sharing RAN resources among slices by using a two-level MAC scheduler that they implemented based on OAI and FlexRAN.

SoftRAN is introduced in [79] as a centralized control plane for radio access networks. SoftRAN abstracts the base stations located in a specific geographical area to manage dense network deployments. In this way, time-critical functions are distributed at the base stations, whereas the rest of the control plane is centralized and separated from the data plane.

The approach of OpenRAN [91] lies in the creation of a new architecture with an underlying transport fabric consisting of a routed

UNIVERSIDAD DE MÁLAGA

IP network. Thus, OpenRAN decomposes the RAN functionality to allow distributed implementations of processing models.

The xRAN [189] initiative aims at decoupling user and control planes to address programmability of dense wireless networks. The initiative consists of an industrial consortium created to build a reference system for LTE-Advanced (LTE-A) and, eventually, for 5G networks.

The CMaaS, introduced in [179], proposes a new architecture hierarchically layered in terms of time criticality. The lower layer would then be the UE controller, followed by the layer with the base station controller, the RAN controller layer and, on the top, the network controller that instructs the lower controllers.

The C-RAN approach was defined in [174] as a service-oriented scheme for resource scheduling and management. The logical architecture of C-RAN separates the physical, control, and service planes, improving the centralized processing efficiency of the system. C-RAN applies to several RAN scenarios, from macrocells to femtocells. Besides, the authors proposed a user scheduling algorithm and a parallel optimum precoding scheme to increase the performance of their approach. In 4G, C-RAN translates into centralized Base-Band function Unit (BBU) and a pool of Remote Radio Head (RRH) distributed across the cell sites. The 3GPP proposes for 5G the division into the Radio Unit (RU), Distributed Unit (DU), and Centralized Unit (CU). Recently, C-RAN architecture has been widely investigated, and this 3-layer RAN architecture outperforms the previous 2-layer solution [183].

3.3.2 *Isolation*

Isolation in network slicing is crucial so that the traffic congestion and faults from one slice do not interfere in other slices' performance. Besides, isolation ensures that the requirements of each 5G specific service are met to guarantee the QoS at each slice. Networks present different kinds of isolation: traffic isolation, physical isolation, and control isolation [82]. Mobile networks, additionally, present the challenge of resource isolation in RAN.

Nojima et al. [120] introduce two methods to briefly modify the scheduling algorithm responsible for allocating RAN resources in mobile networks. The first includes in the algorithm resource isolation in terms of resource block allocation, whereas the second improves the throughput performance by taking into consideration the channel conditions for the dynamic resource block allocation.

Kotulski et al. [96] state that isolation of slices crucial for the provision of security in 5G networks. In this work, the authors propose a graph-based model to quantitatively determine the isolation level of a slice with a layered structure, being the lowest layer the virtual network elements and the top one the end-to-end slice.

UNIVERSIDAD DE MÁLAGA

Schneider et al. [148] present two different models to offer isolation to the network slices tenants. Their first option proposed is to achieve isolation by means of over-the-top security, which does not provide resource isolation per se, but security to the tenant traffic even against the MNO. The second approach is relying on private infrastructure taking into consideration different use cases depending on the vertical setting the slice, such as a whole private 5G network, a private network on-site with roaming for off-site coverage, a private network with public RAN slice, and a private RAN with a 5G gateway core network. They conclude that, although security can be assured, achieving the highest level of isolation is not possible in shared infrastructures.

Ye et al. [180] address the hardware isolation to extend the isolation to the code that accesses the protected data to avoid leakage of sensitive information. To do so, they propose a framework called EvoIsolator that applies evolutionary algorithms in two phases: the first to create and synthesize a secure slice, and the second to optimize the code to reduce communication overhead.

As stated in the security-related section, the protocol IMAKE-GA designed by Sathi et al. [142] provides a secure key establishment among the slice component pairs involved in the communication for secure slice isolation.

Control-plane isolation is addressed by Basta et al. [30] with HyperFlex, an SDN architecture with hypervisor function allocation to ensure isolated control-plane slices and prevent resources from exhaustion.

Kasgari and Saad [90] propose a framework to perform 5G network slicing with effective isolation by relying on the infrastructure provider. In their approach, the scenario has a time-varying number of users, and they consider two types of slices: the self-managed that provide certain capacity, and the Reliable Low Latency (RLL) slices that ensure a determined end-to-end delay and reliability. The authors minimize power consumption and preserve isolation by modeling the problem with stochastic optimization, and then solve the network slicing problem applying the control framework proposed, obtaining the expected isolation and meeting the end-to-end requirements.

Yu et al. propose in [183] a solution for isolation-aware RAN slice mapping to address the resource utilization inefficiency consequence of inter-slice isolation. Their approach, which focuses on the 3-layer C-RAN architecture, presents a heuristic algorithm for the RAN functions placement and the routing and wavelength assignment of traffic flows between those functions.

UNIVERSIDAD DE MÁLAGA

3.3.3 *Mobility management*

One of the primary verticals supported by 5G and beyond networks is the automotive industry. While mobility support and management might be optional to verticals such as the Factories-of-the-Future, which control fixed devices, it is crucial to the industries requiring mobility of their users. Furthermore, mobility requirements of different verticals also differ, making it essential to design service-oriented mobility management procedures to address the challenges of each dedicated slice.

As highlighted by Yegin et al. in [181], the current and past centralized approaches to orchestrate mobility in 5G networks present efficiency issues, such as increased delay in the end-to-end transmissions, overloading of the core network, and decreased network reliability. In contrast, the authors propose a distributed design in which the mobile terminal is responsible for orchestrating the mobility management execution. This design places the intelligence at the edge, which is the starting point of Vehicular Fog Computing (VFC) networks.

VFC, first introduced by Hou et al. in [84], is an architecture that relies on the collaboration of users, both clients and edge devices, to share their resources in terms of communication and computation. In contrast to traditional fog, in which there is a layer between end-users and the edge or cloud, in VFC vehicles and mobile devices are part of that fog layer. VFC was evaluated using both moving and parked vehicles as communication and computation infrastructure. The results showed an enhancement of the capacity and reliability of the network by relying on VFC networks.

Similarly, Zhang et al. [185] propose a hierarchical model for cooperative fog-based intelligent vehicular networks. The model is composed of two layers: the federation of fog elements and the Vehicular Ad Hoc Network (VANET). The survey of Danquah and Altilar [52] presents the challenges of vehicular cloud resource management, considering VANETs as the main components of a dynamic vehicular cloud with mobile resources. The authors propose a decentralized provisioning model based on Peer-to-Peer (P2P) communication, where vehicles with available resources provide services requiring a larger capacity to be executed, offering both Sensing and Network as a Service.

Although vehicular fog computing networks usually rely on a distributed architecture to support seamless handover in fast mobility scenarios, another approach to address the mobility challenges is centralizing the intelligence with an SDN-based solution. Such is the case of Shah et al. [150], who propose integrating SDN technologies into the edge computing environment to orchestrate the hosts responsible for providing service continuity to mobile users.

UNIVERSIDAD DE MÁLAGA

Relying on SDN and containerization at the edge increases the network modularity and portability. In [150], the authors simulate a V2X use case to prove the aforementioned advantages, addressing the interoperability between edge clouds of different MNOs instead of using conventional copy and transfer techniques. However, seamless service migration and mobility management between networks remain a challenge.

Ouyang et al. [124] also propose a centralized SDN controller for dynamic service placement at the edge. Following users' mobility requires frequent service migration among different edge nodes, which could easily overload these edge nodes. Therefore, the authors' approach addresses the cost-efficient aspects of dynamically placing services replacing the service profiles of mobile users among edge nodes.

Additionally, the authors consider extending the framework with Device-to-Device collaboration, so the mobile users can share their computation and communication resources to address the challenges of users' mobility support. The movement prediction issue, which is part of the mobility management challenge, is also discussed by Dalgkitsis et al. in [50], where the authors propose a mobility prediction based on deep learning combined with a genetic algorithm to assist in the orchestration of services. As a result, there is a proactive service migration in edge computing environments with minimal latency and maximal resource utilization.

In [76], Gkounis et al. present a demonstration of the 5GUK Exchange platform, developed in the context of Metro-Haul, MATILDA, and 5GinFIRE projects, which addresses the need for an inter-domain orchestrator in environments with multiple 5G network domains. The platform is an abstraction layer deployed on top of the local ETSI NFV MANO system to act as a service broker and interconnect on-demand cross-domain end-to-end services.

Another approach, introduced by Dang et al. in [51], is the creation of a service-oriented orchestration framework to provide Mobility Management as a Service (MMaaS). This approach derives from the Service-Oriented Computing (SOC) architectures combined with the Autonomous Network Management (ANM) paradigm. To test the autonomy of the orchestrator developed, the authors presented the enhanced mobility management in the autonomous driving use case.

The growth of edge services combined with the high mobility of users has turned into impractical traditional orchestration systems. Aleyadeh et al. address this challenge in [19] by dividing the edge environment into different easily managed components. However, this segmentation must consider the mobility of users, which entails higher processing times and core cloud communication overhead. The authors' approach includes a module for virtual localization with latency-based mapping, a module to map the users according to their mobility, and a final module to merge the information from the previ-

UNIVERSIDAD DE MÁLAGA

ous modules and cluster the users for faster processing and reduced overhead.

The research presented by Slamnik et al. in [155] focuses on the collaborative orchestration of multi-domain 5G edges that support Connected, Cooperative, and Automated Mobility (CCAM) services. The multi-tier orchestration platform designed provides fast reconfiguration of distributed deployments according to the users' mobility patterns and their service/resource demands. It aims at optimizing service continuity and availability of low-latency services in highly dynamic scenarios. Te evaluation of the solution, implemented in the context of the H2020 5G-CARMEN project, shows hat the increase of reference points to distribute the infrastructure relieves the load of the top-level orchestrators, decreasing their overall response time.

3.3.4 *Security in network slicing*

As network slicing is one of the key technologies in 5G and beyond networks, it is due to address the security concerns of orchestrating network slices. Security is common issue to all scenarios, and a crucial point when orchestrating a network with shared resources. Denial of Service (DoS) and exhaustion of resources are the most likely security threats to a network with slices, according to Kotulski et al [97]. In addition, Cunha et al. [48] also consider monitoring, traffic injection, and impersonation attacks. Kotulski et al. consider the limited size of the mmWaves' cells coverage as a form of isolation in terms of radio resources.

In [48], the authors propose cryptography and ciphering as a solution to provide security; specifically, chaos-based cryptography between the user equipment and the base stations for RAN security and Public-Key Infrastructure (PKI) cryptography between slices for inter-slice security. For the intra-slice security, the authors propose a stream cipher and service-oriented authentication to control the access to the core slicing framework.

In [122], Olimid et al. also differentiate security concerns in terms of intra-slice, inter-slice, and at different stages of the slice life cycle, including the preparation of the slice, the installation, configuration and activation phase, the run-time, and the decommissioning. The intra-slice security considers the compliance of the 3GPP requirements at the slice level by implementing end-to-end security procedures, whereas the inter-slice is addressed mainly through isolation mechanisms.

However, applying security measures tailored for each slice is difficult at the orchestration level and not yet resolved in the standard MANO implementation. The distribution of the network across different platforms, administrative domains, and the use of heterogeneous technologies increase the security risks. Thus, standardizing the inter-

UNIVERSIDAD DE MÁLAGA

connection interfaces could only assure slight security. Additionally, the dynamic nature of the slices makes imperative orchestration security policies and security at the orchestrator.

Thus, IoT and multi-access edge computing services represent a major security flaw, threatening privacy, integrity, availability, and authentication, which leads to the need for efficient service-oriented authentication protocols in 5G networks. Furthermore, each service will have specific security requirements, which would restrict the slicing of the network, sharing of the resources, and even the coexistence of different slices that are unable to share their hardware [135][147].

Physical layer security is identified by Yang et al. [177] as a promising approach to provide secure wireless transmissions. The main advantages of this approach, in comparison with cryptography, are the high scalability of the mechanism, and the guaranteed security regardless of the computational capability of the unauthorized device. In the paper, the authors focus on the most relevant enabling technologies for the physical layer, such as Heterogeneous Networks (HetNet), Massive Multiple-input Multiple-output (MIMO), and mmWave.

Sathi et al. designed a protocol [142] to secure intra-slice communication, so-called Implicit Mutual Authentication and Key Establishment with service Group Anonymity (IMAKE-GA) protocol. Thus, trust among entities is enhanced, and association among slice components is ensured. Besides, the robustness of the protocol was verified, and the computation and bandwidth overheads were reduced compared to Type A1 pairing protocols.

Boussard et al. propose in [36] increase network isolation by creating micro-slices on top of 5G network slices. This will improve the configuration of more specific connectivity among devices and applications, defining the micro-slices based on the applications' requirements, and making a more precise differentiation of the slices' conditions. This secure application-oriented slicing was introduced by the authors in [128] and [35], in which the prototype was applied to home environments. In [36], the authors focus on smart scenarios and IoT devices.

The Secure5G framework, introduced in [164], is a deep learning model to prevent attacks based on incoming connections. Secure5G creates a specific slice restricted to a bare minimum QoS to quarantine the threats before infecting the core network while keeping the UE served. The authors propose expanding the Secure5G model for RAN, MEC, and core slicing; and adding secure capabilities to the UE and SIMs.

UNIVERSIDAD DE MÁLAGA

Part III

PROPOSAL AND EVALUATION

This part contains the main contributions of the thesis, with the design of a wireless network architecture for distributed services followed by the development of three different testbeds to evaluate the feasibility of the design proposed. The published work supporting this part includes *"An Architecture for Creating Slices to Experiment on Wireless Networks,"* published in *Journal of Network and Systems Management* in 2020 [169], and *"Dynamic Spectrum Management for European-Wide Research Network,"* published in *IEEE 91st Vehicular Technology Conference* in 2020 [83], for Chapter 4; and *"Expanding GÉANT Testbeds Service to support Pan-European 5G network slices for research in the EuWireless project,"* published in *Mobile Information Systems* in 2019 [134], and *"Validation of NFV management and orchestration on K8s-based 5G testbed environment,"* published in *IEEE Globecom Workshops* in 2022 [113], for Chapter 5.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

4

WIRELESS NETWORKS ARCHITECTURE FOR DISTRIBUTED SERVICES

This chapter presents the architectural principles defined to provide slices, that is, private networks that can be dynamically created with different levels of configuration and control. The resulting infrastructure is supported by a set of distributed Point of Presence (PoP) that transparently manages local and remote resources.

The infrastructure was designed and developed in the context of the EuWireless project[1], aiming at providing a virtual operator for European-wide research as a future-proof framework where new technologies are tested in conjunction with the research on the current ones.

The rest of the chapter presents the main entities of a PoP, different workflows to illustrate their interaction, and its extensibility to integrate new resources, aiming at addressing the first objective of this thesis, established in Section 1.2 as the definition of the architecture.

4.1 POINTS OF PRESENCE SPECIFICATION

The Points of Presence, or PoP, are the core of the infrastructure. A PoP includes the set of hardware and software required to configure and manage a network slice, and can run as a single node or as part of a group of PoPs in case the slice is distributed geographically.

In the infrastructure designed, every PoP follows the same layered architecture, where each layer includes a set of components, and establishes the scope of the functionality offered, as well as the tasks carried out by those components. The high-level architecture of a PoP, shown in Figure 13, includes the following layers:

- Portal & API: The infrastructure is designed to be accessible through a web portal or an API, where we can design slices by customizing or extending some of the available slice templates. These templates are generic slices that include the configurable attributes for every single resource, so that several sub-templates can be derived from the generic one just by establishing a set of values for some attributes. These values can be modified to refine the slice to better suit the specific use case.
- Inter PoP: This layer has a global view of the infrastructure and performs tasks involving multiple PoPs and slices. To do so, this layer stores information on the rest of the PoPs comprising the

1 https://www.euwireless.eu

UNIVERSIDAD DE MÁLAGA

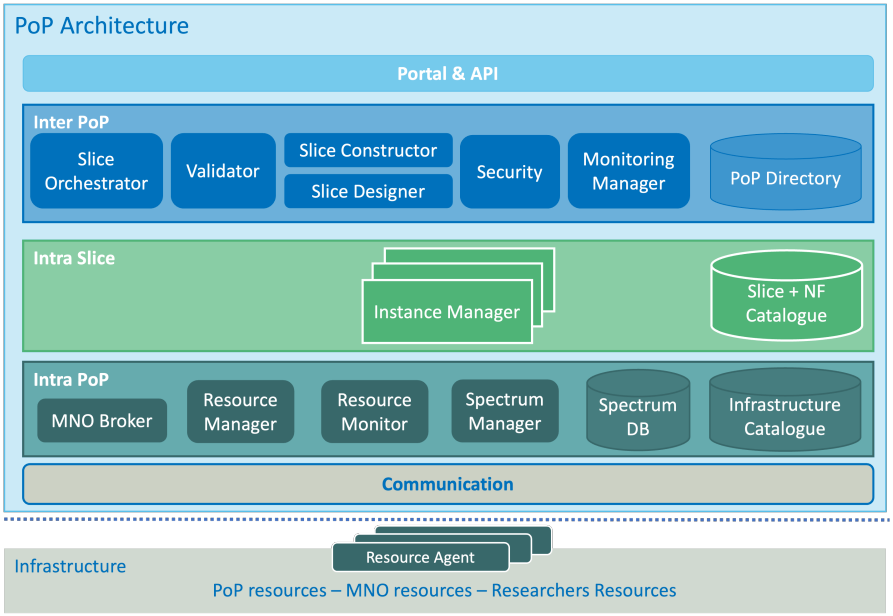

Figure 13: Point of Presence Architecture

infrastructure, including their local resources (not their availability) and their authorized users.

- Intra Slice: This layer is in charge of managing a slice previously designed and deployed, which can span across one or multiple PoPs. To this end, this layer stores the mapping between the slice's abstract description and the actual resources. Since it is likely that we employ a slice template or reuse the same slice description several times, the mapping information will be available to speed up the slice construction phase.

- Intra PoP: this layer interacts with a PoP's local resources. The local resources can have different administrative domains; that is, a PoP's local resources can be owned by the infrastructure itself, by a commercial MNO that shares its resources with the infrastructure, or by a anyone that uses this infrastructure and wants to include their resources into a slice. This layer includes the Infrastructure Catalogue, which keeps information on the resources locally managed by the PoP and their availability, and the spectrum sharing components, such as the Spectrum Database and the Spectrum Manager that interacts with the external repository of the Licensed Shared Access (LSA).

- Communication: this layer provides the connectivity services to the upper layers in order to effectively communicate with other PoPs, by establishing logical end-to-end connections, or with the local resources coming from the infrastructure, researcher testbeds or the MNO infrastructures. In addition, this layer abstracts the network topology and comprises all the low-level

UNIVERSIDAD DE MÁLAGA

protocols such as Transmission Control Protocol (TCP)/IP or any other required to communicate with the MNOs, the resources owned by the researchers, and other PoPs.

When a slice is defined, the resources included must be first mapped into resources of the underlying infrastructure, and then reserved. Figure 14 illustrates this process, where the top of the figure represents the slice definition as a set of virtual resources, such as network functions, virtual machines or links; the middle part depicts the PoP entities (detailed in the following sections) in charge of managing the infrastructure resources that are linked to the virtual resources, and finally, the bottom part portrays the slice deployed in terms of infrastructure resources. Observe that, the resources can be specifically distributed over a geographical area, such as the UE, eNB, and SGW in the location #1, whereas the rest of the EPC functions and external services are mapped to an indifferent location.

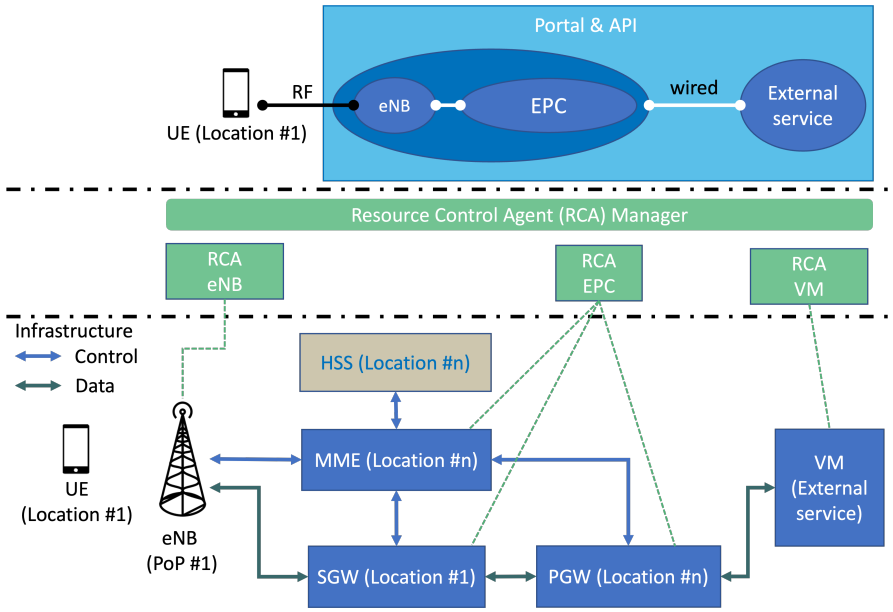

Figure 14: Slice design and mapping

4.1.1 *Portal & API*

The starting point for an experiment is the Portal, where we define the features of the slice to be used in the experiment. However, the Portal acts mainly as a wrapper, translating the slice specification to the API exposed by the PoPs, so we can interact directly with the API to fine-tune resources allocation and perform some low-level or fine-grained configurations.

To support a wide range of experiments and testbeds, we define a generic network slice template that can be customized to better fit each specific use case. Based on this generic template, we also provide

UNIVERSIDAD DE MÁLAGA

a set of pre-configured slice templates targeting specific verticals or use cases. These specialized templates are instances of the generic template that inherit all its attributes, but establishing fixed values within a range of valid values for each attribute.

The Portal also serves as monitoring tool during the slice lifecycle, and for result retrieval after the experiment is finished.

4.1.2 *Inter PoP*

The Inter PoP layer has a global view of the infrastructure. To this end, the entities included in this layer communicate with entities in other (remote) PoPs and also with entities included in the lower layers of its local PoP. Some tasks that require this global view of the infrastructure are the authorization of users, design and creation of a slice, and collection of monitoring information for accounting and billing services.

4.1.2.1 *Slice Orchestrator*

The Slice Orchestrator, from now on the Orchestrator, is the PoP's main administrative entity. It manages and authorizes every aspect of the creation and decommission of the slice as well as the access to resources in slices that can be managed by its local PoP or by a remote PoP.

Each PoP has a single orchestrator entity with the global view of all the slices associated to that specific PoP. The Orchestrator maintains the database that accounts for infrastructure usage. Thus, every resource used in any slice is registered in its database alongside the slice identification and the user requesting it. As it is also the main repository of information for the monitoring procedures, it commands the release of resources that are in an inconsistent state or that have been orphaned by a user disconnection, and informs any other remote Orchestrator of such events in order to act accordingly.

The Orchestrator is the entity that receives user queries, for example to create new slices. In the slice creation process, the Orchestrator is in charge of blocking the local resources required, so no other Orchestrator finds them available, and of sending the queries to the other PoP's Orchestrators if the resources are remote.

Finally, when all the resources are blocked, the Orchestrator sends the slice definition to the Slice Constructor, which will proceed with the configuration and interconnection of the resources following the topology defined.

4.1.2.2 *Validator entity*

The Validator entity checks the slice definition provided through the Portal & API at different levels. First, it makes a syntactic and

UNIVERSIDAD DE MÁLAGA

semantic check of the slice definition in order to find syntactic errors, inconsistent network topologies, or resource misconfigurations.

If the slice definition passes the first check, then, the Validator determines whether the new slice will compromise the performance of the infrastructure and its already running slices. This analysis includes checking the slice priority level, and a pre-deployment analysis of the QoS requirements and/or SLAs specified in the slice definition.

Examples of errors that the validation shall detect include the following:

- Components with missing connections (e.g., the user may request a base station but forgot to add the link to the core network).
- Existence of link loops that divert traffic unnecessarily.
- Mismatched resource allocation, such as requesting a 10 Gbps resource connected to a 1 Gbps link or defining a service with 10 users but registering the credentials for only 5 users.
- Inconsistent network configurations, such as connecting two components without using a link resource.

4.1.2.3 *Slice Constructor*

Once the slice definition is considered valid, the Constructor carries out the configuration of the resources, according to the slice definition, interacting with the entities in the local and remote PoPs to configure and activate the required resources.

The first step is to map the slice definition into the resources forming the infrastructure. For example, if the slice definition includes an isolated virtual MNO with two real base stations, the Constructor has to determine the computational requirements to run the EPC's software, the spectrum capacity needed by the base stations and the links that have to be provisioned to interconnect each element. This information is obtained from the Slice and NF Catalogue, or from the Slice Designer, depending on if there is a predefined slice template already available, or it is required to create a new one, respectively.

After mapping the resources, since the Orchestrator reserved all resources at the administrative level, the Constructor can proceed with the configuration of local and remote resources directly with the corresponding Resource Manager. As soon as all the resources are activated, the Constructor passes all the slice information to the Instance Manager, notifies the Orchestrator of the successful creation and ends its lifecycle.

UNIVERSIDAD DE MÁLAGA

4.1.2.4 *Slice Designer*

As mentioned earlier, the portal includes several templates in order to ease the definition of the slices. These templates include the mapping to real network topologies saved in the Slice and NF Catalogue, ready to be deployed in the infrastructure. However, the slice description may include resources that can be mapped to multiple real resources (physical or virtual) and it is not possible to foresee all possible slices and their corresponding resources required in each case beforehand. Given that some slice definitions may not match an existing template, the Designer maps the slice definition to virtual or physical resources available in the infrastructure.

The design of a new slice from scratch requires two steps. In the first one, the Designer takes the role of an architect that lays out the high-level topology of the slice definition, using abstract information of the resources of the PoPs involved. The second phase is the low-level realization of that design. For example, if the slice definition indicates that two components are connected through a "link", the Designer maps the link resource into physical network interfaces, and virtual switches or routers, i.e., using NFs.

The advantage of using this entity instead of simply having a static map between abstract resources and real ones is the capacity to compose recursive NFs from other functions, and delegate their management to a single Resource Agent (RA). It is worth mentioning that the Designer does not ensure the resource availability, which is the Orchestrator's responsibility; it only finds the appropriate resources to construct the slice and passes them to the Constructor.

4.1.2.5 *Security entity*

This entity supports the authentication and authorization procedures in order to provide a reliable and secure service for both infrastructure users and other PoPs. It maintains the list of users and their access profiles for the local PoP. Thus, the Security entity is in charge of granting or rejecting access to a specific slice or resource. Due to the distributed nature of the infrastructure, there are PoPs located on specific institutions that include their proprietary hardware and software resources, so it is necessary to provide access to users of remote PoPs, or authenticate and redirect them to the appropriate destination.

Hence, Security entities located in different PoPs can request each other's credentials and authorization information about their users. Given the heterogeneous types of users, considering they are affiliated to different academic and research institutions, the Security entity will support external Authentication, Authorization and Accounting (AAA) services such as RADIUS, Lightweight Directory Access Protocol (LDAP), etc., so each institution can apply its own policies

UNIVERSIDAD DE MÁLAGA

on access control. Security tasks can be activated under request (i.e., when a researcher logs into the Portal) or automatically scheduled. There are also security tasks related to the validation of the defined slices, especially when the definition includes resources from commercial MNOs, to detect possible threats to commercial operations and related to the monitoring of the slice. For the validation tasks, the security entity interacts with the Slice Validator.

4.1.2.6 *Monitoring Manager*

The Monitoring Manager collects the monitoring information of all the slices associated to the PoP and generates meaningful reports for experiment analysts. In addition, the Monitoring Manager can also be configured to receive alerts on anomalous conditions encountered by the Resource Agent (RA)s. The general health of the infrastructure is monitored taking advantage of this behaviour, since the information aggregation and alert capabilities are not limited to the information stored in RA messages, but can also be used to process external logs produced by any component.

4.1.2.7 *PoP Directory*

The PoP Directory maintains information about the PoPs that comprise the infrastructure, and their resources. The directory specifies, among other information, how to communicate with the remote PoP entities and the type of resources available in the remote PoP. This information is fundamental to create a slice with resources distributed across multiple PoPs. In this case, the Orchestrator that receives the request for a new slice, checks where these resources are. The information stored in the PoP Directory is static, since it contains the resources but not their availability.

The resources' availability is obtained by querying the corresponding remote PoP. As the infrastructure will not change frequently (number of PoPs and resources), updates of the PoP Directory will only take place occasionally. Thus, updates in the PoP Directories are seen more as administrative tasks that can be performed "by hand" rather than by defining a communication protocol between PoPs to update their directories. In any case, if the number of PoPs grows and the administrative process becomes cumbersome, the static list can be moved to a distributed database hosted in every PoP, where the updating process is done dynamically using an out-of-band communication between PoPs, without altering the semantic and interfaces offered by the directory.

UNIVERSIDAD DE MÁLAGA

4.1.3 *Intra Slice*

The Intra Slice layer includes entities dedicated to a specific slice, from creation to management. These entities are not aware of the distributed nature of the slice and its resources.

4.1.3.1 *Instance Manager*

Once a slice is ready to be used, a specific Instance Manager takes care of the slice's management and monitoring tasks. The Instance Manager deals with the PoP user's requests related to their slice. This PoP user, who provided the slice definition, is considered the slice owner. For example, when requesting the status of the slice or a slice resource, or when a resource must be deactivated/reactivated to test the system's behaviour (e.g. it may shut down a link or activate a new one to check the response of a load balancer). In these cases, the Instance Manager translates the request to the appropriate primitives that the Resource Manager and the RAs understand.

A resource can be in four states, namely reserved, activated, deactivated and unreserved. A reserved resource is assigned to a slice but not in use. An activated resource is configured and ready to function into the slice whereas a deactivated one is still reserved but in a "shutdown" state (its meaning depends on the type of the resource).

In addition, the Instance Manager takes part in the monitoring tasks in two different ways. It serves as a proxy to monitor requests sent by the user and, more importantly, it is also responsible for the low-level monitoring of every resource, aggregating messages by components before sending them to the Monitoring Manager, and paying attention to any alarm or exception thrown by the resources used in the slice it manages.

4.1.3.2 *Slice plus NF Catalogue*

Slice descriptions are expected to be used several times, by the same user performing repetitions of an experiment or by different users requesting the same slice topology (albeit customized for another experiment). The Slice and NF Catalogue is used to save the mapping between the abstract description and the actual components used in the slice, so when a user requests a "known" slice, the Constructor starts sending reservation queries to the appropriate Resource Manager right away instead of employing the time-consuming process of using the Designer.

4.1.4 *Intra PoP*

The Intra PoP layer includes the entities that are only aware of the resources of a slice allocated to the current PoP. These elements do

UNIVERSIDAD DE MÁLAGA

not have a complete view of a slice, as they only interact with the resources associated to the local PoP. This layer also includes the components in charge of the dynamic spectrum management. However, to ease the section reading, the dynamic spectrum management entities are presented in the subsequent section.

4.1.4.1 *Infrastructure Catalogue*

This is a database of the PoP's physical resources. It contains the number of resources present in the infrastructure and their capacity, as well as the number of free resources or available capacity. This component is used by the Slice Orchestrator in order to check resource availability at the slice creation phase, and it is updated every time a slice is started or stopped.

4.1.4.2 *Resource Manager and Resource Agents*

The Resource Manager is in charge of managing the resources associated to the PoP independently of the slice where the resource is used. Since resources can have different nature, the communication can be done using different protocols or primitives. In order to abstract the different nature of the resources, the Resource Manager assumes that for each resource, or resource type there is a RA that acts as a wrapper of the resource which is able to understand the management primitives of the Resource Manager and translates them to the corresponding management commands understood by the physical/virtual resource.

The RAs are responsible for the virtualization and lifecycle management. Based on the literature, a RA should manage any resource based on six control primitives, and thus all of them must be implemented in every RA: Reserve(), to allocate the physical resources and create the instances; Activate(), to configure the instances and put them in service; Reconfigure(), to change configuration parameters and attributes after the first installation; Query(), to obtain information about the resources' state; Deactivate(), to stop the instances without releasing the resources; and Release(), to destroy the instances and return the allocated resources to the available pool. Figure 15 represents the resources' lifecycle controlled by the primitives. It is worth mentioning that only resources that have already been blocked by the Orchestrator can be reserved by the RA as a first resource configuration step. The blocking task performed by the Orchestrator ensures the resources are available and will not be requested by any other slice, whereas the reserve primitive performs the allocation of the physical resources to prepare them for activation.

In the case of MNO resources, the Resource Manager makes the configuration and release of resources through the MNO Broker, which includes, among others, the functionality of the RA. The Resource

UNIVERSIDAD DE MÁLAGA

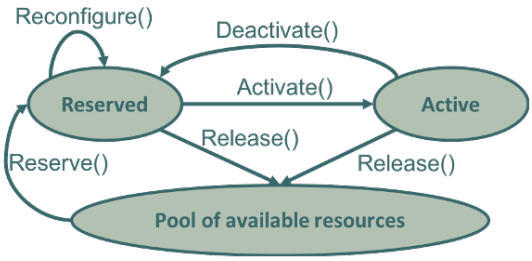

Figure 15: Resource lifecycle state machine

Manager is the single point of entry to manage and monitor the RA that control a PoP's infrastructure.

4.1.4.3 *MNO Broker*

As mentioned earlier, one of the pillars of this architecture is the shared use of MNO's resources in the form of slices, whether radio spectrum, a network function, or computing capacity, in a transparent way. In an ideal world, the access to those resources would be achieved using standardized interfaces and procedures. However, in the real world each commercial network is highly customized to the specific operator's needs.

The Broker acts as a middleware, exposing on one side an interface as similar as possible to the one of a RA, and on the other side conforms to any requirement the operator may impose to access its resources. The reason for distinguishing this component from a RA is that all operators are going to impose their own rules and access restrictions to their resources, and each PoP will have to customize its code to adapt it to the requirements of their local operator. Besides, the MNO Broker will be in charge of updating the so-called Spectrum Database (SD), which is part of the Infrastructure Catalogue.

4.1.4.4 *Resource Monitor*

This entity monitors RA health parameters using low-level and high-performance primitives, and sends them to the Monitoring Manager through the Instance Manager for processing. In addition to the usual polling mechanism to read its value, the resource monitor can also be configured to trigger an alarm when a parameter of the monitored resource reaches a certain threshold.

4.1.5 *Dynamic Spectrum Management*

The dynamic spectrum management, in the form of spectrum access and sharing, is one of the key enablers for 5G and beyond communication systems. Applied to the architectural approach hereby presented, the PoP should provide a unified interface to the slice own-

UNIVERSIDAD DE MÁLAGA

ers from which suitable frequency bands can be reserved, as well as an entity to negotiate the spectrum access for a certain area and time following the local regulations and in harmony with the local MNO. Moreover, depending on the experiment size, it might require bands of spectrum available across multiple PoPs located in several countries. The entity in charge of these negotiations is the Spectrum Manager (SM).

On the other hand, the information on the spectrum resources is stored in the database called Spectrum Repository (SR). The SM combined with the SR use the LSA technique to enable the MNOs spectrum sharing by collecting the information from the external LSA repository, reserving, and releasing spectrum bands [83].

Following this approach, a slice including licensed spectrum as a resource comprises the RAN components, the RAN controller interfacing between the RAN and the rest of the network, the local SM instance, and the SR. This is depicted in Figure 16. The SM sends both the information on available bands and the order to evacuate a spectrum band to the RAN controller, and negotiates the spectrum availability with the MNOs according to the LSA specifications via the LSA repository. This entity is also responsible for collecting the availability data on the PoP's spectrum bands to maintain the SR. Hence, the Orchestrator can verify through the SR the spectrum availability when the slice spans across different PoPs.

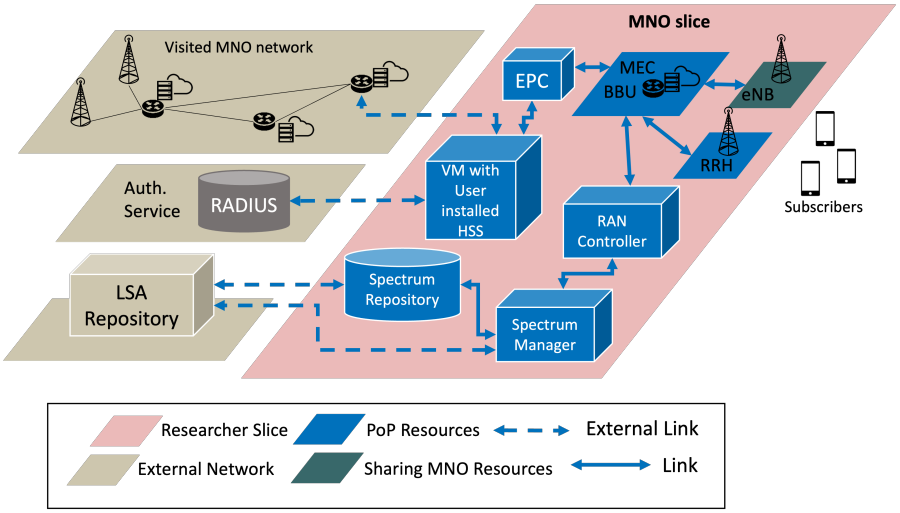

Figure 16: Spectrum sharing slice

4.2 POINTS OF PRESENCE WORKFLOW

This section presents the main interactions between the PoP entities through a collection of message sequence charts. Specifically, the workflows describe the processes of slice creation, slice release and temporary resource pause.

UNIVERSIDAD DE MÁLAGA

To simplify the presentation in the workflows, we assume that the infrastructure user has been previously authenticated. The authentication procedure is delegated to the local PoP's Security entity, which provides a reliable and secure service for both users and other PoPs by maintaining the list of users and their access profiles for the local PoP. Given the distributed nature of the infrastructure, the Security entity supports external AAA services such as RADIUS, LDAP, etc., so each institution can apply its policies on access control.

4.2.1 *Slice creation*

The definition of a network slice relies on a modular virtualization system that supports an extensible catalog of (abstract) resources that can be mapped to a single o a set of physical (or virtual) resources. Each resource exposes a set of attributes that can be tuned to fit the user's needs, as well as the I/O communication ports in order to interconnect with others. These data paths that interconnect resource I/O ports define the network slice topology. In order to deploy a network slice in this infrastructure, the slice definition includes the set of resources that integrates, their attributes configuration, and the underlying network slice topology.

The following subsections describe the workflows associated to creating one slice when the resources are local to a PoP, combined from different PoPs, and combined with the MNO domain, respectively. The last subsection depicts a failure scenario with resources combined from different PoPs.

4.2.1.1 *Slice creation using one PoP's local resources*

Figure 17 depicts the workflow to set up a network slice using only one PoP's local resources. The slice creation starts with a request including the configuration of resources and topology through the portal (or API). In this example, all resources will be managed by a single PoP.

When the definition is ready, the Slice Orchestrator receives the Create Slice query, and delegates validation of the slice definition to the Validator. The Validator performs a syntax check to ensure that the slice definition conforms with the syntax of the slice description language, and a semantic validation to check parameter misconfigurations, such as exceeding link's capacity, inconsistencies in the deployed components, or scenarios that can compromise the performance of the infrastructure or the QoS of the running slices.

If the slice definition is valid, the Slice Orchestrator blocks the resources, changing their status in the infrastructure catalogue to assign that resource to the new slice under creation. At this stage, blocking a resource does not entail communication with the resource itself; it

UNIVERSIDAD DE MÁLAGA

is just a change in its availability in the corresponding Infrastructure Catalogues.

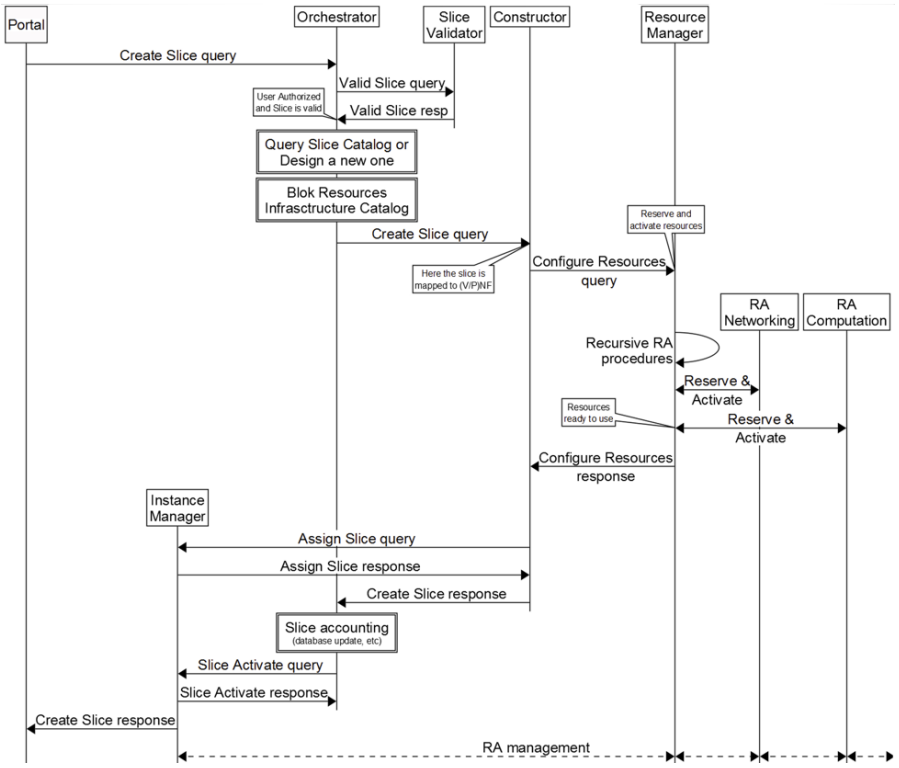

Figure 17: Slice creation using one PoP's local resources

If all the requested resources are available to be blocked, it is time for the Slice Constructor to determine if the definition of the slice is based on a template or if it is a manually crafted slice. In the first case, the Slice Constructor directly sends the request to the Resource Manager, located in the intra-PoP layer, whereas in the second case, the Slice Constructor needs the Slice Designer to map the slice definition into resources before sending the request.

Then, the Resource Manager establishes a connection with the specific resources by means of the RAs to reserve and configure them. In the slice lifecycle, each physical (or virtual) resource has to be reserved, configured, activated, and released. However, different types of resources need different configuration parameters, and, in general, they can present a different management interface.

The role of the RA is abstracting and unifying these common tasks, from the perspective of the Resource Manager. Thus, each resource in the infrastructure has an associated RA that abstracts its actual management into a series of standardized primitives, and the RA is in charge of translating these primitives into the specific sequence of commands understood by the target resource.

When the resources are configured, the Resource Manager informs the Slice Constructor of the successful initialization of the slice. From

UNIVERSIDAD DE MÁLAGA

that point onwards, the Instance Manager deals with the user's request related to their slice, translating them to specific requests or queries to the underlying resources. In addition, the Instance Manager takes part in the monitoring tasks in two different ways: it serves as a proxy for the monitoring requests sent by the user but, more importantly, it is also responsible for the low-level monitoring of every resource, aggregating messages by components before sending them to the Monitoring Manager, and paying attention to any alarm or exception thrown by the resources used in the slice it manages.

4.2.1.2 *Slice creation using multiple PoPs' resources*

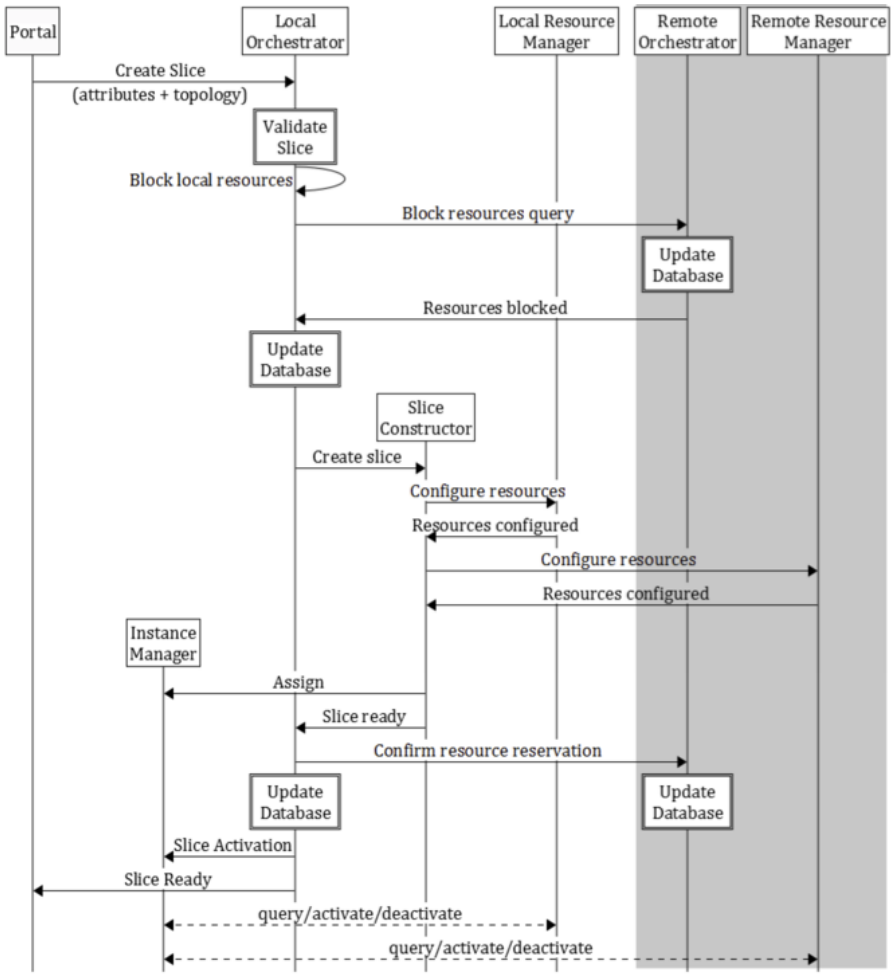

Figure 18: Slice creation using multiple PoPs' resources

Figure 18 describes the workflow to create a network slice that integrates resources distributed in different PoPs. In the multi-site network slice scenario, different PoPs have to be coordinated in order to reserve and deploy the resources included in the slice. The Slice Orchestrator of the local PoP (the PoP associated with the user requesting the slice) is the entity in charge of coordinating the whole process, as

UNIVERSIDAD DE MÁLAGA

it is the PoP's main administrative manager. It receives the request to create the network slice (including the slice description in terms of resources and topology) and coordinates the blocking of resources in the local and remote PoPs.

In the slice creation process, the Orchestrator checks with the Slice Validator that the definition is correct. The Validator only performs the syntactic and semantic correctness of the initial slice definition without considering the current resource availability.

Once the design is cleared, the Orchestrator proceeds to block local resources, so no other Orchestrator finds them available. In the case of remote resources, the Orchestrator sends the query to the Orchestrator of the corresponding remote PoP where the remote resources are connected. The Orchestrator maintains the database that accounts for infrastructure usage.

Thus, every resource used in any slice is registered in its database alongside the slice identification and the user requesting it. This way, as it is also the main sink of information of the monitoring procedures, it commands the release of resources that are in an inconsistent state or that have been orphaned by a user disconnection, and informs any other remote Orchestrator of such events in order to act accordingly.

Finally, when every resource is blocked, the slice definition is sent to the Slice Constructor, which configures and interconnects the resources following the slice definition topology. The Constructor carries out the reservation and configuration of the resources and its mapping into the infrastructure. The Constructor obtains the mapping from the Slice+NF Catalogue, if it is a reused slice, or can request this task from a dedicated entity, called the Slice Designer. Given the resource mapping, the Constructor proceeds with the reservation and configuration of local and remote resources through the corresponding Resource Manager.

If a given slice definition does not match any of the existing templates or definitions in the Slice+NF Catalogue, the Designer maps the slice definition to virtual or physical infrastructure resources. To this end, the Designer first, takes the role of an architect that lays out the high-level topology of the slice definition, using abstract information of the resources of the PoPs involved. Second, it translates the previous topology into the infrastructure resources with specific requirements (e.g. a link with a specific bandwidth, a virtual machine with a number of cores and RAM, a chain of NFs, etc.). It is worth mentioning that the Designer does not ensure the resource availability; it just finds the appropriate resources to construct the slice and passes them to the Constructor.

In the multi-site scenario depicted in Figure 18, the Slice Constructor communicates with the local and remote Resource Managers, which

UNIVERSIDAD DE MÁLAGA

are the entry points to configure and manage the resources associated with each PoP.

When all the resources are configured, each Resource Manager informs the Slice Constructor, who passes the management of the slice to a specific Instance Manager and informs the local Orchestrator, that finally, commands the slice activation and informs the user of the successful creation of the slice. Finally, the Slice Constructor assigns the control of the slice to a dedicated Instance Manager and informs the user through the Portal.

4.2.1.3 *Slice creation using resources from the MNO domain*

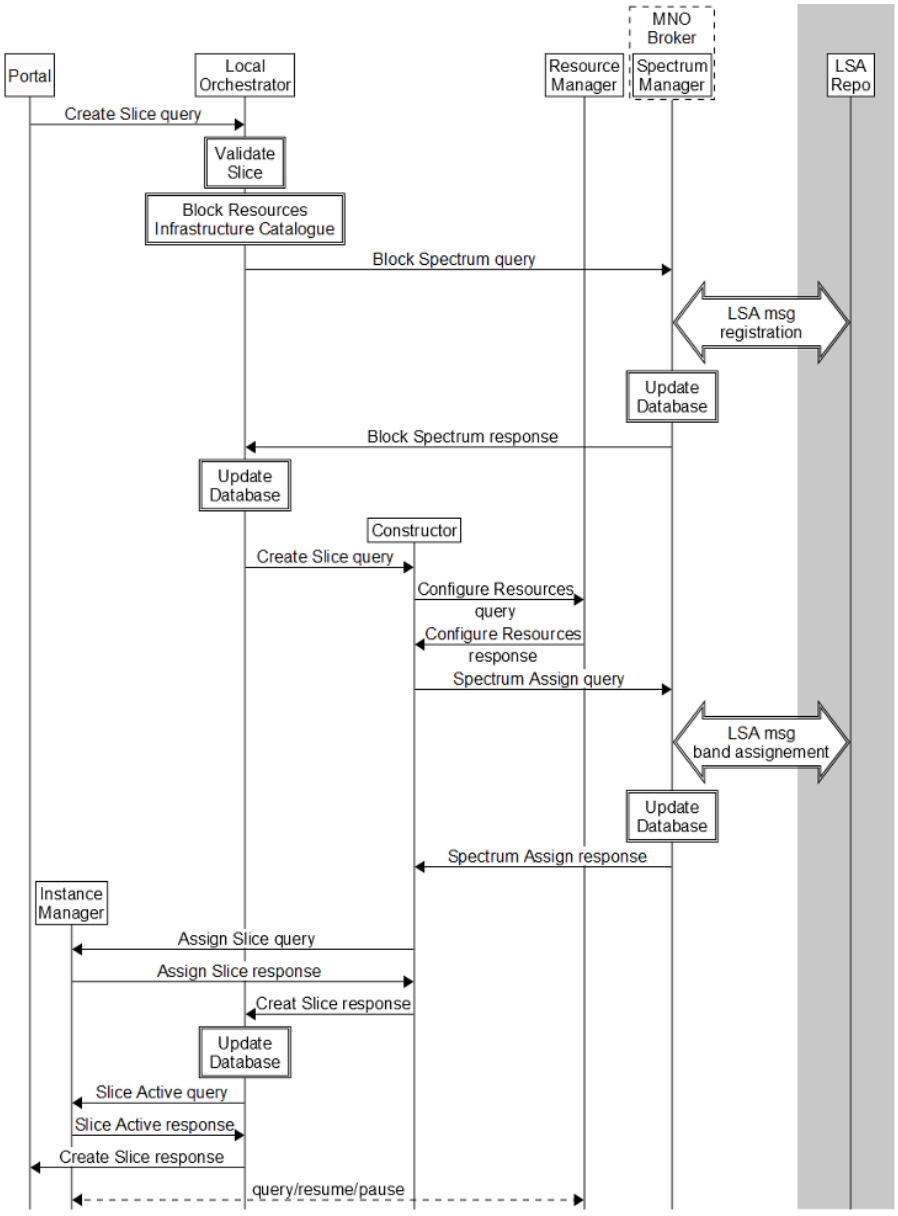

Figure 19: Creation of a slice with shared spectrum

UNIVERSIDAD DE MÁLAGA

The previous workflows assume that the slices do not integrate resources shared by a commercial MNO. For any other case, the MNO Broker interacts with commercial MNOs, and includes the Spectrum Manager entity, which implements all the protocols to reserve, obtain and release frequency bands using LSA technology to access licensed spectrum.

The Spectrum Manager communicates directly with the LSA Repo, which is an entity managed by external regulators where the spectrum users and LSA licensees (i.e., the MNO) register their requirements and constraints to share and use spectrum. Thus, the LSA Repo is an external active entity that provides information on spectrum availability.

Figure 19 depicts the workflow of a slice creation integrating spectrum shared by a commercial MNO using the LSA technology, which involves the registration in the LSA repository, and the reservation of frequency bands. To simplify the diagram, we assume that the spectrum bands used in the slice are associated to the local PoP. As mentioned earlier, spectrum is a resource managed through the Spectrum Manager included in the MNO Broker, who implements the LSA procedures to register in the LSA Repository and request the spectrum bands.

4.2.1.4 *Slice creation failure*

Certainly, different errors can occur during the creation of a slice, such as an invalid description of the slices, unavailability of resources, or connectivity problems with external domain resources. Figure 20 depicts a failing example in which the local Slice Orchestrator has successfully blocked the local and remote resources.

However, when the Remote Resource Manager starts the resource configuration and activation, the RA responds with an error from the resource. In this situation, the Resource Manager sends the error to the Slice Constructor, who deactivates and releases the resources that had been previously activated.

When the resources are released, the Slice Constructor informs the local Slice Orchestrator of the error, which frees the resources on its database and requests the remote Slice Orchestrators to release their associated resources. Finally, the local Slice Orchestrator informs of the problem through the Portal.

4.2.2 *Slice release*

The final step of a slice's lifecycle is the permanent release of its resources. Figure 21 shows a successful slice decommission triggered by the slice owner.

The Instance Manager is the entity in charge of managing this decommission. Thus, this entity receives and propagates the request

UNIVERSIDAD DE MÁLAGA

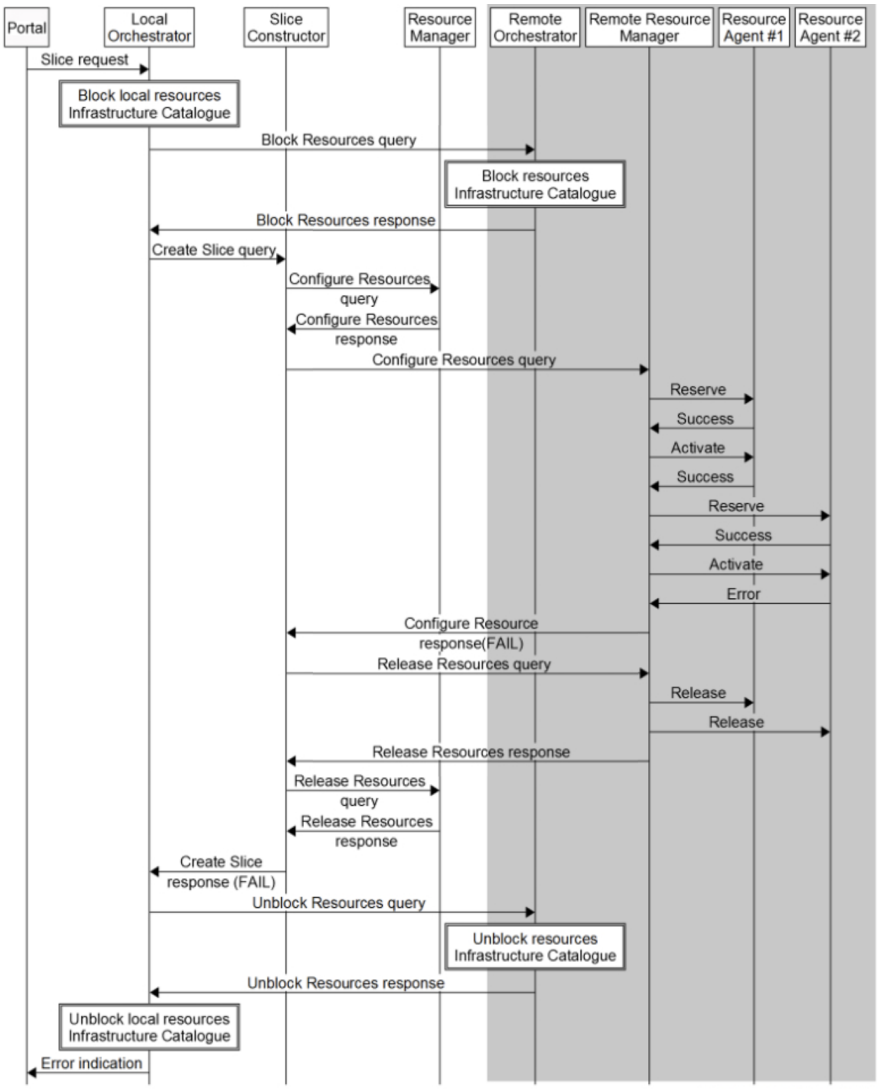

Figure 20: Failed slice creation using multiple PoPs' resources

of decommissioning to the local and remote Resource Managers that will send the *deactivate* and *release* primitives to each resource through the RAs. When all the slice resources are released, the Instance Manager passes the control to the local Orchestrator, which updates the Infrastructure Catalogues with the newly available resources, in collaboration with the remote Orchestrators.

Figure 22 shows the decommissioning procedure when spectrum is involved. In this case, the release of the spectrum includes the communication of the Spectrum Manager with the LSA Repository to release the frequency bands, and to unregister the Spectrum Manager from the repository, since no more information on spectrum availability is required.

UNIVERSIDAD DE MÁLAGA

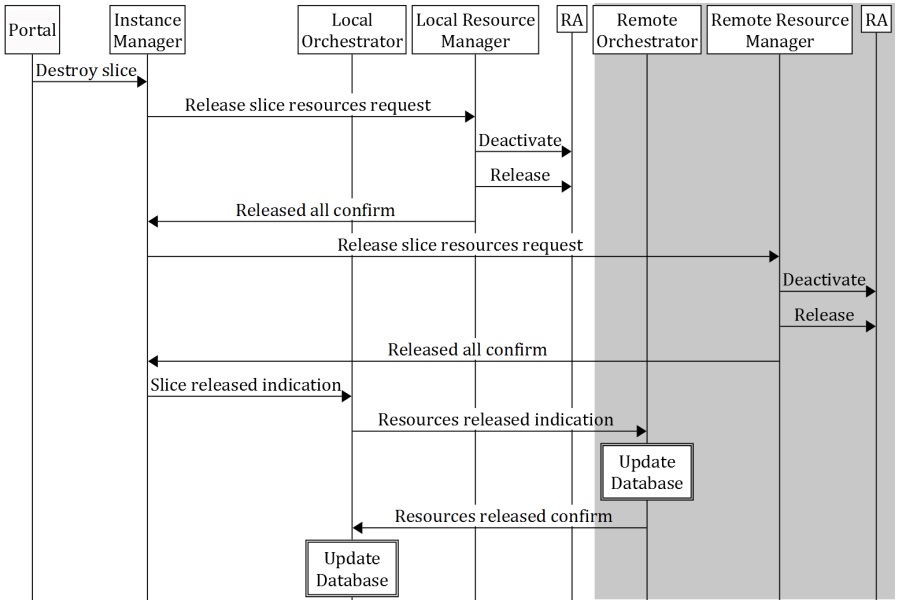

Figure 21: Decommission of a slice

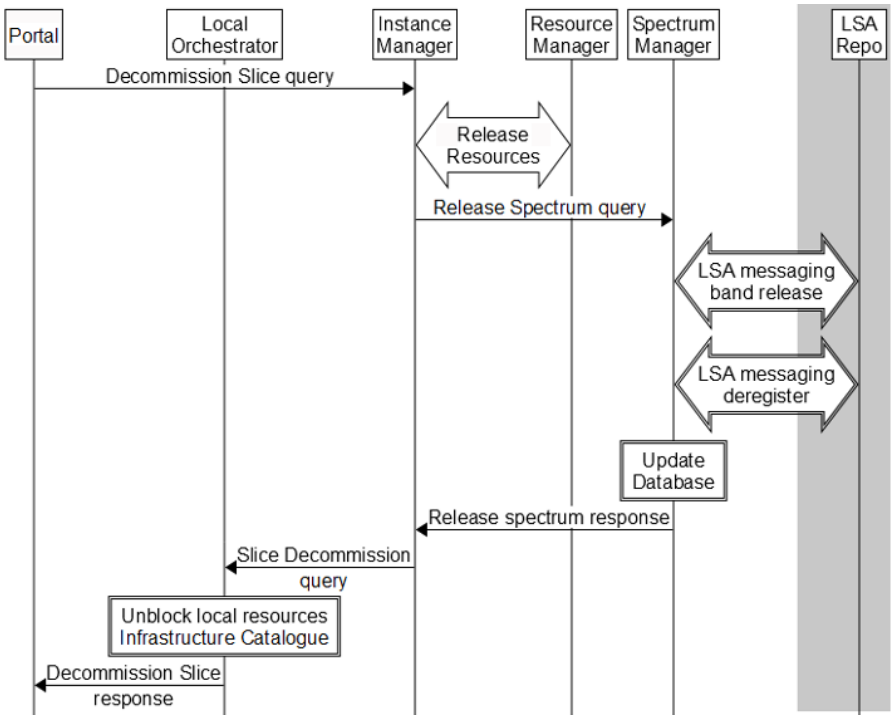

Figure 22: Decommission of a slice with shared spectrum

4.2.3 *Temporary resource deactivation*

During the lifecycle of a slice, the slice owner can temporarily deactivate a resource without releasing it, for instance, to run a particular experiment in which the resource is not required. This process is similar to the slice release. However, in this case, the deactivated resource remains reserved and unavailable to be used in other slices.

UNIVERSIDAD DE MÁLAGA

Hence, the Instance Manager sends a request to deactivate the resource instead of releasing it, and then waits for user interaction to resume its operation through the Portal. During this stage, the deactivated resource remains unavailable in the Infrastructure Catalogue. Even though the temporary deactivation of resources modifies the configuration of the slice and could lead to unstable states, it is not the responsibility of the Instance Manager to ensure the correct operation of the slice after the initial deployment.

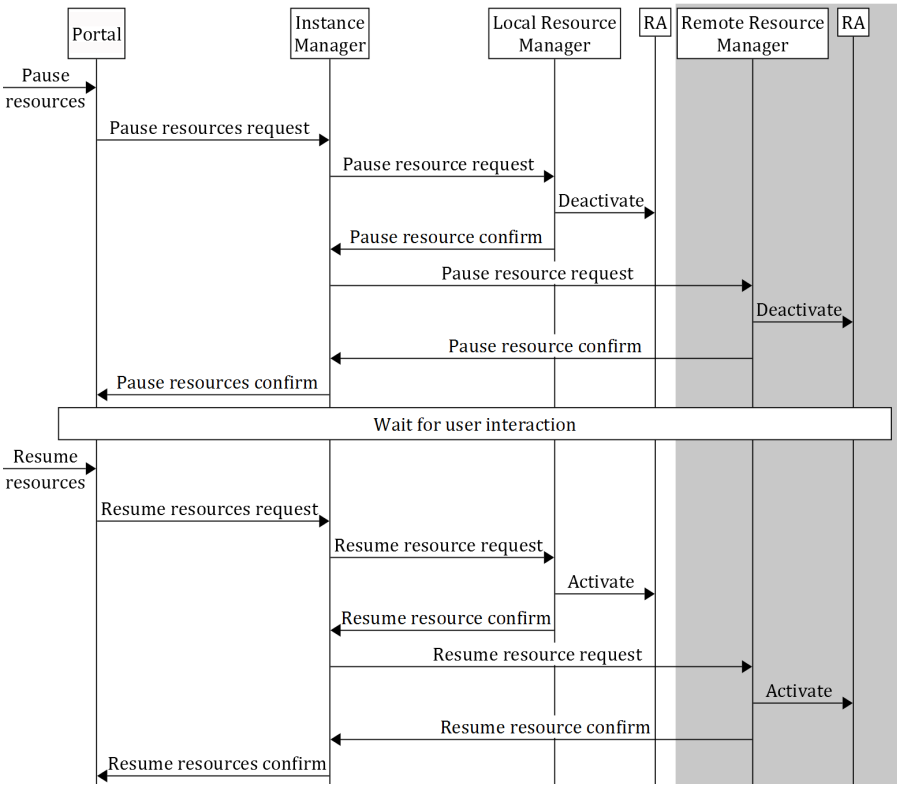

Figure 23: Temporary resources deactivation

Figure 23 shows the workflow of deactivating two resources, one in the local PoP and the other remote, and later re-activating the resources. The slice owner requests the pause/deactivation of these resources, which are integrated in a running slice, using the Portal. Since the slice is running, the request is transferred directly from the Portal to the Instance Manager, who requests the local Resource Manager to deactivate the resource. For remote resources, the Instance Manager sends the request to the remote Resource Manager, located in the remote PoP.

Regardless of where the resources are located, the Resource Manager contacts the resources through the associated RA, which stops the resource and confirms the successful deactivation. To reactivate the resources, the process is the same, starting on the Portal and transferring the request until the resources confirm reactivation. During

UNIVERSIDAD DE MÁLAGA

the whole process, the slice owner can use the provided monitoring tools to detect the correct operation of the slice.

Finally, Figure 24 shows the procedure for the spectrum release when it is initiated by the LSA Repo due to changes on the spectrum availability. In the case that the MNO stops sharing the frequency band, the LSA Repo informs the Spectrum Manager that spectrum is no longer available. The Spectrum Manager then acts as a RA, informing the Instance Manager of this issue, who informs the Orchestrator and the slice owner through the Portal, following a similar approach to the temporary deactivation of resources.

At this point, the user decides between decommissioning the slice if the MNO will not share the frequency bands anymore, or wait until the frequency bands are available again, which is notified through the LSA Repo, to re-activate the spectrum following the same procedure presented in Figure 23 when the Spectrum Manager receives the availability change notification.

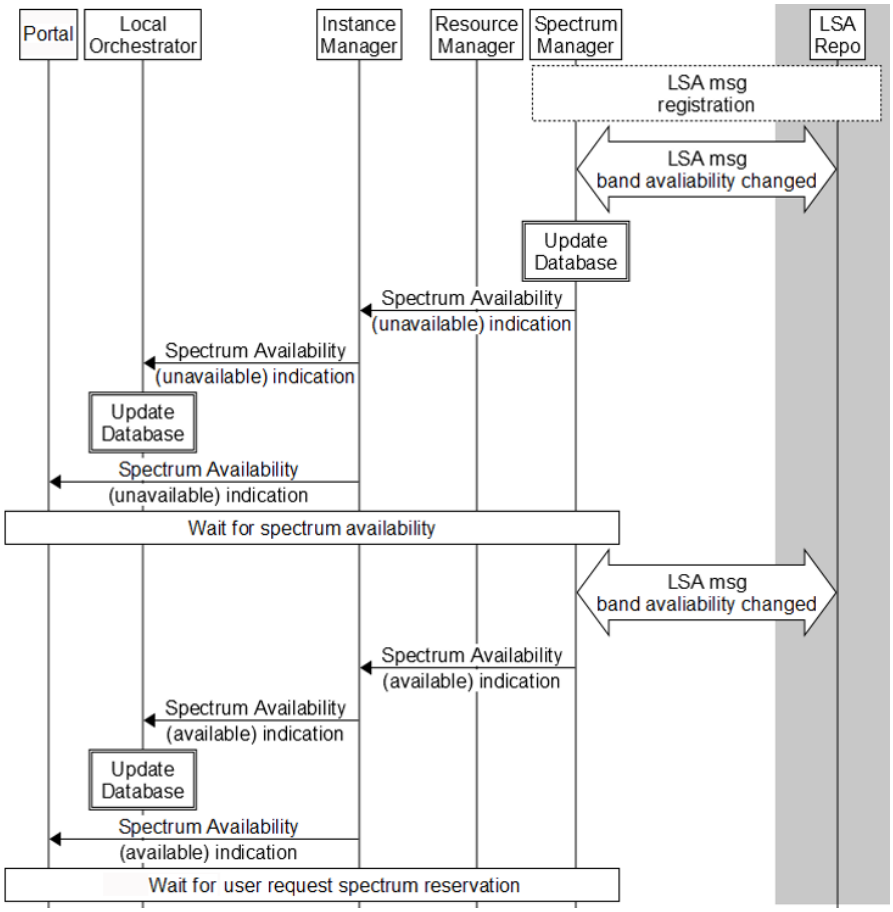

Figure 24: Spectrum availability change

UNIVERSIDAD DE MÁLAGA

4.3 POINTS OF PRESENCE EXTENSIBILITY

As described in Section 4.1, the Resource Manager is able to handle any resource regardless of its nature or administrative domain transparently due to the abstraction provided by the RAs. Thus, there is one RA associated to each type or resource, acting as a middleware to control the resource's lifecycle and trigger the transition among its different states, namely reserved, active, and released. The released state, depicted in Figure 15 as "the pool of available resources", corresponds to the state of blocked by the Orchestrator, so that the resource is available for the slice in question and unable to be used by any other slice.

A direct consequence of this abstraction is the infrastructure flexibility to integrate new resources, as long as there exists a specific RA to control them.

The technology inspiring this approach is GTS, a production service created by GÉANT in 2013 as a tool for researchers to create isolated virtual testbeds within a shared infrastructure, offering highly flexible and efficient control of different networking components [168].

Bearing this in mind, the definition of a resource provided before deployment must include its requirements, ports, and attributes. The slice templates ease this definition assigning default values to the attributes unspecified to provide an initial configuration. Some of these attributes might be reconfigured after the resource has been activated and tested. This section provides two examples of resources integrated in the infrastructure, an EPC and an eNB, which are essential to deploy LTE network slices.

Table 2 presents the definition of a virtualized EPC, that is, the set of requirements, attributes and ports to be configured to deploy an EPC in a slice. Furthermore, it includes the control primitives translation into specific actions on the resource. In the table, the requirements are the requisites in terms of computing resources to schedule and reserve them. The ports define the slice topology, depicting the data paths from and to each resource instance. The attributes for an EPC type of resource include the information on the MNO to register the subscriber, such as the Access Point Name (APN) providing internet access, the Tracking Area Code (TAC), the Public Land Mobile Network (PLMN), which includes the Mobile Country Code (MCC) and the Mobile Network Code (MNC), the authentication parameters, and the list of IP addresses and subscriber identities, the so called International Mobile Subscriber Identities (IMSI). The last block of rows introduces the lifecycle control primitives simplified.

In this example, the resource is a VM running an open source implementation of the EPC, the so-called NextEPC[2], which corresponds to the LTE 3GPP Release 13. Thus, after its definition, the platform is

2 Available at https://nextepc.org/

UNIVERSIDAD DE MÁLAGA

able to automate the provisioning, and the associated RA configures and activates the resource.

The second example is the integration of an eNB, which abstracts a physical resource that radiates in a specific coverage area where we can allocate subscribers. In this case, as depicted in Table 3, the resource includes only one port to communicate with the PoP containing the rest of the slice, and the RA implements the primitives to control the eNB. The only requirement for this kind of resource is the availability of the physical resource itself; and, besides the location of the PoP where the eNB is placed, the only configurable attribute is its IP address.

UNIVERSIDAD DE MÁLAGA

Requirements	CPU	1 processor
	RAM	4 GB
	HDD	25 GB
	OS	Ubuntu 18.04 distribution with NextEPC software
Ports	S1-MME	Connects EPC-eNB(s) and transports control plane information
	S1-U	Connects EPC-eNB(s) and transports data plane information
	SGi	Connects EPC-external IP networks
Attributes	Location	PoP selected to instantiate the VM with the EPC software
	PLMN	PLMN's value, with MCC and MNC
	TAC	TAC's value
	APN	APN's value
	IP Pool	List of IP addresses available for the UEs
	Subscribers	List of subscribers identified by their IMSIs
	Authentication parameters	K code and OP/OPc authentication parameters
Primitives	Reservation	Locates the requirements to instantiate the VM in the PoP and the links for the ports
	Activate	Installs the OS, configures the attributes, and initializes the EPC
	Deactivate	Stops the VM and traffic flow, but maintains the connections and resources
	Reconfigure	Changes configuration parameters and attributes after instantiation
	Query	Provides information on the resource's state
	Release	Deletes the VM and links, and returns the resources to the pool

Table 2: Example for the integration of a virtualized EPC

UNIVERSIDAD DE MÁLAGA

Ports	S1	Connects the eNB with the EPC
Attributes	Location	PoP placing the resource
	IP address	IP address of the eNB to configure the EPC accordingly
Primitives	Reservation	Locates the physical device in the area requested and the specific links' availability
	Activate	Sets the link to allow traffic flow between the eNB and the EPC
	Deactivate	Stops the traffic flow, but maintains the connection and the resource
	Reconfigure	Changes the IP address' value after instantiation
	Query	Provides information on the resource's state
	Release	Deletes the port's link, and returns the resources to the pool

Table 3: Example for the integration of a physical eNB

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

5 PROPOSAL EVALUATION

This chapter describes the design and development of three different experimentation platforms based on the proposed architecture in order to evaluate the feasibility of the proposal to address the second thesis objective, as stated in Section 1.2. These platforms are focused on cellular mobile network research, since its barrier entry is higher than the experimentation on computer networks due to the cost associated to mobile hardware and software and to the lack of access to state-regulated radio spectrum to recreate realistic environments and to use commercial equipment as part of the new developments.

The testbeds presented in this chapter were conceived based on SDN and virtualization techniques, to be able to recreate realistic large-scale networks maintaining this barrier entry at its lower level to provide the researcher community, experimenters and end-users access to low level 4G and 5G resources, network capabilities, and configuration options by combining a pool of heterogeneous resources into a homogeneous interface, without the constraints of a laboratory setting or the rigidity of commercial environments.

Section 5.1 presents the first approach, based on the EuWireless project, which proposes a virtualization based on the deployment of the network components as VNFs. In Section 5.2, the platform is based on CNFs, relying on a more monolithic approach with only one entity in charge of orchestration. Finally, Section 5.3 presents the final architecture of the 5G-EPICENTRE project, which not only deploys CNFs, but also distributes the orchestration among several masters to avoid single point of failure and network overload, among others.

5.1 EUWIRELESS TESTBED

The EuWireless project [115] was envisioned with the purpose of providing researchers a realistic mobile network across Europe to test new developments. The project's main objective was to create a research environment capable of providing access to the usually hidden internal mechanism of the networks but also coexisting with commercial operators, without interfering with commercial exploitation.

In that direction, this section addresses the platform design principles, the resulting design options, and finally the implementation of 5G network slices for research at a large scale reusing a mature technology for automatic creation of network testbeds, namely the GTS [65][156]. Prior to the EuWireless project, the GÉANT research

UNIVERSIDAD DE MÁLAGA

community was using GTS to seamlessly create virtual networks between different locations across Europe.

5.1.1 *Testbed design principles*

Aiming to offer the research community an end-to-end network composed of real resources, the platform relies on the network slicing concept to ensure an abstraction layer that distributes those resources isolated from networks used in other experiments. Bearing this key enabler in mind, experimenters should be able to reserve the resources required for their test from the radio hardware communicating with the UE to the network level, including the core network and any MEC services they may need to perform all kinds of experiments; then run the experiment, and free the resources afterwards. On its side, the platform should be able to provide a private network for each experiment to avoid overlapping or interference between experiments. The design principles identified for creating the EuWireless Testbed in line with the architecture previously proposed are the following:

- Support for concurrent isolated networks, so the infrastructure is able to manage several experiments sharing resources from the physical network in parallel and without interfering with each other.
- Cross-country deployment, to avoid the concentration of research infrastructure around universities and spread the near-access for experimenters across Europe.
- Scalable Points of Deployment as the main core object, to build a distributed architecture that improves scalability and provides high-performance locally.
- Adaptable to integrate new technologies, avoiding relying on current implementations and adding flexibility to adopt new standards and paradigms.
- Fully automated, to ensure the non-exhaustion of resources and experiments maintenance at infrastructure level.

To address the provision of equal opportunities for access and performance to researchers located across Europe, the platform must be decentralized and distributed across PoPs, to enable coverage expansion and scalability. Additionally, it is desirable for these PoPs to be easily connectable with already deployed installations, so as to compose a mesh network of nodes capable of providing the same kind of features to the research community.

All of this suits with the proposal of relying on the PoP as the core object in the EuWireless infrastructure, understanding the PoP as the

UNIVERSIDAD DE MÁLAGA

set of hardware and software required to configure, manage, and run the slice required for an experiment, both as a single node or as a part of a PoP mesh. Ensuring this decentralized interconnection seamlessly will provide the intended scalability, standardizing the interface with the resources available, even if they are geographically located in a different PoP. Additionally, the prospect of the platform growth from an initial Proof-of-Concept (PoC) to large deployments is simplified to deploying and connecting new PoPs, without reconfiguring the whole infrastructure, as depicted in Figure 25.

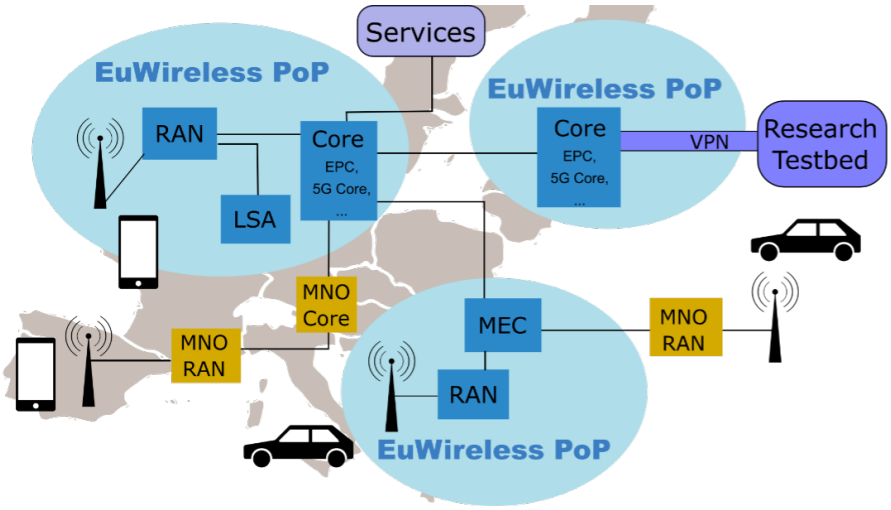

Figure 25: EuWireless high-level deployment overview [115]

Taking all of this into account, the EuWireless testbed architecture must be abstract and expandable enough to ensure a future-proof framework where new technologies can be accommodated to be tested in conjunction with the research on the current paradigm, avoiding a fixed structure from any particular generation of mobile networks. The following subsection discusses some design options for this architecture.

5.1.2 *Testbed design options*

Following the design principles established for the architecture, there are three options that could fit the testbed approach.

The first proposal is to create a full 5G operator owning all the resources required to provide end-to-end network experimentation. The main benefits of this approach are the independence to configure slices and determine the elements required, the provision of direct monitoring and raw performance data from within the network, and the freedom to focus on any layer of the stack. However, the cost of deploying a full operator across Europe and the regulatory and security constraints associated to a multi-country environment makes this proposal unrealistic. Thus, an enhanced first proposal for the

UNIVERSIDAD DE MÁLAGA

testbed architecture, as shown in Figure 26, is owning the core of the network and relying on external operators to ensure network capacity and last-mile access in areas with lower radio coverage.

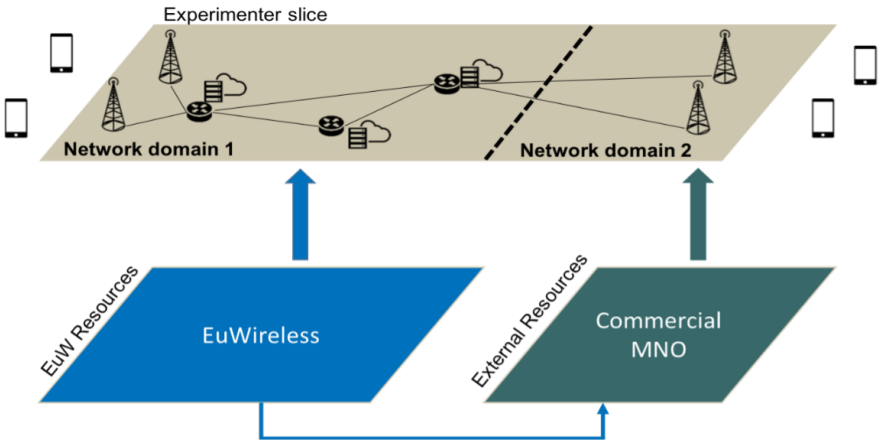

Figure 26: First design option, with owned resources combined with extended coverage provided by commercial MNOs [134]

In contrast, the second design option proposed for the testbed, depicted in Figure 27, is owning only the minimum infrastructure required to provision, configure, and manage the slices, whereas the physical resources are owned by commercial operators.

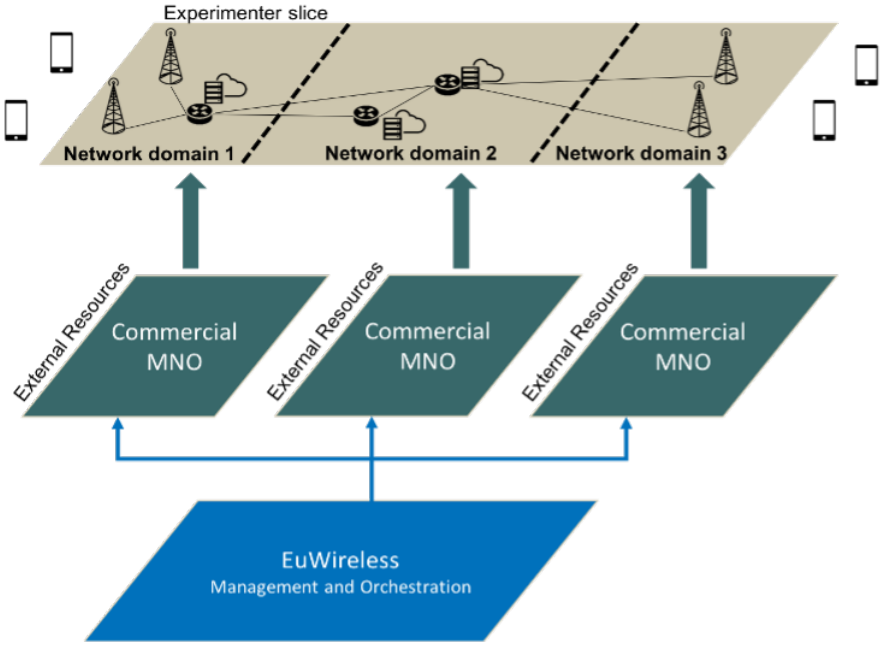

Figure 27: Second design option, managing the resources provided by commercial MNOs [134]

Thus, the testbed acts as a virtual MNO (vMNO) that provides services on top of the commercial operators infrastructure, which could extend the testbed coverage across Europe. With this approach, the

UNIVERSIDAD DE MÁLAGA

testbed would rent the resources required for each experiment, acting as a broker between operators and researchers.

The main drawback is the complete dependence on external operator's resources, which might decrease the QoS offered to the researchers during an experiment for commercial reasons, or limit the access to certain resources and network usage. Additionally, as the only technology currently supporting slices is 5G, experiments would be limited to these resources, and interconnection with different technologies or new paradigms is out of its scope.

However, the previous approaches share a limiting aspect regarding both the low-level components of the network and the technology of the slices provided; this is, providing only 5G slices and restricting access to the low-level components even in the owned infrastructure approach. Bearing this in mind, and the network advances towards virtualization, adding a new level of abstraction to the resources might solve the limitations of both approaches, using the slices themselves as the foundation of a complete network in which it is feasible to deploy and interconnect any network infrastructure as software aggregated in the form of a "raw slice" to support new services. Therefore, there is no longer a restriction related to the technology to experiment, since additional network elements, even entities from different mobile generations, can be instantiated as VNF [89].

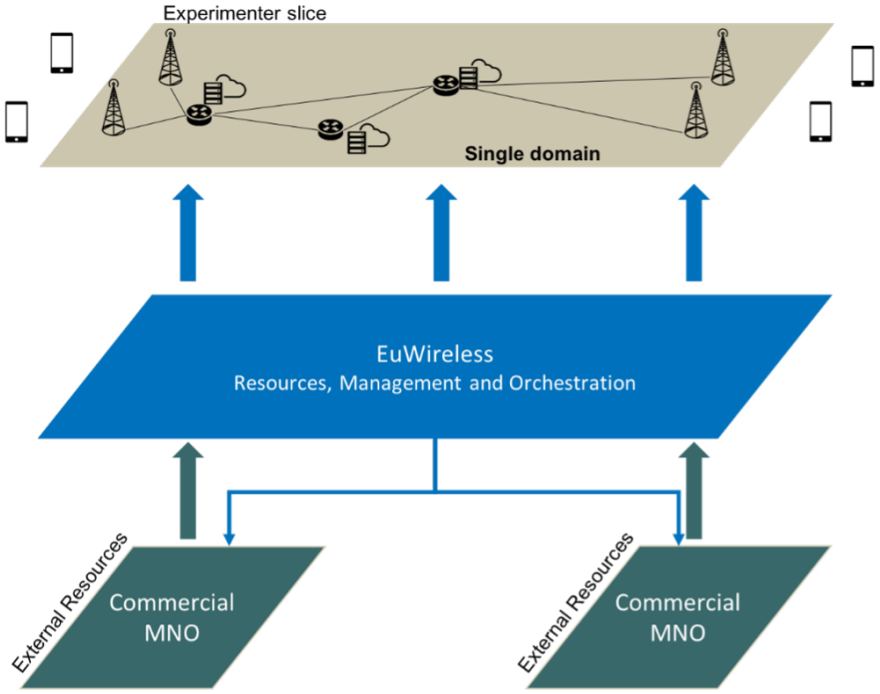

Figure 28: Third design option, with EuWireless acting as slice provider [134]

Moreover, as shown in Figure 28, in this approach the testbed coordination layer is responsible for managing the resources during the

UNIVERSIDAD DE MÁLAGA

experiment lifecycle, so the experiment is independent of the external operators resources status, i.e., if the conditions of the underlying physical network worsen, the coordination layer is able to select a different link transparently to the experimenter.

A single domain can be created for each slice and experiment, and the resources from different operators are treated in a homogeneous way. With this approach, only the infrastructure dedicated to the slice creation and management is out of the experiment scope.

5.1.3 *Testbed implementation*

The third design option allows the most flexibility in the configuration and customization of the network elements, so it is considered the best architecture for the EuWireless testbed infrastructure implementation. For the coordination layer, different standards and technologies available have been evaluated, since it must be able to:

- ensure network isolation among experiments;
- maintain link quality;
- manage and orchestrate different types of resources; and
- offer these resources and capabilities as a single unified slice to the researcher.

In this evaluation, the GTS platform [65][156] emerges as a suitable option for the coordination layer implementation, since it is a network virtualization architecture that provides access to wide area network infrastructure integrated within the GÉANT network footprint.

Combining GTS with the SDN paradigm, it is possible to deploy and refine experimental computer network concepts with real users, network components and conditions at scale. This allocation of wired network infrastructure components to particular projects managed by the experimenter is equivalent in the context of mobile networks to the concept of network slicing, being GTS the slice provider, as presented in Figure 29. Thus, GTS offers dynamic and automated slice provisioning by means of an innovative abstraction layer, that enables a broad range of new network resources to be offered as fully virtualized service objects.

GTS follows a GVM architecture, depicted in Figure 30, which provides an opaque service domain in which the underlying infrastructure supporting the resources provided is hidden to the experimenter. Following this model, the virtual circuits are provisioned with hard QoS guarantees to connect the nodal objects, which might be VMs fine tuned by the experimenter or dedicated hardware platforms, such as bare metal servers, where the user has access to the entire physical server interconnected with a fully virtualized SDN.

UNIVERSIDAD DE MÁLAGA

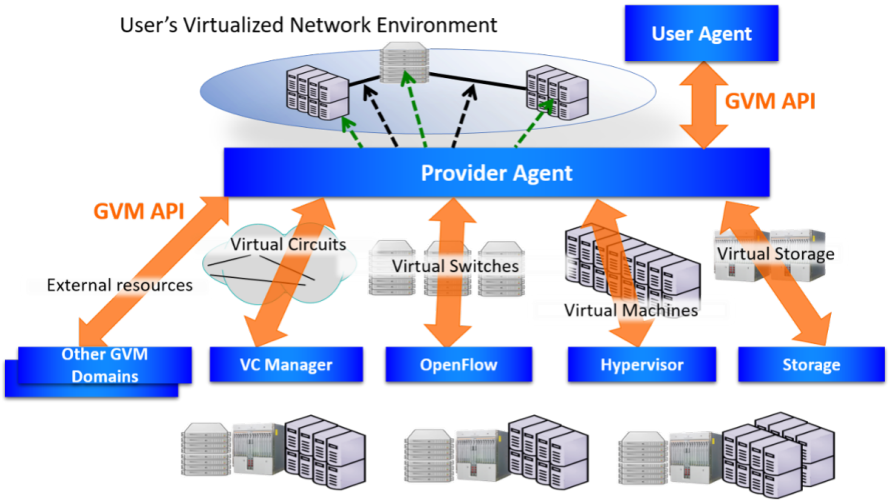

Figure 29: Virtualized network environment provided by GTS [134]

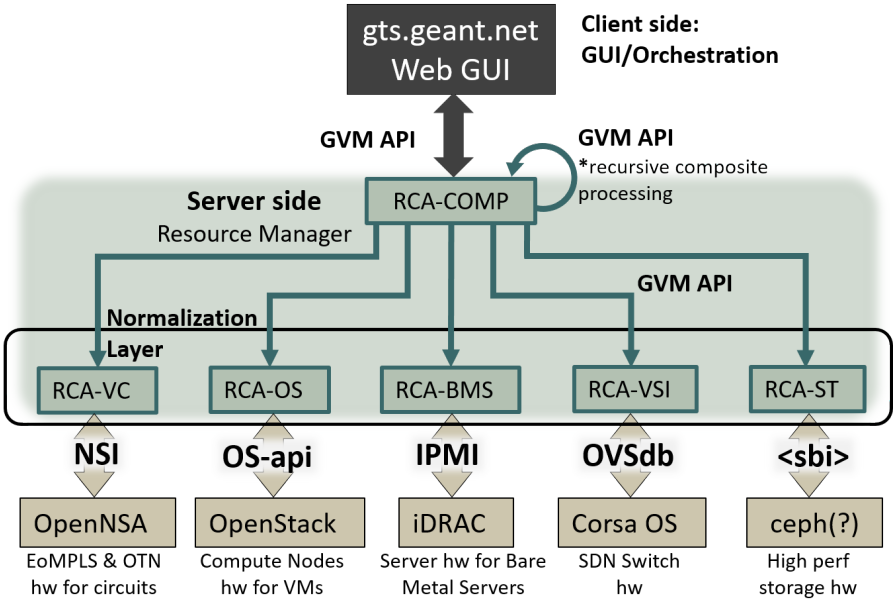

Figure 30: Generalized Virtualization Model [168]

The resources sharing common features are grouped into classes, and each resource class states the behavior and attributes to be specified by the experimenter when defining a resource. Among these attributes, the I/O ports and networking capabilities determine how the resource communicates with other resources. Hence, ports are used to define data paths from/to each resource, sketching the network slice topology by connecting with other ports.

As a result, the definition of a network slice is simplified to specify the set of resources required and the interconnection between them by means of port adjacency. All of these resources are scheduled and placed spatially within the geographic footprint of GTS' reachability.

UNIVERSIDAD DE MÁLAGA

To interact with the objects, the experimenter relies on a set of technology-agnostic primitives to change the state of the resource through its lifecycle, presented in Figure 31. Following the same principles as the Network Service Interface (NSI) [141], this model facilitates architecture escalation, since any resource might be added regardless of the technology simply by delivering its lifecycle definition, allowing the infrastructure provider to maintain control of their infrastructure design and engineering.

As the internal mechanisms for reserving and activating the resources vary for each resource class, GTS implements the so called RCA to adapt these mechanisms to each specific resource class. Consequently, each RCA implements five basic control primitives to move the resource through its states:

reserve To instantiate the resource and allocate the physical infrastructure components. Moves the resource to the reserved state.

activate To configure the hardware and the VNF, moving the resource from reserved to the active state.

deactivate To erase the VNF's configuration and move the resource from active to reserved.

release To shutdown the VNF and destroy the instance.

query To get the resource state information.

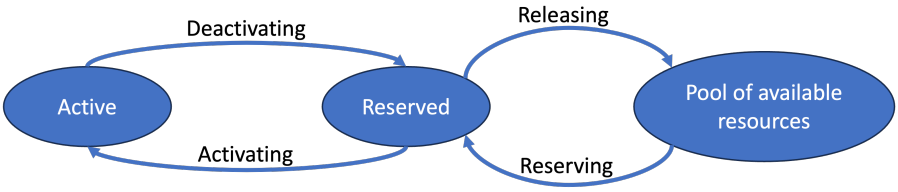

Figure 31: GTS object's lifecycle [134]

Bearing these concepts in mind, the EuWireless coordination layer extends the GTS environment to include 5G network components by defining and implementing their lifecycle and primitives. These components are deployed as virtualized functions that interact with each other through their virtual I/O ports, acting as the components' interfaces, which makes these ports' definition fundamental in each component implementation. Figure 32 depicts this GVM extension to deploy slices with 5G access capabilities.

In the case of the AMF entity, the virtual object includes at least eight ports to represent the following 5G interfaces towards the other 5G components: N1 to connect with the UE, N2 for connection to the RAN, N8 for the UDM, N11 for the SMF, N12 for the AUSF, N14 for connection with other AMF entities in the network, N15 for the

UNIVERSIDAD DE MÁLAGA

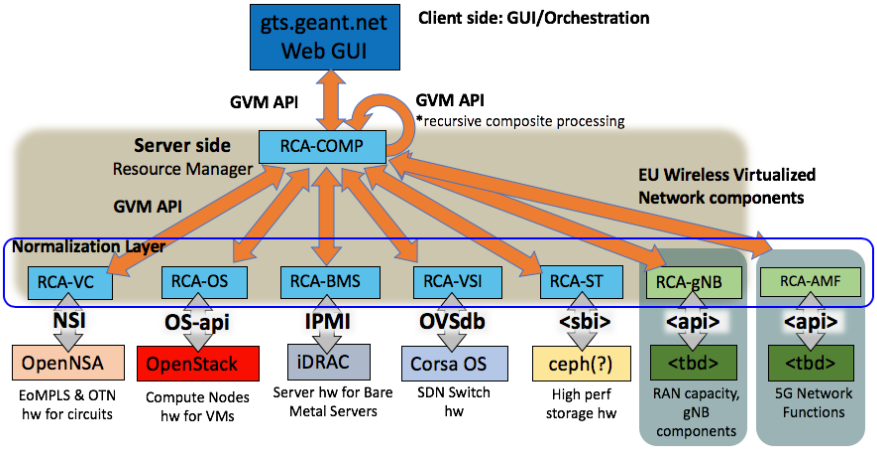

Figure 32: GVM architecture extended with EuWireless RCAs [134]

PCF, and N22 for the NSSF. If the AMF is deployed according to a serviced-based architecture, the ports required are: the Namf for the connection to the control plane, and the N1 and N2 as in the previous case. In both cases, the virtual circuits deployed to link the ports are prepared to transport any protocol the entities require.

Following the approach of using control primitives for the lifecycle, after the resource reservation and initialization, entities are configured and contextualized as network functions included in the 5G slice.

In the case of the gNB, this reservation translates into finding a compute node with capabilities to run the virtual gNB software stack, and with port and spectrum capacities; this is, ensuring the availability of spectrum capacity, and reserving the port capacity towards the IP network and the computing requirements in terms of memory, cores, disk capacity, and specific capabilities to connect to the physical gNB agents. The activation translates into powering on the VMs, establishing bridges to the physical gNB ports, and configuring them for spectrum sharing.

Table 4 presents the actions to be performed in response to each GVM primitive for the main 5G components implemented by the EuWireless testbed as an extension of the GTS environment. To simplify the table, the query primitive is omitted. For every component, the query primitive gets the connection status. Additionally, in the case of the UE, it also gets measurements on connectivity; for the gNB, it gets the logical association status and the number of UE connected; for the AUSF, the registered users; and for the SMF, the traffic routing.

5.1.4 *Testbed experiments*

Following the GTS implementation approach, the proof-of-concept of this testbed integrates radio resources, spectrum sharing, and mo-

UNIVERSIDAD DE MÁLAGA

	Reserve	Activate	Deactivate	Release
UE	Locate a UE and connect	Attach to 5G slice	Detach from 5G slice	Free from testbed
gNB	Ensure the gNB availability	Connect N2, N3 interfaces. Start PLMN broadcast	Close logical connection to AMF, UPF	Return resource to pool
AMF	Instantiate AMF VNF	Connect N8,N11, N12 interfaces	Disconnect the interfaces	Shutdown VNF
UPF	Instantiate UPF VNF	Connect N4, N6 interfaces	Disconnect the interfaces	Shutdown VNF
AUSF	Instantiate AUSF VNF	Connect to AMF. Register users	Disconnect the interfaces	Shutdown VNF
SMF	Instantiate SMF VNF	Connect to AMF, UPF. Configure UE's IP pool	Disconnect the interfaces	Shutdown VNF

Table 4: Primitives definition for 5G components [134]

bile network entities into an homogeneous user interface mapping the EuWireless concepts to the GVM architecture, to show the feasibility of deploying EuWireless 5G slices that meet the testbed design requirements. This proof-of-concept consists on the deployment of a single PoP located at the University of Malaga premises. Different experiments were carried out to demonstrate the fulfillment of the design objectives:

- Isolated and concurrent slices support;
- Remote access guarantee for the platform users;
- Automated process for slice provisioning; and
- Possibility of multi-domain and mobile technologies resources integration.

The PoP is an isolated GTS point of deployment, extended with two cellular resources (the EPC and the eNB) that combined with the already existing GTS' resources, enable the remote deployment of a a full operative LTE slice on top of the PoP. Besides the PoP to provide the infrastructure, a computer is required to remotely design and manage the slices, and a pair of UEs provisioned with SIM cards to test the proper functioning of the slice. As shown in Figure 33, the

UNIVERSIDAD DE MÁLAGA

researcher is able to deploy multiple slices to run concurrently over the same underlying infrastructure. In this case, as there is only one eNB available on the testbed, the other two slices include other types of resources, such as VMs and virtual circuits.

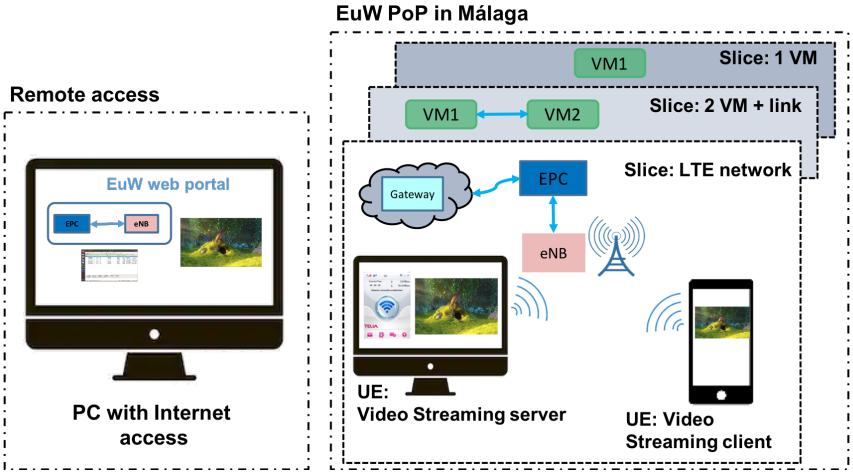

Figure 33: Experiment performed as Proof-of-Concept [169]

The remote access to the testbed by the experimenters is granted through the EuWireless Portal, which is a web portal that runs on top of the PoP and provides authentication to the EuWireless users. This portal is depicted in Figure 34, with the three slices deployed: the first slice running an isolated VM (Provider ID 483), the second running two VMs connected through a link (Provider ID 484), and the third one running the fully operative LTE network (Provider ID 486).

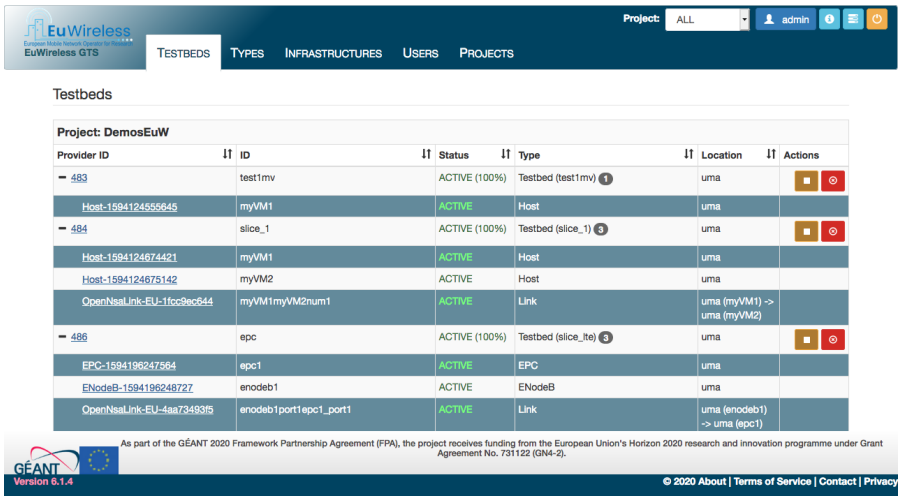

Figure 34: EuWireless Portal over GTS [169]

Another feature, even though is not explicit in Figure 34, is the capability to include resources from different administrative domains. In this case, the eNB is an external domain, so it can be integrated into the slice and connected with the rest of the resources by means of a

UNIVERSIDAD DE MÁLAGA

link, but its low-level configuration, such as its IP address, is external to the testbed, which owns the links and the EPC.

To define the slice, the experimenter must follow a language based on GTS' domain-specific language to describe the low-level configuration of the resources owned by the testbed. Figure 35 presents the definition of the LTE slice, including the eNB as an external domain resource with only a port to connect it with the rest of the slice; and the configuration required by the EPC, i.e., the IP pool that specifies the set of addresses available to assign to the UEs, the MME IP address for the S1 interface, the list of accepted IMSIs, the MNO's PLMN, the TAC, and the authentication parameters. Since the EPC is deployed as a VM, its fine tuning also includes the location of the compute node, the image used and its flavour.

As GTS is responsible for the automatic slice provisioning and deployment, the last step of the proof-of-concept is testing the LTE slice performance, since the other two slices only include validated resources from the GTS legacy environment.

In [125], the authors present a design to evaluate the performance of a service running in a mobile network. Following the same procedure, a point-to-point video streaming session over the slice using two mobile subscribers is set to test the slice performance. These subscribers are in the eNB coverage area and communicate through the EPC deployed on the Malaga PoP; and their UEs are a computer with an LTE modem and an Android mobile phone, both with EuWireless SIM cards. By using the VLC media player software, the computer acts as the streaming server and the mobile phone as the client.

By monitoring the streaming sessions with the Wireshark packet sniffer running in the EPC VM (Figure 36), the attach procedure of one of the UEs is observed. In the figure, we can also find the exchange between the MME, with IP address 10.102.81.35, as defined in Figure 35, and the eNB, with IP address 10.102.81.60 (configured externally by the resource owner).

The last set of messages depicted correspond to the user registration performed after the attach procedure, with IP address 45.45.0.10 from the IP pool defined in Figure 35. Once both UEs are registered, the streaming session can start, checking the video is indeed properly displayed on the client side with an average throughput of 14 Mbps for video transmission with 50 seconds of duration and a segment length of 1460 Bytes.

This traffic and the correct visualization of the video on the client UE demonstrate that the LTE network deployed is fully operative.

Finally, to demonstrate the slice isolation and performance preservation regardless of the slice running concurrently, the 45.45.0.0/16 network was configured on the other two slices without LTE resources and several transmissions were conducted.

UNIVERSIDAD DE MÁLAGA

```
slice_lte {
  id = "epc"
  epc { id = "epc1"
        location = "uma"
        imageId = "NextEPC.vmdk"
        flavorId = "c2r8h25"

        ipS1addr = "10.102.81.35"
        ipS1mask = "255.255.255.0"
        IMSIs = "001010000012305 001010000012378"
        plmn { mcc = "001"
               mnc = "01"
        }
        apn = "euwireless.apn"
        tac = 1
        uePool = "45.45.0.0/16"
        authentication {
            akey = "00112233445566778899aabbccddeeff"
            encryp = true
            OPc = "000102030405060708090a0b0c0d0e0f"
        }
        port { id="port1" }
  }

  enodeb { id="enodeb1"
               port { id="port1" }
  }

  connectLink = { p1,p2 ->
        def lnk = link {
           id = "${p1.parent.id}${p1.id}${p2.parent.id}_${p2.id}"
           port { id = "src" }
           port { id = "dst" }
        }
        adjacency p1, lnk.src
        adjacency p2, lnk.dst
        lnk
  }
  connectLink(enodeb1.port1, epc1.port1)
}
```

Figure 35: LTE slice description [169]

The first test, to check the UEs reachability from the other slices' VMs, resulted negative, proving that despite the three slices share the same network and physical infrastructure, isolation is achieved.

The results from the second test, to verify the slice performance regardless of sharing the infrastructure, are represented in Figure 37. This test consisted in using *iPerf3* for TCP and User Datagram Protocol (UDP) transmissions while changing the number of active slices to

UNIVERSIDAD DE MÁLAGA

Time	Source	Destination	Protocol	Length	Info
220.299705…	10.102.81.60	10.102.81.35	S1AP/NAS-EPS	166	InitialUEMessage, Attach request, PDN connectivity request
220.321442…	10.102.81.35	10.102.81.60	S1AP/NAS-EPS	138	DownlinkNASTransport, Authentication request
220.417123…	45.45.0.9	192.168.43.139	GTP <TCP>	110	63655 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM
220.479818…	10.102.81.60	10.102.81.35	S1AP/NAS-EPS	138	UplinkNASTransport, Authentication response
220.480381…	10.102.81.35	10.102.81.60	S1AP/NAS-EPS	118	DownlinkNASTransport, Security mode command
220.519785…	10.102.81.60	10.102.81.35	S1AP/NAS-EPS	134	UplinkNASTransport, Security mode complete
220.529762…	10.102.81.35	10.102.81.60	S1AP/NAS-EPS	286	InitialContextSetupRequest, Attach accept, Activate default EPS
220.600774…	10.102.81.60	10.102.81.35	S1AP	118	InitialContextSetupResponse
220.600869…	10.102.81.60	10.102.81.35	S1AP/NAS-EPS	122	UplinkNASTransport, Attach complete, Activate default EPS beare
220.600908…	10.102.81.35	10.102.81.60	SCTP	62	SACK
220.601661…	10.102.81.35	10.102.81.60	S1AP/NAS-EPS	142	DownlinkNASTransport, EMM information
220.737207…	45.45.0.9	8.8.8.8	GTP <DNS>	119	Standard query 0x0d8a A [redacted].com
220.746507…	45.45.0.9	8.8.4.4	GTP <DNS>	119	Standard query 0x0d8a A [redacted].com
220.794238…	10.102.81.60	10.102.81.35	SCTP	62	SACK
222.307178…	45.45.0.10	45.45.0.255	GTP <NBNS>	146	Registration NB DESKTOP-[redacted]<20>
222.307204…	45.45.0.10	45.45.0.255	GTP <NBNS>	146	Registration NB DESKTOP-[redacted]<00>
222.307215…	45.45.0.10	45.45.0.255	GTP <NBNS>	146	Registration NB WORKGROUP<00>

Figure 36: UE attach and registration [169]

check its effect on the transmission performance. For the TCP case, due to the iPerf implementation, the bandwidth fluctuated between 0 and 6 Mbps, whereas for the UDP, the average bandwidth was the specified in the iPerf test.

Thus, three transmissions, one TCP with 0-6 Mbps, one UDP with 10 Mbps, and another UDP with 15 Mbps, were performed twice each: firstly, running only the slice being tested, and secondly, with another slice running in parallel and exchanging traffic at 1 Gbps.

Figure 37 demonstrates that the slice performance is unaffected for the UDP case, in which the transmission bandwidth is fixed through the *iPerf3* tool, depicting no changes in the average bandwidth when the second slice is running. For the TCP case, the slice performance is kept between 0 and 4 Mbps. This is a consequence of the transmission bandwidth variation, due to the tool implementation and its TCP window size, which adjusts automatically and is affected by the second transmission running in parallel.

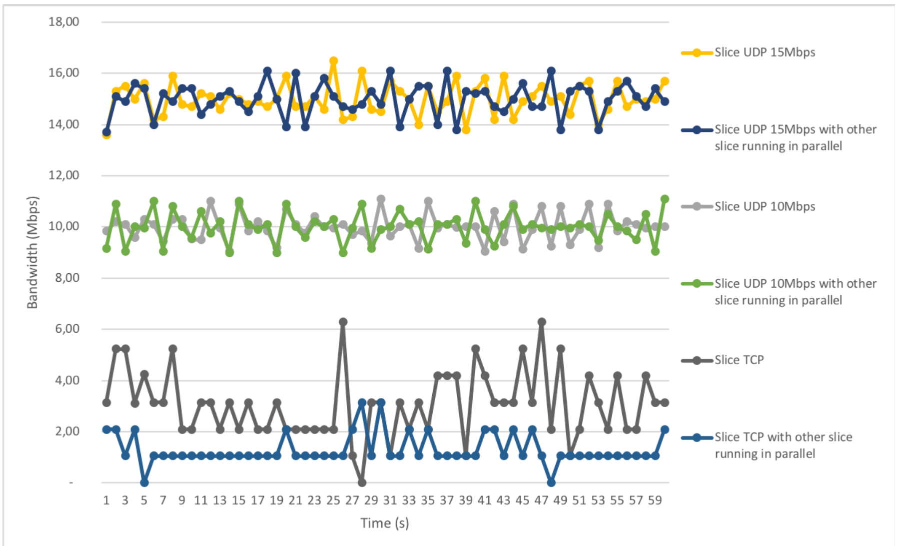

Figure 37: Bandwidth obtained in the TCP and UDP transmissions [169]

UNIVERSIDAD DE MÁLAGA

5.2 5G-EPICENTRE MALAGA TESTBED (EARLY STAGE)

The 5G-EPICENTRE project aims at facilitating experimentation on the Public Protection and Disaster Relief (PPDR) vertical by providing an open-standard, innovative and inter-operable end-to-end 5G experimentation ecosystem that meets the requirements of the different PPDR operational scenarios. This testbed evaluates the feasibility of the architecture proposed in Chapter 4 in a container-based environment. To that end, the platform follows a cloud-native approach with a service-oriented architecture to ensure agile developments and deployments while maintaining its compliance to a wide range of vertical solutions.

This section addresses the platform design principles, main design option, and the testbed's first implementation, followed by a proof-of-concept of a very early stage. The platform final architecture definition, deployment, and validation is presented on subsequent Section 5.3.

5.2.1 *Testbed design principles*

The cloud-native approach is the way of designing, building, and running applications based on the cloud computing paradigm to provide agile, scalable, and quickly made and modified deployments that connect easily with other services and applications. In this context, some enablers to develop cloud-native applications are the continuous integration model, the different container engines, and the system orchestrators.

However, the most common virtualization of network functions consists on the encapsulation of those functions as VNFs, not only in the context of software development but also in the telecommunications systems, since the transformation from a PNF into a VM is easier than containerization. The 5G-EPICENTRE project, in contrast, follows the containerization approach, based on the following expected advantages over the classical ETSI MANO reference architecture:

- Flexibility to adapt to changing network requirements, which increases innovation prospects by deploying and getting to market rapidly new services and applications.
- Scalability to meet the requirements of novel 5G services.
- Automatic provisioning and orchestration of network function and services to improve system efficiency.
- Dynamic network slicing to cost-effectively deploy VNF and CNF.
- Cost-effectiveness not only in deployments, but also reducing CapEx and OpEx compared to traditional network infrastructures.

UNIVERSIDAD DE MÁLAGA

Since the cloud-native technology enables an easier and more efficient and reliable development, deployment, and management of services, it is considered a key enabler in driving the adoption and growth of 5G and beyond networks and services, which implies the main requirement for this platform is being designed as a container-based architecture with the previous advantages taken as design requirements for the testbed.

5.2.2 *Testbed design options*

As mentioned in the design principles definition, this platform is required to accommodate a container-based infrastructure to manage and orchestrate the deployment and operation of containerized applications from the PPDR vertical. The underlying 5G infrastructure is provided by an already existing testbed with 5G capabilities, in this case the 5GPPP 5GENESIS project, further described in [98], which in its Malaga platform includes a fully virtualized core network combined with 5G NR for indoor and outdoor deployments.

This design option, as depicted in Figure 38, is similar to the design reached with the EuWireless project, with the difference of having a full 5G testbed as underlying infrastructure instead of a MNO, and a cloud-native orchestration layer acting as the middleware between the experimenter and the 5G underlying testbed.

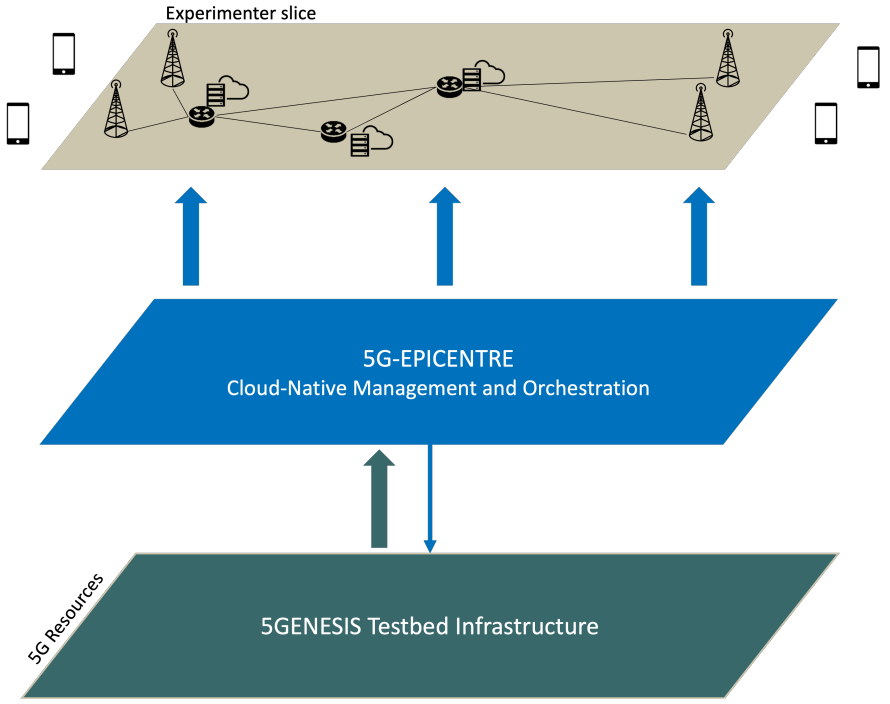

Figure 38: Design option, as a cloud-native slice provider

UNIVERSIDAD DE MÁLAGA

5.2.3 *Testbed implementation*

Taking into account the enhancements to the reference NFV architecture when deploying microservices containerized, the ETSI GR NFV-IFA 029 [59] updated the NFV-MANO architecture to support CNFs by adding a container runtime environment called the Container Infrastructure Service (CIS), a container image repository of images called the Container Image Registry (CIR), and a manager to deploy and monitor the containerized services called the CIS Management (CISM). Since K8s is considered to be MANO-compliant, it is possible to map its exposed services and master-worker architecture to these new components. Furthermore, the work presented in [26, 93, 101, 106, 129] demonstrates that K8s is the most likely orchestrator for beyond 5G networks, since it is able to fulfill both the CISM, and the VIM and VNFM functionalities.

Hence, regarding the implementation of the cloud-native layer, and bearing in mind that K8s is the de facto standard orchestration system for containerized applications, the first stage of the platform is deployed as an elementary K8s-based environment, with only one master and two worker nodes, simply to prove the feasibility of the containerization approach. The K8s cluster connects with the external networks and the 5GC Standalone instance, with both user and control planes configured, that attaches to the existing 5GENESIS radio resources. Figure 39 displays this first approach of the K8s-based architecture implemented in Malaga.

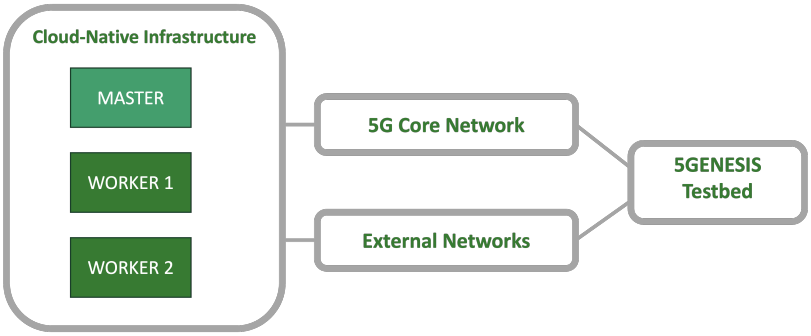

Figure 39: Malaga Platform K8s-based architecture (first stage)

5.2.4 *Testbed experiments*

Considering this is an early stage of the architecture, the proof-of-concept aims at proving the cloud-native approach feasibility in the context of the PPDR vertical. To this end, *RedZinc's BlueEye Handsfree*[1], which is a wearable video solution for telemedicine applications, was deployed on top of the K8s cluster. The solution consist of a head-

1 https://redzinc.net

UNIVERSIDAD DE MÁLAGA

set including a video camera that sends live point of view video to remote doctors to oversight the patient on their way to the hospital.

The connection between the application on the headset and the doctor's hotdesk is established via Traversal Using Relay NAT (TURN) servers placed between the application and the video server, and between the video server and the hotdesk. The video server handles the unidirectional video from the application and to the hotdesk, and the bidirectional audio and data from and to both ends, sending those media connctions through the TURN servers, as depicted in Figure 40, extracted from [49].

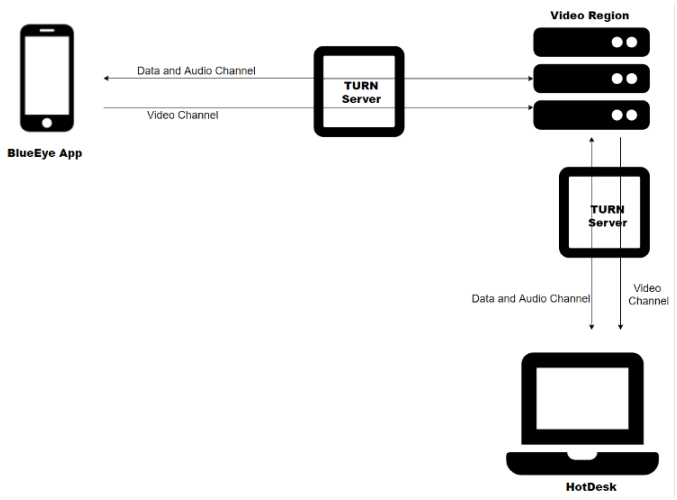

Figure 40: BlueEye Application building blocks [49]

Concerning the cloud-based deployment, the service relays on the following components:

- The Application Management hosted permanently in the cloud.
- The Video Region, deployed on demand on top of the K8s infrastructure, containing both the TURN servers and the video server containerized.
- The *BlueEye* application on the end devices, used to access the Application Management via external networks to log in the service.

Hence, the *BlueEye* application sends the video and audio from the camera to the video region, who establishes the media connection with the hotdesk, as shown in Figure 41. Containerizing the video region provides scalability and allows critical communications on the move by automizing the deployment process at the edge. A description of the video region's deployment and service is applied through the K8s master, in which both the TURN server and the video server are defined as containers that conform one deployment. Figure 42 shows the description of the service in YAML format and Figure 43 shows the deployment's one.

UNIVERSIDAD DE MÁLAGA

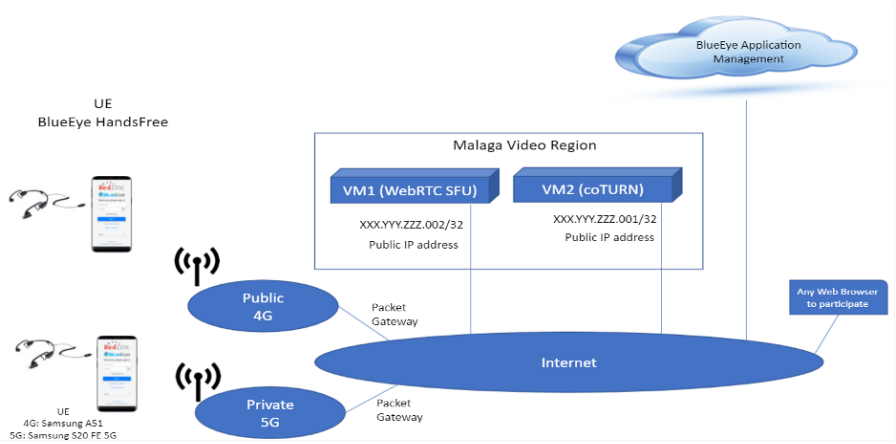

Figure 41: BlueEye Application deployed as Proof-of-Concept (PoC) [49]

```
apiVersion: v1
kind: Service
metadata:
  name: blueeye-service
  annotations:
    metallb.universe.tf/address-pool: mobile-network
spec:
  type: LoadBalancer
  selector:
    app: blueeye
  ports:
  - name: tcp9
    port: 9443
    targetPort: 9443
    protocol: TCP
  - name: tcp
    port: 443
    targetPort: 443
    protocol: TCP
  - name: udp20000
    port: 20000
    protocol: UDP
    targetPort: 20000
```

Figure 42: K8s-based BlueEye video region service

The images used to create the containers are stored and managed in the centralized project's image repository. The service that exposes the deployment to the external network provides the IP address from a pool of addresses available for K8s to use from within the 5GENESIS mobile network, to ensure connection with the end devices using the *BlueEye Handsfree* equipment.

As mentioned early, this proof-of-concept was mainly to show the feasibility of containerizing a PPDR application and its deployment on

UNIVERSIDAD DE MÁLAGA

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: blueeye-deployment
  labels:
    app: blueeye
spec:
  selector:
    matchLabels:
      app: blueeye
  template:
    metadata:
      labels:
        app: blueeye
    spec:
      containers:
      - name: video
        image: registry.blueeye.video:60600/services/video/5g-
            epicentre/vnf:latest
        ports:
        - name: puertotcp
          containerPort: 443
          protocol: TCP
        - name: puertoudp20000
          containerPort: 20000
          protocol: UDP
      - name: turn
        image: registry.blueeye.video:60600/services/turn/5g-
            epicentre/vnf:latest
        ports:
        - name: puertotcp
          containerPort: 9443
          protocol: TCP
      imagePullSecrets:
      - name: redzinc
```

Figure 43: K8s-based BlueEye video region deployment

the fly on top of a K8s-based infrastructure taking mobile resources from an already working 5G testbed. This was successfully achieved when connection between the *BlueEye* end devices was established and media was bidirectionally transmitted among them. Additionally, there were some other KPI achieved, such as maintaining the video quality better than 640x480 during 90% of the time, getting less than 5% of data packet loss, deploying the application in less than 60 seconds, and getting less than 50% of infrastructure loads in terms of CPU, RAM, and disk usage.

UNIVERSIDAD DE MÁLAGA

5.3 5G-EPICENTRE TESTBED

With the feasibility of cloud-nativizing the network proven, this section presents the final architecture of the 5G-EPICENTRE project platform located in Malaga, its implementation as a distributed orchestration system, and an advanced proof-of-concept with a complex application deployed as a chain of microservices.

5.3.1 *Testbed architectural design*

To realistically cloud-nativize the network, it is required designing an architecture that supports both the NFV-MANO reference architecture and the 5G testbed resources, while integrating the CISM functionality via K8s. This design, adapting the previous reference from a containerized point of view, is presented in Figure 44.

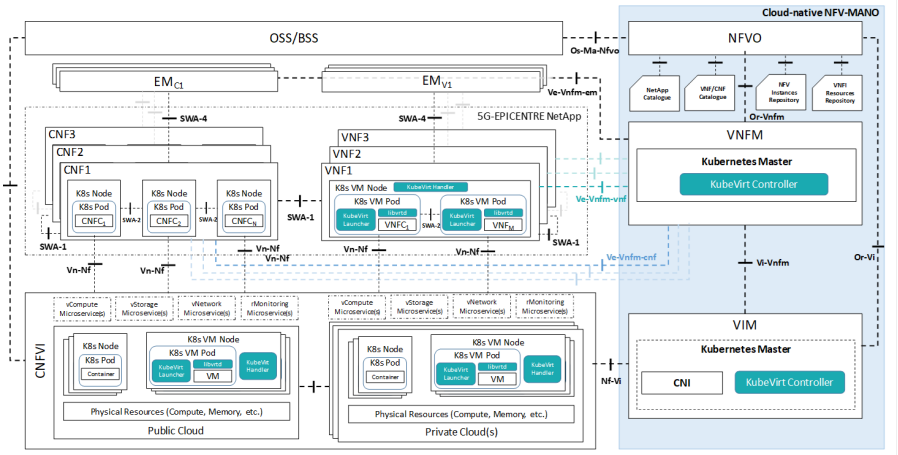

Figure 44: Reference implementation proposal for cloud-native NFV-MANO [113]

This architecture makes use of K8s to manage both the VNFs and the CNFs by integrating the CISM functionality into the VIM and the VNFM. Hence, it has capacity to host 5G cloud-native Network Application (NetApp)s composed by a chain combining containers and VMs. This is possible due to K8s add-on components created to manage VM's lifecycles.

The left side depicts a container-based platform abstracting the underlying infrastructure and providing support for NetApp deployments in the form of microservices chain. On the right part of the figure, based on the work in [24], we can observe the cloud-native NFV-MANO implementation building blocks, which also includes an update of the reference points to manage the K8s-based VNFM and the lifecycle of the NFs. Hence, a K8s master is mapped to the VNFM block in charge of scheduling, management, and deployment of NetApps, and to the VIM block, for virtulized resources management.

UNIVERSIDAD DE MÁLAGA

Following the same augmentation for the Cloud-native NFV Infrastructure (CNFVI), it should be composed of the containerization layer on top of the cloud resources, as proposed in [43]. To increase the granularity, the VNFs should be divided into a chain of components, following a microservices-based approach in which each component is encapsulated into a separated container, either by building new microservices, or by decomposing VNFs into smaller functional entities deployed as microservices [45]. To keep these VNFs' chains aligned to the ETSI VNF reference architecture, the CNFs expose their interfaces to the NFVI to access the resources, to the VNFM for the lifecycle management, and to the other CNFs to compose the chain abstracting the 5G NetApp [22, 126].

The cloud-native NFV-MANO is required to maintain a backward compatibility with VM-based deployments, and manage workloads of both VMs and containers combined. To this end, as K8s is unable to support VM orchestration as it is, the KubeVirt add-on is used for VM management inside the K8s-based cluster. Combining K8s with this add-on, the CISM functionality is easily embedded into the ETSI NFV-MANO reference architecture, as shown in Figure 45, which is an adaptation of the reference architecture from [70].

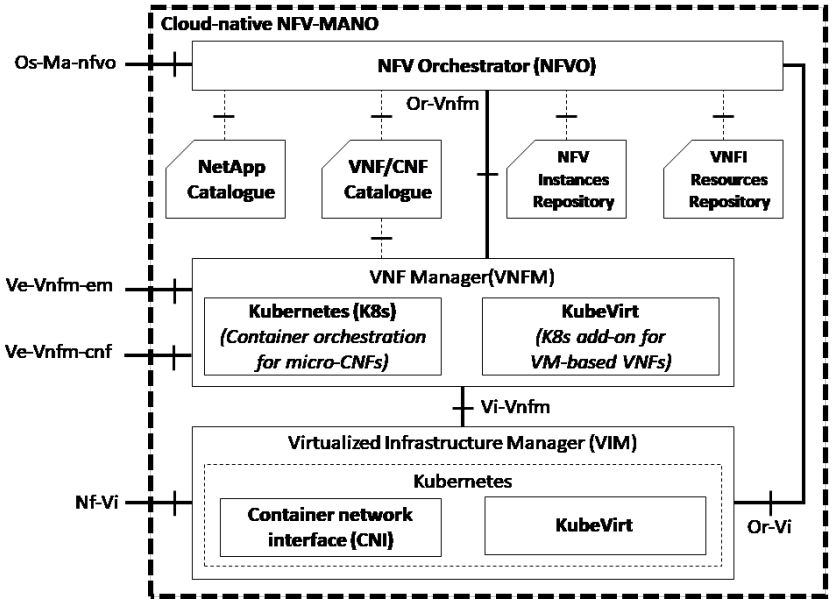

Figure 45: CISM functionality fixed into the ETSI NFV-MANO reference architecture [113]

In conclusion, to create a manageable and robust network in which services are redundant and highly available, the functions should be divided intro smaller entities, so-called microservices, which usually are encapsulated into containers.

UNIVERSIDAD DE MÁLAGA

5.3.2 *Testbed implementation*

Microservices orchestration and containerization itself are not trivial tasks. For the implementation of this testbed, again K8s is used, since it simplifies the management of large and dynamic systems while providing a framework to run distributed and scalable systems resiliently and with failover capabilities. Nevertheless, in this case the K8s deployment is distributed, with a multi-master multi-node architecture to provide high-availability and meet the reliability requirements of PPDR services while addressing high-demand operations of 5G applications and technologies.

As depicted in Figure 46, the infrastructure deployed includes in its control plane a total of three master nodes with a stacked etcd topology. Physically, this translates into a deployment with three different hardware nodes, one acting as the main data center and two acting as edge nodes, so services are distributed across different locations based on the experiment requirements. Regarding the workers, this deployment includes three nodes, distributed as the masters across the physical nodes. Additionally, a fourth node dedicated to the storage is connected to the rest of the infrastructure as another worker, with lower processing capabilities but higher storage available for applications with persistent storing requirements. The requests received from the researchers are handled by the K8s API, which includes a high availability proxy and a load balancer, to ensure the correct load distribution among the control plane nodes and the worker nodes.

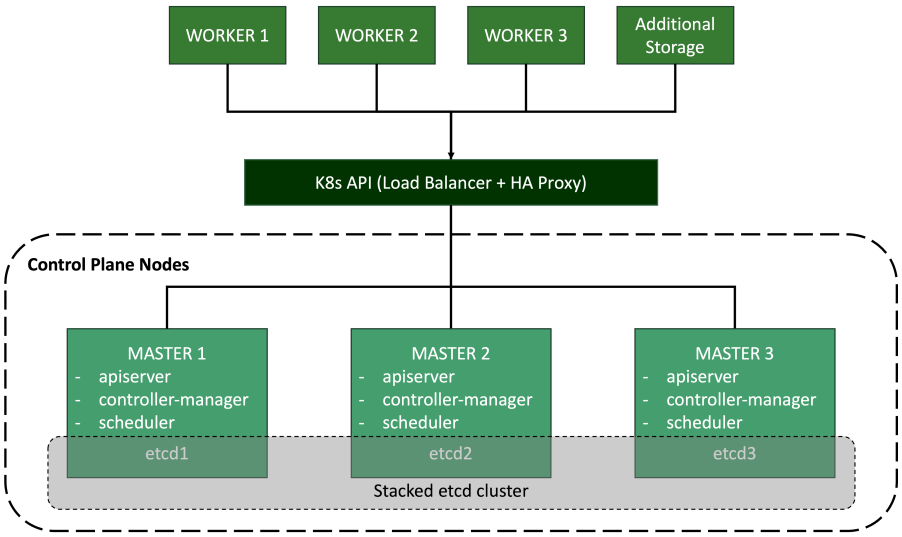

Figure 46: K8s-based distributed infrastructure deployed

To follow the reference implementation proposed in the previous section, this infrastructure orchestrates both CNFs and VNFs by means of combining *Docker* as the container runtime with *KubeVirt* as the VMs workload manager. To retrieve data from the NFVI and the core

UNIVERSIDAD DE MÁLAGA

network for KPIs monitoring, an instance of *Prometheus* is deployed containerized inside the worker plane. *Prometheus* is an open source monitoring framework that retrieves metrics from any resource in the infrastructure via node exporters, and includes an *Alert Manager* service for the management of system alerts derived from that monitoring. In this architecture, a *RabbitMQ* message broker connects to *Prometheus* and publishes the data retrieved from the experiments via MQTT queues.

For the container network implementation, the open source project *Calico* is used. *Calico* provides an scalable solution for K8s networking by relying on a flat, non-overlay network model to route traffic and provide network-level security for containers and applications. In addition to *Calico*, *MetalLB* is used for load balancing since the infrastructure is deployed on top of a bare-metal hardware; whereas the *NGINX Ingress controller* handles the access control. *MetalLB* is an open source load balancer driver used in systems bare-metal that do not support Layer 2 load balancing to provide external access to the services in the K8s cluster; and the *NGINX Ingress controller* provides load balancing and custom traffic routes for services running on the cluster, optimizing the cluster network traffic.

Regarding the storage, it is dynamically provisioned by combining the open source distributed file system *GlusterFS* with the *Heketi* tool, that simplifies the administration of *GlusterFS* by allowing the creation and management of volumes in the K8s cluster. Hence, the storage is distributed not only among the designated storage nodes but also among workers to guarantee scalability, geographical diversity, and minimal downtime in case of failure.

For the resources management, the command-line interface *kubectl* is used to interact with the K8s cluster via the API. Thus, any researcher with access to the infrastructure is able to manage the resources from their experiment via *kubectl*. For security reasons, this access to the infrastructure is based on an authorization protocol combined with a role-based access scheme, which differentiates each experimenter's functions and prevents them from using any cluster administration function. Additionally, the Role-Based Access Control (RBAC) works in association with the organization of different namespaces to guarantee isolation between the experimenter's environments, to avoid collateral issues from one experiment to affect other researchers' deployments. To this end, the monitoring tools, the cluster system, and the experimenters each have a different namespace, so that isolation and network slicing are met.

Figure 47 presents the testbed's final K8s-based architecture, in which the NFVI deployed connects to the external network, the existing 5G resources from the 5GENESIS testbed, and the 5G core network, which contains an *Athonet* 5GC Standalone instance software-based and fully virtualized, with both user and control planes configured, and at-

UNIVERSIDAD DE MÁLAGA

tached to the existing 5GENESIS radio resources available at the testbed location.

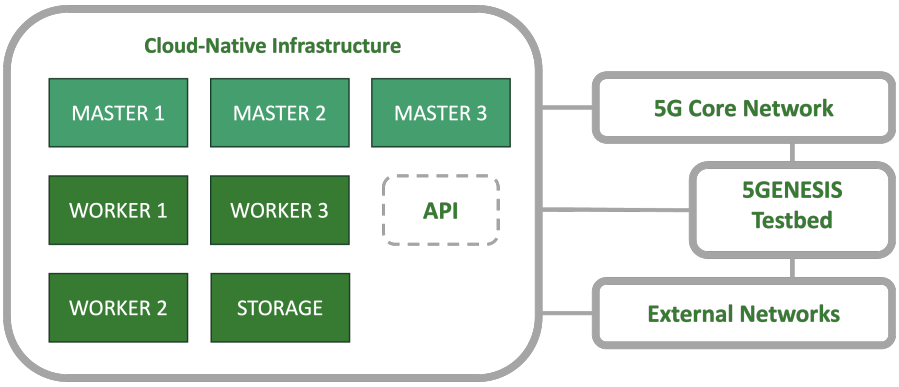

Figure 47: Malaga Platform K8s-based architecture (final stage)

5.3.3 *Testbed experiments*

The deployment of a novel 5G PPDR communications system such as *Mobitrust*[2], a situational awareness NetApp, can be used to validate the implementation approach through experimentation. *Mobitrust* provides support to Command and Control Center (CCC) operations in the field by using end-user devices to monitor, retrieve, and collect data from diverse sources that is then transmitted over the 5G network and processed in the CCC to bring situational awareness in PPDR operations. In Figure 48, this architecture is presented with a end-user retrieving data by means of a body-kit equipped with sensors and real-time text, audio, and video transmission capabilities.

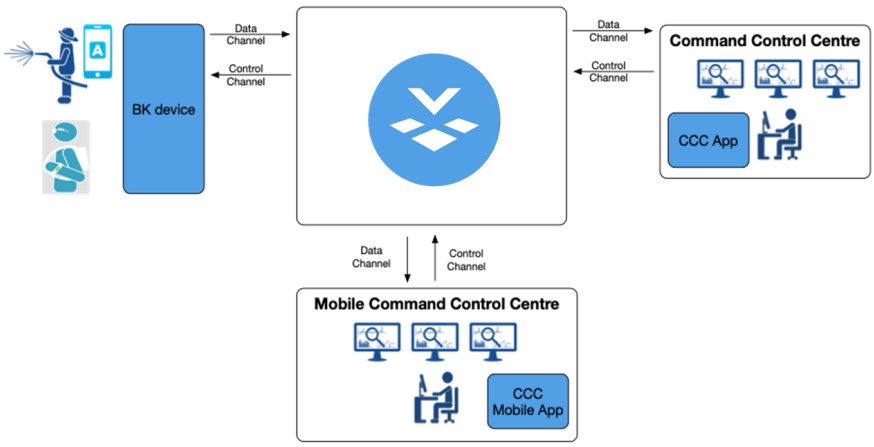

Figure 48: Mobitrust Application Architecture [113]

The application can be divided into a set of microservices to be deployed as containers on top of the cloud infrastructure, guaranteeing

2 https://mobitrust.onesource.pt

UNIVERSIDAD DE MÁLAGA

the availability and scalability of the system. As depicted in Figure 49, the microservices composing the *Mobitrust* application are the following:

- The Portal: it is the frontend entity for the desktop users at the CCC to interact with the system through a browser. In the case of mobile users, the portal is replaced by the operational application.
- The Gateway: it is the component interfacing between the previous CCC frontend entities and the system backend. In the case of end-user devices, this interface to the backend is provided by the message broker.
- The WebRTC server: it provides support to the end-users for audio and video streaming to the CCC components.
- The Monitor: it perform health checks on the end-users devices to provide status updates and trigger alarms on the CCC.
- The Orchestrator: it is responsible for the users' authentication and authorization.
- The TICK stack: it handles the metrics collection, computing, and reporting tasks. It has been further divided into the stack components to be deployed individually, this is the Telegraf, InfluxDB, and Kapacitor entities.
- The PostgreSQL: it is a relational database to store information on the users, their access, and their transmissions.
- The End-user device simulator: it is used to validate the application deployment by simulating end-user tasks, such as authenticating, providing sensor values, and transmitting video.

Additionally, an ingress controller is used in combination with a load balancer to map the CCC access to the application microservices.

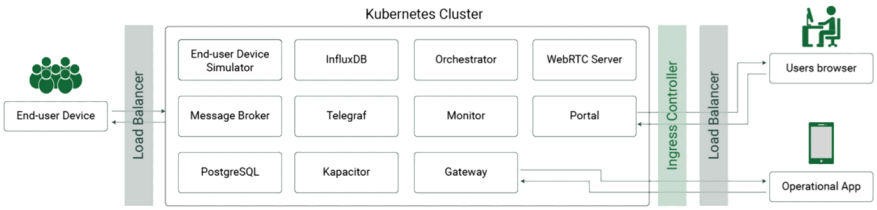

Figure 49: Mobitrust Application deployed as Proof-of-Concept

With regards to the K8s deployment, these entities are described in a set of *YAML* files that define their container requirements and configurations in terms of configmaps, secrets, volume claims, services, deployments, and ingress controllers. For the containers, the

UNIVERSIDAD DE MÁLAGA

images of each entity are available in a private Docker registry repository. The configmaps include the components settings. The secrets are used to prevent confidential data from showing on the NetApp code, such as the the access to the repository to retrieve the container images or the *Mobitrust* service's Transport Layer Security (TLS) certificate. The volume claims are used to request persistent storage for entities such as the Telegraf to store the metrics retrieved, or the PostgreSQL for the users information. The services expose and map the application's ports to the specific container's ports. The set of containers required per microservice compose the deployments, together with the information on how to deploy those containers inside the cluster or the number of replicas required. The ingress routes the external connections to the internal services.

Hence, the experimenter deploying the *Mobitrust* application creates the *YAML* files with the definition of the resources required and applies them by using kubectl commands to attack the K8s API. The *MetalLB* load balancer handles the experimenter requests in terms of resource distribution, the *Calico* service establishes the network components to communicate the microservices composing the deployment, and the *NGINX Ingress controller* sets the routes for the external traffic management inside the cluster. If persistent storage is required, the definition of the pod with that requirement includes a persistent volume claim that is then processed by the *Heketi* server configured in the master, which communicates with the *GlusterFS* server from the storage node to dynamically allocate the volume. This deployment process is highly flexible and scalable, and allows the experimenter to update any component by simply editing the configuration of any resource defined and applying the new configuration to the already working environment.

For this proof-of-concept, following the same approach as in the first testbed, underlying traffic is generated to ensure that the results from the experiment are relevant, considering the stressful network environment characterizing a real PPDR operation. Two iPerf agents, one deployed behind the core network as a application function and the other in a 5G UE, generate the traffic to overload the 5G network end-to-end.

Regarding the 5G radio setup, the Malaga testbed operates a 5G private network based on a 5G Stand Alone (SA) deployment with four 5G NR Time Division Multiplexing (TDD) cells in band n78 at 3.5GHz and with an associated channel bandwidth of 50 MHz per cell. With a 2x2 MIMO with 256QAM modulation, the scheduling configuration can reach 342 Mbps per carrier. In addition, the gNBs have a proactive scheduling feature to generate additional uplinks and decrease the latency of the scheduling request procedure. As a result, the average measured latency of the scenario is in the order of 10-12 ms [113].

UNIVERSIDAD DE MÁLAGA

Figure 50 lists the pods what were deployed in this proof-of-concept to create a *Mobitrust* instance on top of the cloud-native infrastructure.

```
NAME                                  READY   STATUS    RESTARTS   AGE
message-broker-fc65485b7-x42nt        1/1     Running   0          110s
mt-device-65b6946955-6bf2d            1/1     Running   0          33s
mt-gateway-6c77f466c5-g478x           1/1     Running   0          2m32s
mt-kpi-manager-7ff688d8f7-chcg4       1/1     Running   0          31s
mt-monitor-647df94599-w2qlp           1/1     Running   0          41s
mt-orchestrator-6566b465df-vhdbp      1/1     Running   0          46s
mt-portal-5dc7dbdbdf-xkncv            1/1     Running   0          2m8s
postgresql-85dd7c678c-8dsxl           1/1     Running   0          3m11s
tick-influxdb-85f94ff884-5lj9w        1/1     Running   0          86s
tick-kapacitor-5b84494d45-vtqjf       1/1     Running   0          34s
tick-telegraf-5788f79b8b-zhd9q        1/1     Running   0          36s
webrtc-67b444cdd6-ppntq               1/1     Running   0          115s
```

Figure 50: Mobitrust K8s pods deployed

Thus, the proof-of-concept includes the 5G radio set up, the *Mobitrust* application deployed on the K8s distributed platform, and a real user equipped with a bodykit that includes a camera, a 5G modem, and a set of sensors. When the bodykit starts, it authenticates into the platform. On the other side, the CCC operator connects to the Portal, authenticates, and opens the dashboard; which is handled by the Orchestrator and the Message broker. The streaming multimedia recorded with the bodykit is then transmitted through the WebRTC server, together with the sensor data, and summarized into the dashboard for the CCC operator to visualize.

To validate the foreseen advantages of using cloud-native technology to deploy 5G PPDR applications, the following KPIs are retrieved, and their results are depicted in the figures further down:

- Platform deployment time: Time (in seconds) elapsed since the K8s deployment starts until every pod is in ready state.
- Device authentication time: Time (in milliseconds) elapsed since the bodykit (end-user device) is turned on until the reception of the acknowledgment.
- Sensor data latency: Time (in milliseconds) elapsed since the end-user sends sensor data in bulk format until it is received by the CCC operator processed by the server.
- Incident notification time: Time (in milliseconds) elapsed since events are identified until the CCC operator and the mobile CCC operator are notified.
- End-to-end Standard Definition Multimedia Latency: Time (in milliseconds) elapsed since the end-user device starts sending multimedia until it is displayed at the CCC.

UNIVERSIDAD DE MÁLAGA

- End-to-end High Definition Multimedia Latency: Time (in milliseconds) elapsed since the CCC operator requests multimedia until the contents are displayed in the dashboard. In this case, the algorithms for encoding and compressing the streams are more efficient than in the Standard Definition case, and the delivery to the CCC is automated and tailored to the available resources.

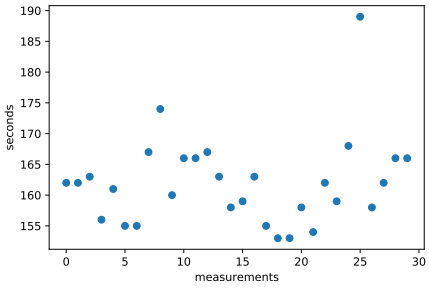

Figure 51: Deployment Time

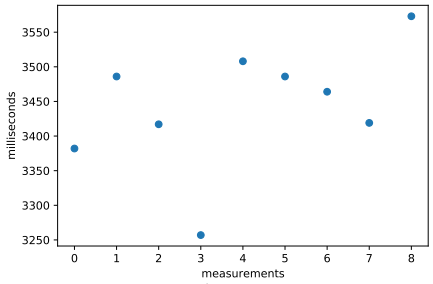

Figure 52: Authentication Time

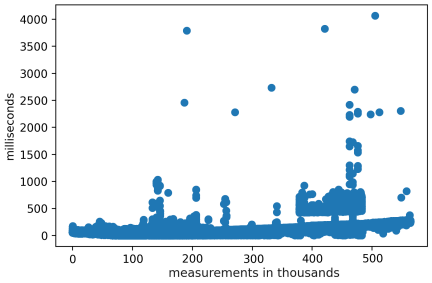

Figure 53: Sensor Data Latency

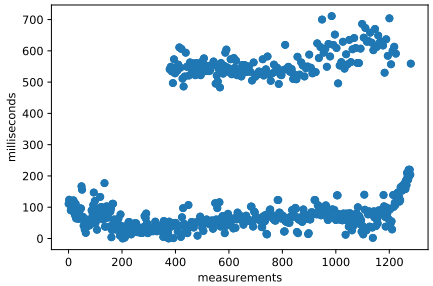

Figure 54: Incident Notification Time

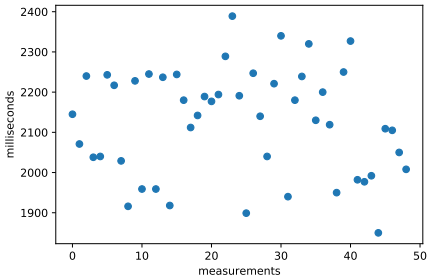

Figure 55: SD Multimedia Latency

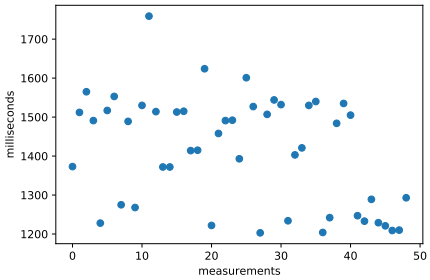

Figure 56: HD Multimedia Latency

These results are in line with the expectations extracted from the PPDR applications requirements defined in terms of latency, reliability, and capacity [11], the operational needs for emergency groups [6], and the detailed analysis on the 5G use in the context of public safety [62]. Specifically:

- Platform deployment time (Figure 51): It is expected to be less than 5 minutes at the edge and 10 minutes for on-premise platforms. In this experiment, multiple iterations were performed and the average deployment time is under 3 minutes.
- Device authentication time (Figure 52): It is expected to be less than 1 second, whereas in this experiment the results are be-

UNIVERSIDAD DE MÁLAGA

tween 3 and 3,5 seconds. This can be explained due to the lack of coverage in some areas included in the route followed by the end-user carrying the bodykit, combined with the latency increment in mobility experiments with handover scenarios, and will be further pursued in future research.

- Sensor data latency (Figure 53) It is expected to be less than 250 milliseconds, which is achieved during the majority of the experiment considering the number of measurements is in the order of hundreds of thousands.
- Incident notification time (Figure 54): It is expected to be less than 1 second, which is achieved in both communications, to the CCC operator at the non-emergency zone, and to the mobile CCC operator at the emergency zone. However, the graph shows a difference between the notification time to the two operators, due to the variation in their location and coverage.
- End-to-end Standard Definition Multimedia Latency (Figure 55): It is expected to be less than 500 milliseconds if the streaming is live, or 1 second if the transmission is not real-time, so the target for this KPI is not met.
- End-to-end High Definition Multimedia Latency (Figure 56): It is expected to be less than 600 milliseconds if the streaming is live, or 1 second if the transmission is not real-time, which is our case. Compared to the Standard Definition case, the algorithms for encoding and compressing the High Definition streams are more efficient, so the latency is reduced. However, not enough to meet the target KPI, although results are promising.

The results as a whole still present room for improvement in the application part to meet the highly demanding requirements of the PPDR vertical. Nevertheless, with regard to the deployment part and the use of cloud-native technologies, results are promising and show a clear improvement in the expected KPI.

UNIVERSIDAD DE MÁLAGA

Part IV

APPLICATION

This part contains the research on microservices and network applications to increase the granularity of the services and their distribution towards easing their management and orchestration. The published work supporting this part includes *"A network application approach towards 5G and beyond critical communications use cases,"* published in *Frontiers in Communications and Networks* in 2024 [23].

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

6

INCREASING DISTRIBUTION WITH MICROSERVICES

The use of virtualization solutions and software components in the definition of 5G and beyond technologies, motivated by the services evolution to adapt to new infrastructures and systems, has turned these networks into agile networks that are assembled and configured for specific use cases by means of automation and programming. Likewise, network slicing allows the same infrastructure to host multiple networks, each with different requirements and configuration depending on their vertical, which also implies the need for on-demand reconfiguration procedures that are cost-efficient for the network operator. This is possible by adopting the NFV paradigm and decomposing the VNFs into microservices.

In this context, cloud-native solutions provide the abstraction from the infrastructure required to easily adapt to new technologies while maintaining compatibility with legacy solutions, and services containerization enables a lightweight virtualization for microservices creation and chaining, in order to unlock higher performance and easily manage these distributed services. Thus, distributing the services by increasing their granularity rises as an alternative to distributed orchestration systems, which could be also used in combination with them.

This chapter presents the microservices approach to show how the NetApp ecosystem and delivery model unlocks the benefits of 5G technology for the verticals operators and end-users. In the following sections, the NetApp concept is further explained, and applied to the 5G experimentation platform proposed in Section 5.3 to integrate the cloud-native solutions into the architecture proposal. Finally, the resulting NetApp-oriented experimentation platform is evaluated with a series of experiments targeting the PPDR vertical. With this chapter, the last objective of the thesis, defined in Section 1.2 as the creation of the experimentation platform and its demonstration by use cases application, is achieved.

6.1 NETWORK APPLICATIONS

The concept of Network Application, or NetApp, appeared for the first time in the European Commission Horizon 2020 Framework Programme Call for Information and Communication Technologies under Topic ID ICT-41-2020 "5G PPP – 5G innovations for verticals with third-party services", where it was defined as a chain of VNFs

UNIVERSIDAD DE MÁLAGA

linked to address specific requirements of the novel 5G network verticals [63].

At their early stage, NetApps were understood as merely NFs meeting the ETSI NFV specification. Although, it is possible to combine these NFs to create complex end-to-end network services in which traffic flows through the NFs as if they were enchained. These chains provide novel network services for vertical systems, both standalone or by building integrated software, besides simplifying large-scale deployments by exposing the network capabilities via APIs, so access to the network resources is delivered through the northbound interfaces.

Thus, the NetApps offer a middleware or abstraction layer to the vertical applications to support them in consuming the resources from the different elements that compose the 3GPP 5G architecture. Officially, NetApps are defined in [146] as *software facilitating an interaction layer between vertical applications and the network control plane*, which means that by using a NetApp, the network slice and its characteristics are customized to meet the requirements of a dedicated task from a vertical application, to be able to host the slice's end-users and their needs. To simplify the vertical access to the network architecture, the northbound APIs are exposed in a standardized and trusted manner also through those NetApps, so that vertical services can take advance of that exposure and the novel network capabilities, such as network data analytics collection [127], mobility [20], and 5G QoS management mechanisms [47, 116].

Regarding the NetApp ecosystem and delivery model, there are different approaches to integrate the NetApps in the already existing 5G infrastructures. The main classification in the NetApp implementation, differentiates them based on the utility for which they have been designed, this is, as vertical-specific, if the NetApp software is designed to meet requirements that are characteristic to a vertical, or vertical-agnostic, if those requirements can apply to different verticals and the NetApp composes a generic solution.

Nevertheless, the majority of the 5GPPP initiatives that research on the different NetApp implementations, follow a Continuous Integration/Continuous Delivery (CI/CD) Development and Operations (DevOps) methodology to simplify the NetApp implementation and design, while evolving towards a microservices-oriented architectural design with container-based services and applications.

Bearing this in mind, there are two possibilities for a NetApp to provide service to vertical applications:

- As a vertical-agnostic NetApp, which is an integrated part within the vertical application offering 5G and beyond QoS management solutions, KPI analysis, or security services that are common to different verticals; or
- As a vertical-specific NetApp, which exposes the APIs to provide control capabilities over the network infrastructure, aiming at

UNIVERSIDAD DE MÁLAGA

providing support to a concrete vertical, and consuming the vertical-agnostic NetApps to also offer their additional functionality to the vertical services.

6.2 NETAPP-ORIENTED 5G EXPERIMENTATION PLATFORM

One particularity of the 5G networks is enabling and promoting interaction between the network control plane and third party tenants or external elements. As the main NetApp goal is customizing the network to fit the specific requirements of third party tenants and vertical applications, while interoperating with the underlying network, NetApps are considered a key enabler for this dynamic interaction and control plane mechanisms standardization. This makes mobile networks more accessible to service developers, and grants them dedicated network resource allocation permissions.

According to the standard specification in [7], NetApps provide different network features via dedicated standardized 3GPP interfaces, and operate by means of AF on the 5GC control plane. This results in that an AF can interact with the targeted 5GC NF exposed by the NEF, and even without NEF mediation if that AF is trusted by the network, which makes for vertical services possible to identify themselves and then request specific conditions from the network, creating a value chain of NFs.

An "hybrid" interaction approach, as defined in [146], allows not only the adjustment of the network to meet a vertical set of requirements, but also for the vertical to leverage the entities responsible for operating advance services and exploiting 5G capabilities. The components in this interaction approach are:

- The Vertical System (VS), which represents the end-to-end vertical solution deployed over a 5G network, and includes different components, client applications, and servers located either in the vertical service provider domain or as adjacent services instantiated and managed by the 5G (infrastructure operator.
- The Vertical Application (VA), which delivers vertical-specific functionalities and is usually combined with other VAs to compose the VS.
- The NetApp, which abstracts the 5G network components to expose them as API calls for their consumption.

Hence, NetApps are considered integration elements that combine their service with those of the VAs. In the practice, it is common that NetApps take parts of the VS to be reused in a different VS from the same solution provider.

Following the design and implementation presented in Section 5.3, the 5G-EPICENTRE project provides a platform to deploy NetApp-based developments oriented to the 5G PPDR vertical. The high-level

UNIVERSIDAD DE MÁLAGA

experimentation infrastructure implemented includes four different and independent testbed facilities with a K8s-based management architecture to orchestrate the containerized applications, and a federation layer on top of them with a Karmada control plane architecture [159]. As depicted in Figure 57, to provide access to those underlying testbed resources there is a centralized, cloud-native experimentation platform that includes a backend layer to orchestrate the vertical application deployment under the desired test conditions, and a frontend layer to expose the platform to the researchers.

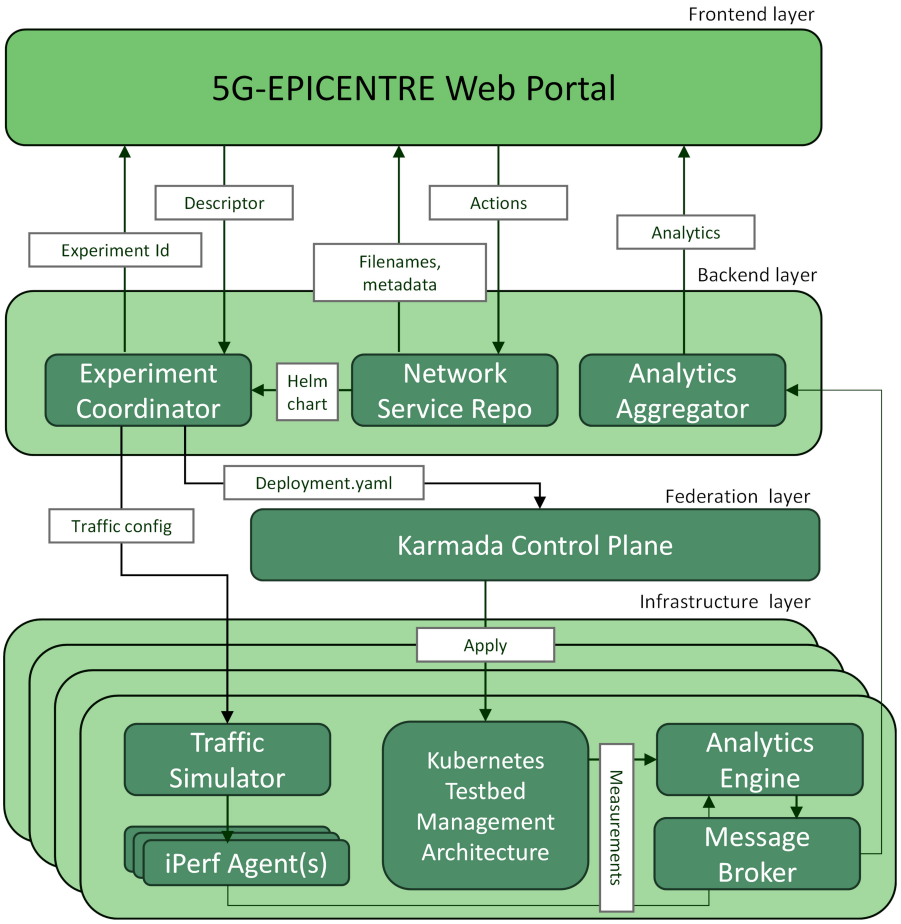

Figure 57: 5G-EPICENTRE high-level layer structure [23]

Most of this platform's prototypes evolved from a pure OpenStack ecosystem following the ETSI MANO reference to include the support for K8s acting as the VNFM. This NetApp-based testbed allows vertical application developers to experiment and demonstrate the enhanced capabilities of the 5G network and its benefits in the context of PPDR agencies and end-users, guaranteeing:

- Privileged QoS to PPDR vertical services.
- Prioritization of certain emergency data flows by integrating management capabilities of 5G QoS Identifier (5QI).

UNIVERSIDAD DE MÁLAGA

- Transmission of the data streams from PPDR communications over a 5G network to meet its time frame and quality requirements.

There are three possible integration options for the NetApps into the 5G network system:

- As-a-Service: The NetApp exposes the APIs for the VA to consume. The VA is fully deployed in the vertical service provider domain.
- Hybrid: Part of the VA, such as the part deployed on the edge, is delegated to the infrastructure provider and resides in the platform's set of NetApps, whereas the rest of the VA is located in the vertical service provider domain and interacts with the the NetApp through its exposed APIs.
- Coupled/Delegated: The whole VA is delegated to the infrastructure provider, and the NetApp is embedded into the VA and managed by the operator.

As stated previously, the 5G-EPICENTRE platform leverages on container-based virtualization technologies to provide robust and secure service deployments, and highly available services in the context of the PPDR vertical to support this NetApp approach.

6.3 TOWARDS A HIGHLY GRANULAR AND DISTRIBUTED ARCHITECTURE

Adopting a cloud-native, container-based architecture that implements the NetApp delivery model denotes the evolution of existing 5G facilities towards microservices-oriented solutions. In this context, the 5G-EPICENTRE platform addresses resource fragmentation through a federation of multiple K8s-based experimentation facilities via multi-cluster container deployment, enabled by Karmada K8s management and federation tools, to ensure that multiple NetApps are deployed and run concurrently.

This cloud-native architecture provides the means to deploy, manage, test, and operate not only the NetApps but also their underlying infrastructure. Hence, the implementation of the platform architecture presented in Section 5.3, provides high availability and redundancy for mission critical deployments by resting on a multi-master, multi-node, K8s-based infrastructure in which the nodes are deployed on a dedicated environment that integrates different physical servers.

As a result, this 5G NetApp-oriented platform is able to integrate both distribution of the services and the orchestration, aiming at a highly granular and distributed architecture, and thus improving the first proposal of the architecture since this service granularity eases

UNIVERSIDAD DE MÁLAGA

the adaptation of the network to constantly changing environments and requirements.

In addition, taking into account that a NetApp is a chain of functions, both physical and virtual and, thus, some microservices might be out of the cloud-native infrastructure, the implementation includes an ingress controller to expose the routes from outside the cluster to the internal services and provide the external microservices access to the internal ones. Following the reference implementation in [113], when an experiment is launched, a Helm chart is built, creating a chain with every microservice and network component required, and indicating the testbed facility owning each resource to deploy them properly.

When interacting with the K8s cluster, the NetApp owner establishes their context for the deployment to ensure isolation, the service account provided for the authentication, and the K8s API to balance the NetApp queries among the control plane nodes and guarantee high-availability. Then, the NetApp owner interact with the cluster via YAML files, i.e., if the NetApp includes a Pod (which is the smallest deployable unit in a K8s environment) requiring persistent storage, that Pod's YAML should define a claim that the platform will handle through the Heketi server, to automatically create and bound a GlusterFS volume to that Pod. Moreover, if the NetApp interacts with external components, the experimenter must create a YAML describing a LoadBalancer service, which assigns an address from the MetalLB IP address pool that will expose that Pod to the network, making it reachable to everyone in and out of the cluster [23].

However, to unlock the benefits of a distributed architecture is not enough with distributing the services into microservices with higher granularity. A multi-cloud environment combined with a cross-platform MANO solution arises as a combination of both orchestration and service distribution, and enables the definition and execution of microservices composed NetApps across different PoPs.

Through the K8s API, the cross-platform MANO can interface with the clusters deployed in each individual underlying 5G platform. In the context of the 5G-EPICENTRE platform, the testbed federation solution is deployed via the K8s-based NFVO that manages and provides access to service resources in each federated experimentation facility while offering multi-cluster K8s orchestration across different domains, abstracting the multi-cluster testbed federation as a single testbed for the NetApp owner perception. This proposed NFVO is based on the Karmada management system, which deploys cloud-native applications spanned across different clusters, and allows the K8s resource pooling across infrastructures, adding another layer of abstraction understood as the federation layer.

To this end, Karmada provides a unified information model that wraps the different aspects of each platform's API, so the platform owners keep control of their 5G resources while there is a general

UNIVERSIDAD DE MÁLAGA

adaptation to each cluster particularities. It also provides a single layer to manage all the platform components from the different clusters as if it was a single cluster deployment, combines all the resources available to offer a wider experimentation framework to the NetApp provider, and allows modifying, activating, and deactivating different features with a plug-in based architecture in which each task from the deployment of a service is performed by a plug-in. Figure 58 depicts this high-level architecture of the cross-platform federation, which enables the deployment of NetApps across different platforms from a single interface.

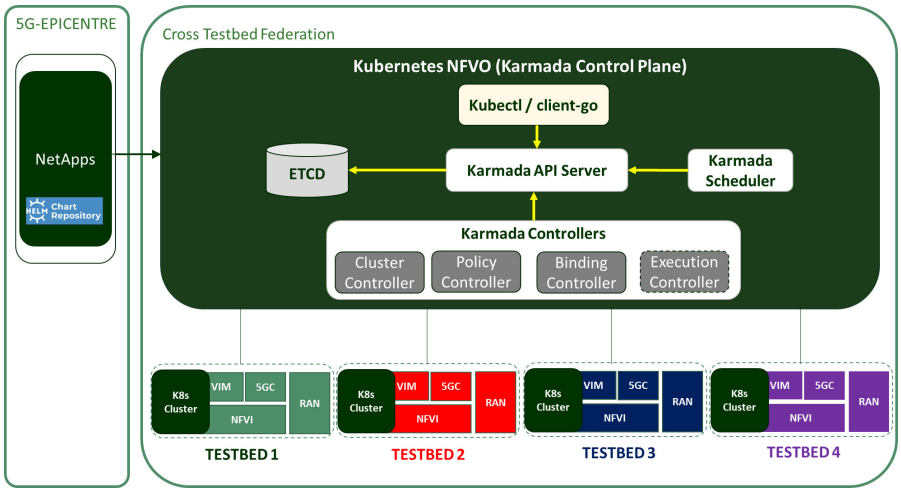

Figure 58: Cross-platform federation high-level architecture [23]

6.4 LEVELS OF INTERACTION FOR 5G VERTICAL SYSTEMS

This section presents a series of experiments to prove the actual enhancements of applying a NetApp approach to the 5G ecosystem and its vertical services under different levels of interaction between vertical and platform application components. Specifically, three different 3GPP standard-compliant Mission Critical Everything (MCX) applications from the PPDR vertical are tested, focused on real-time group communications, situational awareness platforms, and augmented reality solutions respectively, to demonstrate the feasibility of integrating mission critical communications into 5G networks, and thus overcoming the current narrowband limitations.

6.4.1 *Tight coupling: MCX Solution*

The *Nemergent Mission Critical Services* from *Nemergent Solutions*[1] provides a solution for Mission Critical Push-to-Talk (MCPTT), Mission Critical Data (MCData), and Mission Critical Video (MCVideo) follow-

1 www.nemergent-solutions.com

UNIVERSIDAD DE MÁLAGA

ing the 3GPP standard for MCX solutions in mobile broadband environments, and evolving their architecture towards compliance to the NetApp delivery model.

For this demonstration, the NetApp architecture defined includes both physical and virtual services chained into a VS to create a microservices-based end-to-end MCX solution. Hence, this VS, as depicted in Figure 59, includes a number of containerized services and network elements compliant to what the 3GPP defines as MCX components: one Application Server CNF that includes the Participating and Controlling application servers, and a set of Management server CNFs, such as the Configuration Management System (CMS) for configuration, the Group Management System (GMS) for groups management, the Key Management System (KMS) for the keys, and the Identity Management System (IdMS) for the identities. In addition, the NetApp includes a database and a monitoring module containerized to provide additional management services.

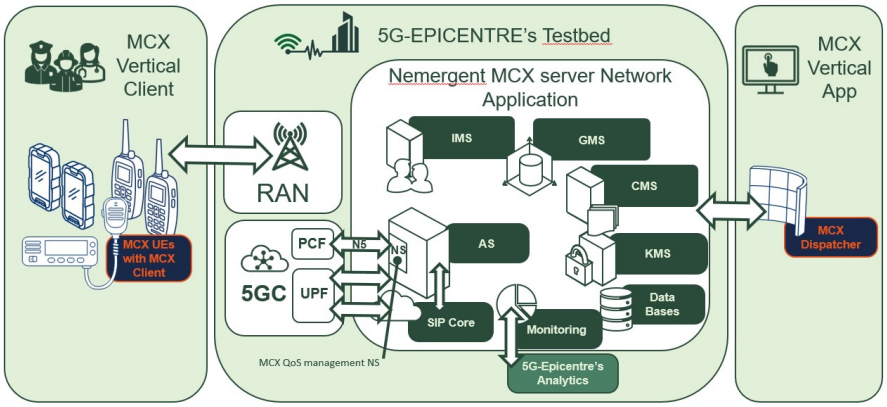

Figure 59: Overview of the *Nemergent* MCX services chaining [23]

In this architecture, the tight coupling of NetApp components within the VS itself and with the testbed infrastructure operator integration elements is evident. This allows *Nemergent* to deploy their solution and, via requests, to impact the control plane of the network directly from its own VS [23]. This NetApp integrates into the infrastructure at a different degree, depending on the infrastructure owner needs, i.e., as an over-the-top solution, or as a complex hybrid system interacting with the network infrastructure itself.

Moreover, the NetApp includes in its microservices' chain a MCX QoS management NS to exploit the N5 interface for mission critical communications traffic priorization over the 5G network, which allows the VS to control the priorities and QoS by interacting with the 5GC control plane functions.

Regarding the experimentation process, the first part of this proof-of-concept includes the analysis on the network conditions to guarantee that network slicing and resources sharing technologies adopted

UNIVERSIDAD DE MÁLAGA

by the novel 5G infrastructures do not affect the mission critical communications performance.

To this end, multiple experiments were performed with the MCX QoS management NS and different slice parametrizations to evaluate the results when changing the underlying traffic model.

Thus, the MCX service performance was tested under demanding traffic conditions, as shown in Figure 60, with (depicted on the right) and without (on the left) network slicing policies to evaluate the End-to-End MCPTT Access Time (KPI 2), standardized and defined in [12]. The results depicted in Figure 60 show that if the traffic on the network increases significantly, the degradation on the KPI 2 average value is higher for the case without slicing policies, where a few samples in the worst-case scenario almost overcome 1000 ms, which is the limit value defined in the standard for this KPI 2, while for the same traffic applying slicing policies the MCX service is not affected.

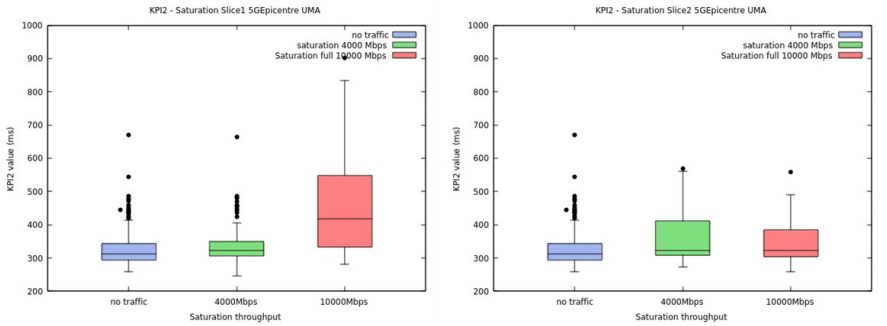

Figure 60: Benchmarking slicing impact through KPI 2 [23]

In conclusion, a slice parametrized specifically for mission critical communications in a scenario applying network slicing is not affected by high traffic on the underlying network, thus ensuring the end-users requirements are met in the most demanding conditions.

The second part of this proof-of-concept addresses the benefits of combining the NetApp approach with 5G technology in terms of service virtualization and deployment versatility, since the time required for service deployment, redeployment, self-recovery, or auto-scaling is reduced to a minimum towards meeting the most demanding needs of the PPDR vertical applications.

In this context, development and experimentation have been conducted towards generating a fully CNF-based solution, with light-weight virtualization entities to improve deployment time, as depicted in Figure 61. A PPDR related experimental scenario has also been built to demystify inter-MCX communication, where a new MCX server NetApp is deployed from scratch in approximately one minute, in contrast with a VM-based solution, where a similar deployment could take hours. Furthermore, this new deployment takes place in a multi-cluster K8s environment, where another NetApp instance is already deployed.

UNIVERSIDAD DE MÁLAGA

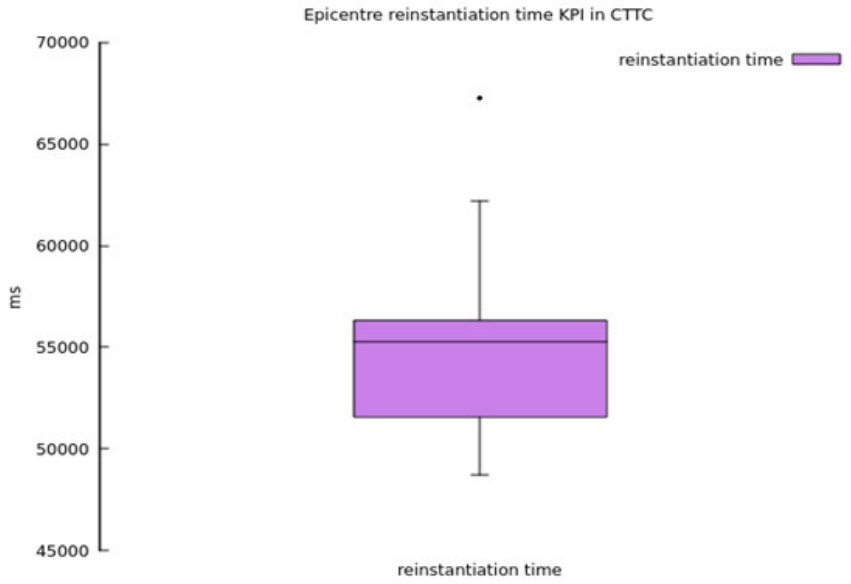

Figure 61: Re-instantiation time at a different cluster [23]

This experiment illustrates how the adoption of agile deployment solutions significantly improves the instantiation process in comparison with other deployment models, and how two independent MCX VS located in two separated clusters are able to communicate with each other as soon as the NetApp is deployed, which simplifies MCX services instantiations at the edge.

6.4.2 *Loose coupling: Situational awareness platform*

Another key enabler for the MCX services in the context of 5G networks are the IoT systems. The NetApp delivery model in this case can be demonstrated by means of the same situational awareness platform presented in Section 5.3.3, the *Mobitrust* platform from *OneSource*[2] for CCC PPDR operations.

In this use case, the platform gathers data automatically from multiple sensors integrated onto IoT devices integrated in bodykits worn by first responders on site to obtain full awareness from the field during disaster response situations. The data included is obtained from bio-sensors, positioning devices (geographical and indoor), internal communication systems, vehicles, drones, shared devices, and real-time real-time text, audio and video transmissions.

The VS architecture, as depicted in Figure 62, includes different NetApp components that are delegated to the infrastructure, which proves a looser coupling between NetApp, the VA components, and the testbed integration elements.

2 www.onesource.pt

UNIVERSIDAD DE MÁLAGA

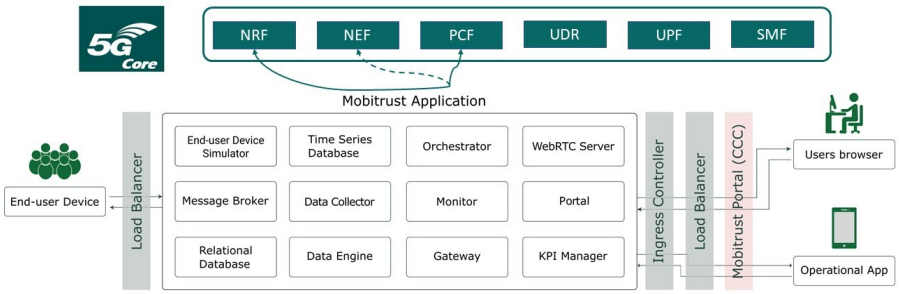

Figure 62: Overview of the *OneSource* platform services chaining [23]

In this case, the Portal is the VA component located in the vertical service provider domain, interfacing with the end-users via web interface or mobile application, and with the NetApp via service API calls. On the other hand, the Orchestrator component included in the NetApp chain interfaces with the 5GC acting as an AF to request certain QoS from the 5G network for a specific communication, aiming at improving the quality of the streams retrieved by the IoT devices for their monitoring.

As a result, not only the IoT systems are demonstrated as key enablers, but also the edge computing paradigm since the microservices composing the VS can be located as close to PPDR end-users as possible, to achieve a higher awareness of the field operations, improve the streams quality, and reduce sensors latency.

The benefits of combining the NetApp delivery model with the 5G technologies were proven in scenarios with dedicated slicing mechanisms implemented. The target KPIs analyzed to attest this conclusion are the network Round-Trip-Time (RTT) and the messages delay.

For the network RTT, as depicted in the upper side of Figure 63, results were on average 29.50 ms for a congested slice (on the left part), and 24.87 ms for a dedicated slice (on the right). The message delay obtained, shown in the lower part of the figure, reached on average 28.25 ms for a congested slice and 24.20 ms for a dedicated slice.

These results are in line with the target KPIs extracted from [11], [6], and [62]. Specifically:

- RTT: It is expected to be less than 20-40 ms for PPDR applications in 5G networks with dedicated slices, and less than 50 ms in congested networks, reaching up to 100 ms if network slicing is not present. In this case, Figure 63 depicts an average RTT of less than 30 ms with network slicing and regardless of the congestion.

- Message delay: It is also expected to be less than 20-40 ms if the slice is dedicated, less than 50 ms in congested networks, and in this case less than 100-300 ms without network slicing. In

UNIVERSIDAD DE MÁLAGA

the experiments, results achieved again a message delay of less than 30 ms for both cases in a scenario with network slicing, as shown in the lower part of Figure 63.

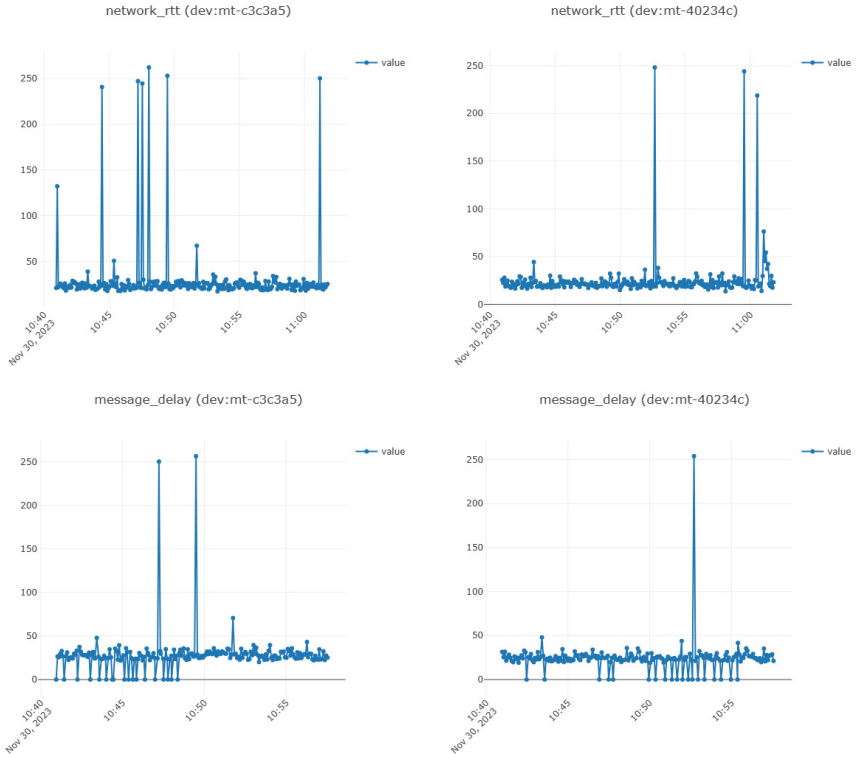

Figure 63: Network RTT and Message-Delay KPIs (Slicing scenario) [23]

6.4.3 *Platform-agnostic: AR Solution*

Between the technologies most affected by the wide deployment of 5G networks are the mobile Augmented Reality (AR) applications. Bearing this in mind, *ORamaVR*[3] offers a real-time remote AR 3D rendering and streaming VS for the PPDR vertical to provide first responders a layered view of deformable 3D objects such as bones or organs overlaid on top of an injured body at the field during disaster response situations by means of a light-processing, portable, and battery-efficient AR Head-Mounted Display (HMD). The goal of this AR solution is to support first responders in emergency situations by providing real-time, step-by-step instructions to perform critical surgical operations.

To this end, the VS integrates two microservices chains, as presented in Figure 64. The first chain, located on the AR HMD, includes microservices to record and transmit user events, to decode

3 www.oramavr.com

UNIVERSIDAD DE MÁLAGA

and project the rendered streams, to receive video streams from the VS' streaming services, and to activate the edge components.

The second chain, deployed at the edge, includes on the other end microservices to update the AR scene, to render the scene triggered by the HMD inputs, to encode, compress and stream the video generated, and for physics calculations on the deformable objects. Both chains are part of *Orama*'s MAGES Software Development Kit [188]. Figure 64 presents the VAs interconnected according to the NetApp approach, with both chains highlighted in a red dashed box each.

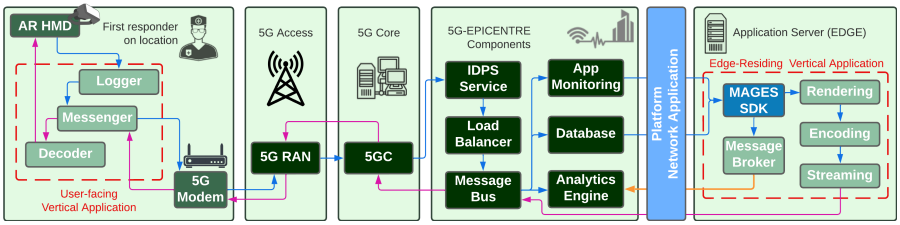

Figure 64: Overview of the *Orama* AR services chaining [23]

This AR solution deployment validates the capacity of the proposed NetApp delivery model to meet highly demanding VA deployment due to its requirements in terms of bandwidth and strict latency. Bearing in mind that many applications were designed for legacy networks and without considering a transformation towards cloud-native environments, the capacity to connect any VS with 5G functionalities through the service APIs of a NetApp is considerably valuable. This capacity refers not only to the abstraction from the underlying network but also to the ability to translate the VS' network requirements to 5G network resource allocation requests.

This experiment aims at demonstrating the benefits of the NetApp approach for real-time AR-based applications with challenging needs, such as the strict synchronization required between the AR rendered streams and the real space, by means of maintaining a high fidelity and quality video stream transmitted continuously and with minimal latency to the HMD.

In this case, the high processing load in terms of CPU and Graphical Processing Unit (GPU) would drain the HMD battery rapidly if the application followed a monolithic approach. By following a NetApp approach with multiple microservices handling the different utilities of the HMD, the VS is able to offload the AR pipeline's heavy-duty processes to the edge residing VAs, thanks to the low latency and high bandwidth provided by the 5G network underlying.

The experiment results demonstrate that the performance optimization is achieved by following the VS implementation, that reaches a high bitrate of approximately 500 Mbps in network-stressed situations, with a mean throughput of 408 Mbps and a low average latency of 10.68 ms due to the 5G network underlying, surpassing the known

UNIVERSIDAD DE MÁLAGA

limitations of 4G networks. These results are shown in Figure 65, where the possibility of streaming a high amount of data with minimal latency to the HMD through the 5G network is proven. Due to the HMD's inability to process the large amount of data received and a non-existent QoE guaranteed, the average packet loss is however 75%, and the rendered frames per second on the HMD drop significantly, in contrast to the frames per second rendered on the VA located at the edge.

To mitigate the packet loss and improve the AR experience, the same experiment was performed following a QoE-driven approach, reducing the bitrate to 50-100 Mbps. In the same network-stressed conditions, the throughput obtained was continuous and stable at an average of 68.56 Mbps with a mean latency of 10.63 ms, as depicted in Figure 66, with a packet loss of 60% and similar results for the frames per second rendered on the VA located at the edge and the ones processed at the HMD. In spite of the packet loss, end-users reported a remarkable QoE with minimal stuttering and lag in the video stream and a high framerate, resulting in an optimized user immersion.

With regards to the previously mentioned HMD battery life, the VS implementation reached a 30% extension, which enhances the HMD on-site usability. However, advances on the headsets technology are expected (and required) to overcome the limitations observed on the QoE for bitrates greater than 300Mbps, to improve the quality and fidelity of the AR experience.

UNIVERSIDAD DE MÁLAGA

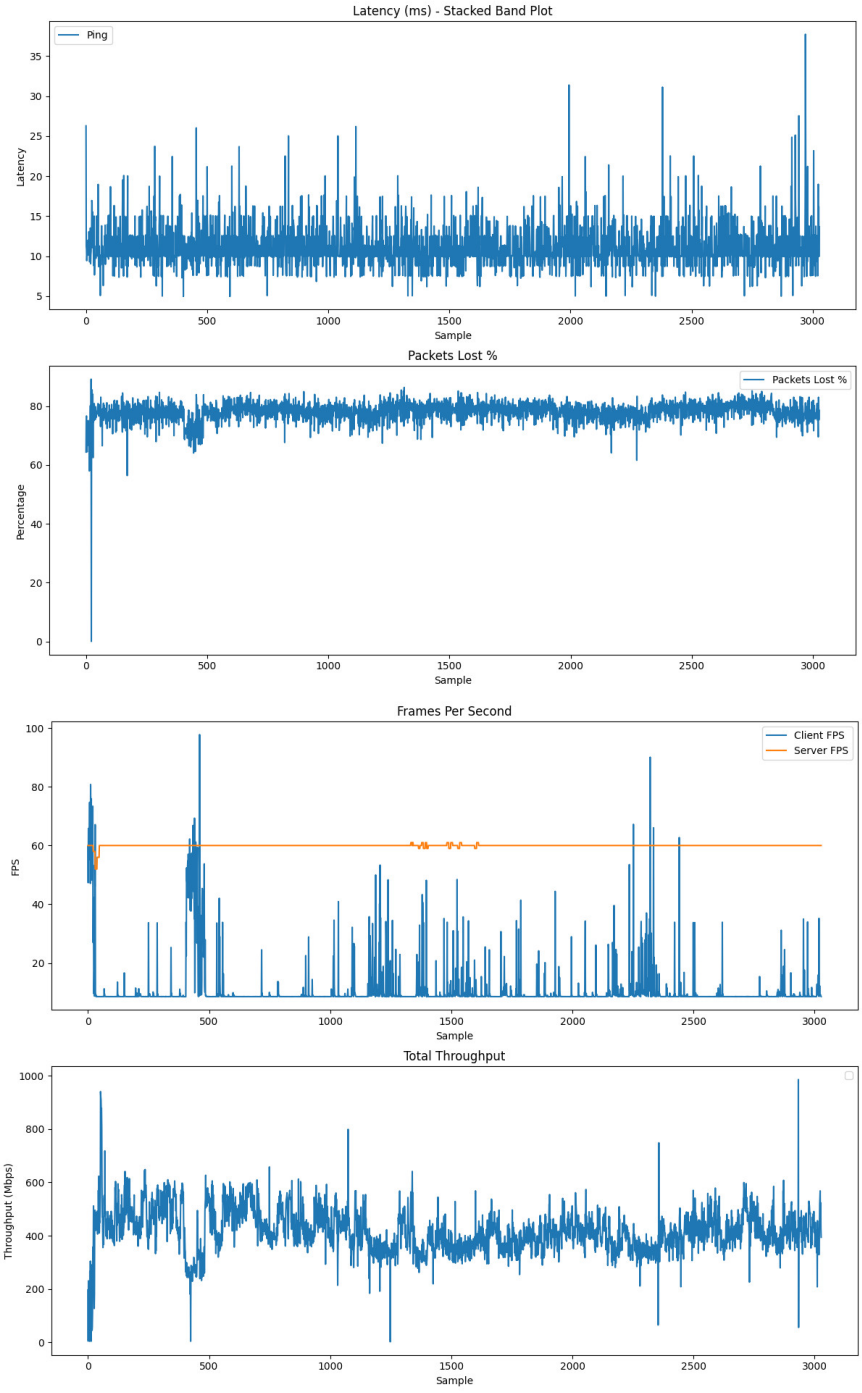

Figure 65: KPIs obtained for a bitrate of 500 Mbps [23]

UNIVERSIDAD DE MÁLAGA

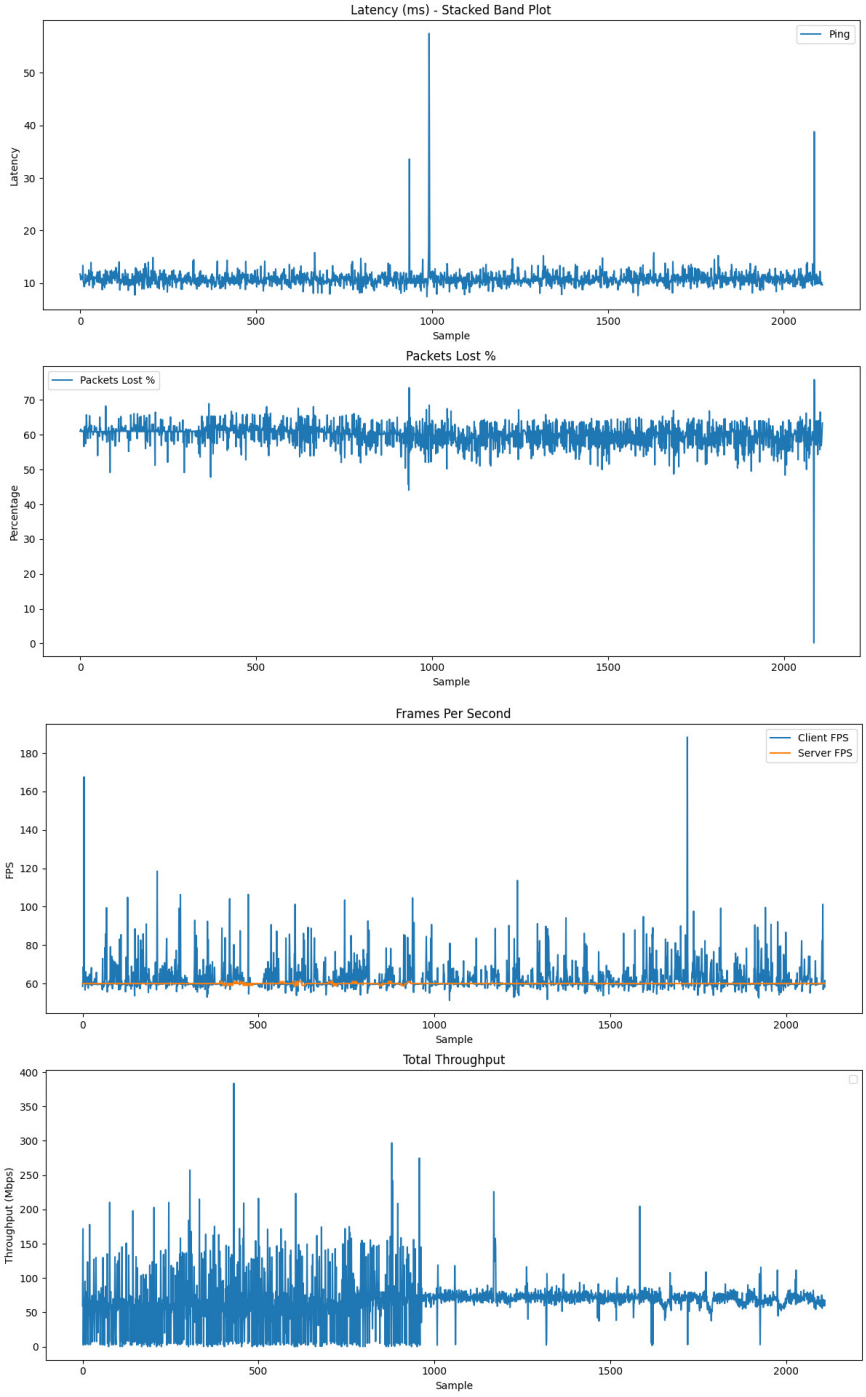

Figure 66: KPIs obtained for a bitrate of 50-100 Mbps [23]

UNIVERSIDAD DE MÁLAGA

Part V

FINAL REMARKS

This part summarizes the thesis, outlines directions for future research, and includes the list of publications and projects associated to this thesis.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

7

CONCLUSIONS AND FUTURE WORK

7.1 CONCLUSIONS

This thesis presents the creation of a research environment to deploy 5G slices with resources distributed across multiple locations and managed by an equally distributed orchestration system. The motivation behind this vision is the existing limitation in the field of cellular network research due to the demanding requirements to perform realistic mobile experiments.

Due to the integration of the cloud computing paradigm and different virtualization technologies to novel cellular networks' architectures, the challenges of virtualization are inherited by telecommunications networks. Specifically, challenges related to efficiently orchestrating the resources to manage the creation of network slices adapted to the novel verticals' requirements, benefiting from virtualization's added value to the network.

In this context, a common feature observed in the existing major experimentation platforms is the centralization of resource orchestration, which entails several challenges addressed by a distributed environment. Through the study of the literature related to the orchestration problem, the deficiencies in centralized orchestration systems have been identified, together with the challenges of distributing the orchestration, answering the Research Question 4 (RQ4): *Which are the main challenges of orchestration systems and what are the benefits and particular challenges of distributing the orchestration?*

This thesis provides a solution to offer temporary networks in the context of cellular networks research by designing an architecture based on the implementation of distributed and connected Points of Presence that include a heterogeneous set of technologies. The architecture hereby proposed represents the state of the art in the deployment of mobile networks, being the experimental customized networks provided to the researchers equivalent to the network slices tailored to a determined vertical in a 5G environment. Moreover, it represents a further step from the simple commercial exploitation of radio resources and computational power, putting them at the service of the researchers that are unable to access the required infrastructure otherwise. This solution answers the Research Question 1 (RQ1): *Which are the main enabling technologies required in an end-to-end experimentation platform to support realistic cellular tests?*

In order to evaluate the feasibility and maturity of the proposal, three approaches have been defined, implemented, and tested, based

UNIVERSIDAD DE MÁLAGA

on various virtualization techniques and with a different grade of granularity and distribution of the services and the resource orchestration.

i The first approach is the EuWireless testbed, which relies on the GTS platform as the base for physical and virtual resource management, that has already proved its usefulness in academic and research environments. By means of the GTS' virtualization model, the resources can be abstracted as virtual machines, extending the GTS platform to support experimental cellular networks. The resulting testbed minimizes the development and debug time required to test different approaches and configurations, showing an improvement compared to traditional experimentation platforms.

ii The second approach is an early stage of the 5G-EPICENTRE Malaga testbed, which is based on a containerized environment that, despite relying on a more monolithic approach with only one entity in charge of orchestration, improves the process of developing and deploying new services for a wide range of vertical solutions. The architectural design of this testbed is aligned with the reference and standards, presenting a solution for both cloud-native NFV and MANO that relies on Kubernetes for the orchestration and management of the resources and services. Kubernetes simplifies the management of large and dynamic systems while providing a framework to run distributed and scalable systems resiliently and with failover capacities. This approach aims at answering the Research Question 3 (RQ3): *Are traditional virtualization technologies enough to meet the requirements of novel cellular networks?*

iii The third approach is the definitive 5G-EPICENTRE testbed, which distributes the orchestration with a multi-master multi-node architecture to ensure high-availability and reliability while addressing high-demand operations oriented to meet the requirements of new 5G applications and technologies integration. With this approach, the Research Question 2 (RQ2) is answered: *Are centralized functions ready to support the creation of customized and temporary networks as the key enabler to provide service in novel cellular networks?*

The proposal's feasibility and maturity is evaluated through the experimentation on three different platforms that apply the architectural design from three different approaches, this is, distributed orchestration architecture with VNFs, monolithic orchestration architecture with CNFs, and distributed orchestration with CNFs. The main conclusion extracted resides in the nature of the distribution to be applied. Thus, despite the initial hypothesis of distributing the resources orchestration, further analysis and experimentation demon-

UNIVERSIDAD DE MÁLAGA

strates that distributing the services provided by increasing their granularity improves the infrastructure efficiency and eases the real-time deployments. Furthermore, the combination of both distributions, as presented in the last testbed, with fine-grained cloud-native functions distributed and managed though a distributed Kubernetes-based orchestration, has proven superior performance.

In this line, the last part of this thesis explores the benefits of creating a distributed cloud-native experimentation infrastructure to provide an environment suitable to adapt different 5G-enabled mobile applications to the Network Application model. The experimental evidence obtained attests to the improvements offered through both the Network Application model and the use of Kubernetes as key enabler in the orchestration of abstracted resources to create network slices in the context of the 5G PPDR vertical, in way of enhanced network capacity, better slicing mechanisms, and better handling during spike periods. This is projected to help PPDR service developers to deal with the high flow of data that is foreseen for future mobile communications.

7.2 FUTURE WORK

There are several open research topics that require further investigation to enhance this thesis repercussion and extend its scope.

Firstly, some of the identified orchestration challenges are still unresolved. The state-of-the-art of orchestration provided a sketch on how distributing the orchestration system addresses the challenges related to scalability, automation, and resiliency. However, further research on dynamic slicing, resource sharing and allocation, mobility management, wireless resources virtualization, isolation, and security, is required.

Secondly, the implementation of a large infrastructure like EuWireless or 5G-EPICENTRE at a regional, national, or global level would have a real impact on the manner in which researchers and small companies can validate their new technologies and services. Future work in this context includes the full implementation of relevant 5G components based on a generalized virtualization model to offer 5G virtual slices as testbeds to the general research communities. The increase of the flexibility of the slices provided to support more elaborated experiments that consider different scenarios is also a pending topic in the context of creating a valuable platform for researchers.

Thirdly, further research is required to analyse the impact of the Network Application model interaction with the 5G network, for QoS management and 5G slicing in stressed network conditions; the deployment versatility and cross-domain coordination and communication for PPDR agencies mobilized in the emergency areas; and edge instantiation of PPDR services.

UNIVERSIDAD DE MÁLAGA

Finally, the platforms hereby defined and deployed were tested focusing on the PPDR vertical. The remaining vertical sectors, which include automotive, manufacturing, media, energy, health, and smart cities, also define a set of KPIs and requirements that might be applied to these experimental platforms to provide a more complete environment for 5G research towards studying the consequences of the Network Application approach and evaluate whether its benefits also extend to the rest of the 5G verticals.

In December 2024, the Smart Networks and Services Joint Undertaking (SNS JU) presented their Work Programme 2025, which covers the strategic plan for Horizon Europe covering the 2025-2027 period. This Work Programme includes a stream that focuses on service provision enablers and PoCs to consolidate EU-wide experimental infrastructures focusing on 6th Generation telecommunications cloud. Thus, the main goal for this stream is creating a research and development 6G telecommunications cloud combining pan-European platforms with service provision enablers to test and experiment with candidate 6G technologies, which is in line with the objectives and results stated through this thesis, and presents an opportunity to continue working towards the evolution beyond 5G networks.

7.3 PUBLICATIONS AND PROJECTS

This section includes the publications, activities, and funding of this thesis.

7.3.1 *Journals and International Conferences*

- **J1.** A. Rios, B. Valera-Muros, P. Merino-Gomez, and J. Sobieski, "Expanding GÉANT Testbeds Service to Support Pan-European 5G Network Slices for Research in the EuWireless Project," in Mobile Information Systems, Vol. 2019, Article ID 6249247, doi: 10.1155/2019/6249247 [134]

- **J2.** B. Valera-Muros, L. Panizo, A. Rios, and P. Merino-Gomez, "An Architecture for Creating Slices to Experiment on Wireless Networks," in Journal of Network and Systems Management 29, 1 (2021), doi: 10.1007/s10922-020-09571-8 [169]

- **J3.** K. C. Apostolakis, B. Valera-Muros, N. di Pietro, P. Garrido, D. del Teso, M. Kamarianakis, P. Tomás, H. Khalili, L. Panizo, A. Díaz Zayas, A. Protopsaltis, G. Margetis, J. Mangues-Bafalluy, M. Requena-Esteso, A. Gomes, L. Cordeiro, G. Papagiannakis, and C. Stephanidis, "A network application approach towards 5G and beyond critical communications use cases," in Frontiers in Communications and Networks, Vol. 5 - 2024, doi: 10.3389/frcmn.2024.1286660 [23]

UNIVERSIDAD DE MÁLAGA

- **C1.** B. Valera-Muros and P. Merino-Gomez, "Is GÉANT Testbeds Service compliant with ETSI MANO?," in Proceedings of the 2019 IEEE 2nd 5G World Forum (5GWF), Dresden, Germany, 2019, pp. 502-507, doi: 10.1109/5GWF.2019.8911622 [168]
- **C2.** I. Harjula, L. Panizo, B. Valera-Muros, J. Pinola, M. Hoppari, and A. Flizikowski, "Dynamic Spectrum Management for European-Wide Research Network," in Proceedings of the 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), Antwerp, Belgium, 2020, pp. 1-6, doi: 10.1109/VTC2020-Spring 48590.2020.9129017 [83]
- **C3.** G. Margetis, B. Valera-Muros, K. C. Apostolakis, A. Díaz Zayas, L. Panizo, P. Tomás, "Validation of NFV management and orchestration on Kubernetes-based 5G testbed environment," 2022 IEEE Globecom Workshops (GC Wkshps), Rio de Janeiro, Brazil, 2022, pp. 844-849, doi: 10.1109/GCWkshps56602.2022. 10008690 [113]

7.3.2 *Other dissemination activities*

- **D1.** Demo presentation of the EuWireless Project in 28th edition of European Conference on Networks and Communications (EuCNC) Valencia, Spain, Jun. 2019.
- **D2.** Oral presentation of "Is GÉANT Testbeds Service compliant with ETSI MANO?," in Proceedings of the XXVI Jornadas de Concurrencia y Sistemas Distribuidos (JCSD '19) Zaragoza, Spain, Jun. 2019.
- **D3.** Oral presentation of "Expanding GÉANT Testbeds Service to support Pan-European 5G network slices for research in the EuWireless project," in Proceedings of the XIV Jornadas de Ingeniería Telemática (JITEL '19) Zaragoza, Spain, Oct. 2019, doi: 10.26754/uz.978-84-09-21112-8

7.3.3 *Related Projects and Funding*

- **Project EuWireless;** European Union's Horizon 2020 research and innovation program under grant agreement No. 777517.
- **Project 5GENESIS;** European Union's Horizon 2020 research and innovation program under grant agreement No. 815178.
- **Project 5G-EPICENTRE;** European Union's Horizon 2020 innovation action program under grant agreement No. 101016521.
- **Project EVOLVED5G;** European Union's Horizon 2020 innovation action program under grant agreement No. 101016608.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

BIBLIOGRAPHY

[1] 3GPP. *TR 21.902, Evolution of 3GPP system*. Tech. rep. ETSI, 2003. URL: https://www.3gpp.org/ftp//Specs/archive/21_series/21.902/.

[2] 3GPP. *TS 23.060, version 4.11.0 Release 4; General Packet Radio Service (GPRS); Service description; Stage 2*. Tech. rep. ETSI, 2006. URL: https://www.3gpp.org/ftp//Specs/archive/29_series/23.060/.

[3] 3GPP. *TS 36.101, Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception*. Tech. rep. ETSI, 2007. URL: https://www.3gpp.org/ftp//Specs/archive/36_series/36.101/.

[4] 3GPP. *TS 36.104, Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) radio transmission and reception*. Tech. rep. ETSI, 2007. URL: https://www.3gpp.org/ftp//Specs/archive/36_series/36.104/.

[5] 3GPP. *TS 21.201, Technical Specifications and Technical Reports for an Evolved Packet System (EPS) based 3GPP system*. Tech. rep. ETSI, 2008. URL: https://www.3gpp.org/ftp//Specs/archive/21_series/21.201/.

[6] 3GPP. *TR 22.862, version 14.1.0 Release 14; Feasibility study on new services and markets technology enablers for critical communications; Stage 1*. Tech. rep. ETSI, Oct. 2016. URL: https://www.3gpp.org/ftp//Specs/archive/22_series/22.862/.

[7] 3GPP. *TS 23.501, version 15.2.0 Release 15; System Architecture for the 5G System*. Tech. rep. ETSI, June 2018. URL: https://www.3gpp.org/ftp//Specs/archive/23_series/23.501/.

[8] 3GPP. *TS 23.502, version 15.2.0 Release 15; Procedures for the 5G System*. Tech. rep. ETSI, June 2018. URL: https://www.3gpp.org/ftp//Specs/archive/23_series/23.502/.

[9] 3GPP. *TS 23.503 , version 15.2.0 Release 15; Policy and Charging Control Framework for the 5G System*. Tech. rep. 23.503. 2018. URL: https://www.3gpp.org/ftp//Specs/archive/23_series/23.503/.

[10] 3GPP. *TS 29.002, version 15.5.0 Release 15; Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); Mobile Application Part (MAP) specification*. Tech. rep. ETSI, 2019. URL: https://www.3gpp.org/ftp//Specs/archive/29_series/29.002/.

UNIVERSIDAD DE MÁLAGA

[11] 3GPP. *TS 22.261, version 17.9.0 Release 17; Service requirements for the 5G system*. Tech. rep. ETSI, Dec. 2021. URL: https://www.3gpp.org/ftp//Specs/archive/22_series/22.261/.

[12] 3GPP. *TS 22.179, version 19.1.0; Mission Critical Push to Talk (MCPTT); Stage 1*. Tech. rep. ETSI, 2023. URL: https://www.3gpp.org/ftp//Specs/archive/22_series/22.179/.

[13] Amazon Web Services (AWS). *Whitepaper: ETSI NFVO Compliant Orchestration in the Kubernetes/Cloud Native World*. Oct. 2022. URL: https://docs.aws.amazon.com/whitepapers/latest/ETSI-NFVO-compliant-orchestration-in-kubernetes/ETSI-NFVO-compliant-orchestration-in-kubernetes.html.

[14] R. Abhishek, D. Tipper, and D. Medhi. «Network Virtualization and Survivability of 5G Networks: Framework, Optimization Model, and Performance.» In: *2018 IEEE Globecom Workshops (GC Wkshps)* (2018), pp. 1–6.

[15] R. Abhishek, D. Tipper, and D. Medhi. «Resilience of 5G Networks in the Presence of Unlicensed Spectrum and Non-Terrestrial Networks.» In: *2020 16th International Conference on the Design of Reliable Communication Networks DRCN 2020*. IEEE, Mar. 2020, pp. 1–6. DOI: 10.1109/drcn48652.2020.1570604438.

[16] I. Afolabi, J. Prados, M. Bagaa, T. Taleb, and P. Ameigeiras. «Dynamic Resource Provisioning of a Scalable E2E Network Slicing Orchestration System.» In: *IEEE Transactions on Mobile Computing* (2019), pp. 1–1. DOI: 10.1109/TMC.2019.2930059.

[17] S. Agarwal, F. Malandrino, C. Chiasserini, and S. De. «Joint VNF Placement and CPU Allocation in 5G.» In: *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*. 2018, pp. 1943–1951. DOI: 10.1109/INFOCOM.2018.8485943.

[18] O. U. Akguel, I. Malanchini, V. Suryaprakash, and A. Capone. «Service-Aware Network Slice Trading in a Shared Multi-Tenant Infrastructure.» In: *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*. 2017, pp. 1–7. DOI: 10.1109/GLOCOM.2017.8254586.

[19] S. Aleyadeh, A. Moubayed, and A. Shami. «Mobility Aware Edge Computing Segmentation Towards Localized Orchestration.» In: *2021 International Symposium on Networks, Computers and Communications (ISNCC)*. 2021, pp. 1–6. DOI: 10.1109/ISNCC52172.2021.9615795.

[20] G. Amponis, T. Lagkas, M. Zevgara, G. Katsikas, T. Xirofotos, I. Moscholios, and P. Sarigiannidis. «Drones in B5G/6G Networks as Flying Base Stations10.1109/IOTM.001.2200214.» In: *Drones* 6.2 (Feb. 2022), p. 39. ISSN: 2504-446X. DOI: 10.3390/drones6020039.

UNIVERSIDAD DE MÁLAGA

[21] A. Antonescu, P. Robinson, and T. Braun. «Dynamic Topology Orchestration for Distributed Cloud-Based Applications.» In: *2012 Second Symposium on Network Cloud Computing and Applications*. 2012, pp. 116–123. DOI: 10.1109/NCCA.2012.14.

[22] K. C. Apostolakis et al. «Cloud-Native 5G Infrastructure and Network Applications (NetApps) for Public Protection and Disaster Relief: The 5G-EPICENTRE Project.» In: *2021 Joint European Conference on Networks and Communications; 6G Summit (EuCNC/6G Summit)*. IEEE, June 2021. DOI: 10.1109/eucnc/6gsummit51104.2021.9482425.

[23] K. C. Apostolakis et al. «A network application approach towards 5G and beyond critical communications use cases.» In: *Frontiers in Communications and Networks* 5 (Feb. 2024). ISSN: 2673-530X. DOI: 10.3389/frcmn.2024.1286660.

[24] D. Arampatzis et al. «Unification architecture of cross-site 5G testbed resources for PPDR verticals.» In: *2021 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*. IEEE, Sept. 2021. DOI: 10.1109/meditcom49071.2021.9647591.

[25] G. Arfaoui et al. «A Security Architecture for 5G Networks.» In: *IEEE Access* 6 (2018), pp. 22466–22479. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2018.2827419.

[26] O. Arouk and N. Nikaein. «Kube5G: A Cloud-Native 5G Service Platform.» In: *IEEE Global Communications Conf. (GLOBECOM)*. 2020, pp. 1–6. DOI: 10.1109/GLOBECOM42002.2020.9348073.

[27] S. T. Arzo, C. Naiga, F. Granelli, R. Bassoli, M. Devetsikiotis, and F. H. P. Fitzek. «A Theoretical Discussion and Survey of Network Automation for IoT: Challenges and Opportunity.» In: *IEEE Internet of Things Journal* 8.15 (Aug. 2021), pp. 12021–12045. ISSN: 2372-2541. DOI: 10.1109/jiot.2021.3075901.

[28] J. Augé and M.Enguehar. «A network protocol for distributed orchestration using intent-based forwarding.» In: *2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. 2019, pp. 718–719.

[29] I. Baldin, A. Nikolich, J. Griffioen, I. I. S. Monga, K. Wang, T. Lehman, and P. Ruth. «FABRIC: A National-Scale Programmable Experimental Network Infrastructure.» In: *IEEE Internet Computing* 23.6 (2019), pp. 38–47. DOI: 10.1109/MIC.2019.2958545.

[30] A. Basta, A. Blenk, Y. Lai, and W. Kellerer. «HyperFlex: Demonstrating control-plane isolation for virtual software-defined networks.» In: *2015 IFIP/IEEE International Symposium on Integrated*

UNIVERSIDAD DE MÁLAGA

Network Management (IM). 2015, pp. 1163–1164. DOI: 10.1109/INM.2015.7140460.

[31] A. Baumgartner, V. S. Reddy, and T. Bauschert. «Mobile core network virtualization: A model for combined virtual core network function placement and topology optimization.» In: *Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft)*. 2015, pp. 1–9. DOI: 10.1109/NETSOFT.2015.7116162.

[32] M. Berman, J. S. Chase, L. Landweber, A. Nakao, M. Ott, D. Raychaudhuri, R. Ricci, and I. Seskar. «GENI: A federated testbed for innovative network experiments.» In: *Computer Networks* 61 (2014), pp. 5–23. DOI: 10.1016/j.bjp.2013.12.037.

[33] M. Berman, C. Elliott, and L. Landweber. «GENI: Large-scale distributed infrastructure for networking and distributed systems research.» In: *2014 IEEE Fifth International Conference on Communications and Electronics (ICCE)*. 2014, pp. 156–161. DOI: 10.1109/CCE.2014.6916696.

[34] B. Blanco et al. «Technology pillars in the architecture of future 5G mobile networks: NFV, MEC and SDN.» In: *Computer Standards & Interfaces* 54 (2017), pp. 216 –228. ISSN: 0920-5489. DOI: https://doi.org/10.1016/j.csi.2016.12.007.

[35] M. Boussard, D. Bui, R. Douville, P. Justen, N. Le Sauze, P. Peloso, F. Vandeputte, and V. Verdot. «Future Spaces: Reinventing the Home Network for Better Security and Automation in the IoT Era.» In: *Sensors* 18 (Sept. 2018), p. 2986. DOI: 10.3390/s18092986.

[36] M. Boussard, N. L. Sauze, S. Papillon, P. Peloso, and R. Varloot. «Secure Application-Oriented Network Micro-Slicing.» In: *2019 IEEE Conference on Network Softwarization (NetSoft)*. 2019, pp. 248–250. DOI: 10.1109/NETSOFT.2019.8806629.

[37] T. V. K. Buyakar, A. PC, B. R. Tamma, and A. Franklin. «Scalable Network Slicing Architecture for 5G.» In: *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. MobiCom '18. ACM, Oct. 2018, pp. 684–686. DOI: 10.1145/3241539.3267762.

[38] M. Caballer, S. Zala, Á. López García, G. Moltó, P. Orviz Fernández, and M. Velten. «Orchestrating Complex Application Architectures in Heterogeneous Clouds.» In: *Journal of Grid Computing* 16.1 (2017), pp. 3–18. DOI: 10.1007/s10723-017-9418-y.

[39] A. Capone, M. Cesana, I. Malanchini, and V. Suryaprakash. «Making the case for a real-time market of wireless resources with dynamic network slicing and sharing.» In: *2017 IEEE 22nd International Workshop on Computer Aided Modeling and*

UNIVERSIDAD DE MÁLAGA

Design of Communication Links and Networks (CAMAD). 2017, pp. 1–5. DOI: 10.1109/CAMAD.2017.8031630.

[40] G. Carella, M. Pauls, A. Medhat, L. Grebe, and T. Magedanz. «A Network Function Virtualization framework for Network Slicing of 5G Networks.» In: May 2017.

[41] G. Castellano, F. Esposito, and F. Risso. «A Distributed Orchestration Algorithm for Edge Computing Resources with Guarantees.» In: *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. 2019, pp. 2548–2556. DOI: 10.1109/INFOCOM.2019.8737532.

[42] D. A. Chekired, M. A. Togou, L. Khoukhi, and A. Ksentini. «5G-Slicing-Enabled Scalable SDN Core Network: Toward an Ultra-Low Latency of Autonomous Driving Service.» In: *IEEE Journal on Selected Areas in Communications* 37.8 (Aug. 2019), pp. 1769–1782. ISSN: 1558-0008. DOI: 10.1109/jsac.2019.2927065.

[43] Y. L. Chen and A. Bernstein. «Bridging the Gap Between ETSI-NFV and Cloud Native Architecture.» In: *Proc. SCTE/ISBE Fall Tech. Forum*. 2017, pp. 1–27.

[44] D. Chowdhury, R. Das, R. Rana, A. D. Dwivedi, P. Chatterjee, and R. R. Mukkamala. «AUTODEEPSLICE: A Data Driven Network Slicing Technique of 5G network using Automatic Deep Learning.» In: *2022 IEEE Globecom Workshops (GC Wkshps)*. IEEE, Dec. 2022, pp. 450–454. DOI: 10.1109/gcwkshps56602.2022.10008652.

[45] S. R. Chowdhury, M. A. Salahuddin, N. Limam, and R. Boutaba. «Re-Architecting NFV Ecosystem with Microservices: State of the Art and Research Challenges.» In: *IEEE Network* 33.3 (May 2019), pp. 168–176. ISSN: 1558-156X. DOI: 10.1109/mnet.2019.1800082.

[46] J. Chung, N. Pho, and I. Armuelles Voinov. «Cell-Orch: Towards End-to-End Orchestration of Multi-domain 5G Networks.» In: *2018 25th International Conference on Telecommunications (ICT)*. IEEE, June 2018, pp. 416–420. DOI: 10.1109/ict.2018.8464862.

[47] L. Cordeiro, P. Tomás, N. di Pietro, E. Atxutegi, and A. Díaz Zayas. *Quality of Service Control Mechanisms to Support PPDR Network Applications in 5G and Beyond*. 2023 EuCNC & 6G Summit - Posters. Available at https://www.5gepicentre.eu/wp-content/uploads/2023/06/Cordeiro23_QoS_Control_Mechanisms_Support_PPDR_NetApp_5G_Beyond_final.pdf. 2023.

UNIVERSIDAD DE MÁLAGA

[48] V. A. Cunha, E. da Silva, M. B. de Carvalho, D. Corujo, J. P. Barraca, D. Gomes, L. Z. Granville, and R. L. Aguiar. «Network slicing security: Challenges and directions.» In: *Internet Technology Letters* 2.5 (2019), e125. DOI: 10.1002/itl2.125.

[49] *D1.1 5G-EPICENTRE experimentation scenarios preliminary version, Public Deliverable*. Tech. rep. 5G-EPICENTRE, 2021.

[50] A. Dalgkitsis, P. Mekikis, A. Antonopoulos, and C. Verikoukis. «Data Driven Service Orchestration for Vehicular Networks.» In: *IEEE Transactions on Intelligent Transportation Systems* 22.7 (2021), pp. 4100–4109. DOI: 10.1109/TITS.2020.3011264.

[51] X. Dang, M. A. Khan, and F. Sivrikaya. «An Autonomous Service-Oriented Orchestration Framework for Software Defined Mobile Networks.» In: *2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*. 2019, pp. 277–284. DOI: 10.1109/ICIN.2019.8685919.

[52] W. M. Danquah and D. Turgay Altilar. «Vehicular Cloud Resource Management, Issues and Challenges: A Survey.» In: *IEEE Access* 8 (2020), pp. 180587–180607. DOI: 10.1109/ACCESS.2020.3027637.

[53] P. Demeester, P. Van Daele, T. Wauters, and H. Hrasnica. «FED-4FIRE: the largest federation of testbeds in Europe.» dut. In: *Building the future internet through FIRE*. UGent, 2016, pp. 87–109. ISBN: 978-87-93519-12-1.

[54] D. Dietrich, C. Papagianni, P. Papadimitriou, and J. S. Baras. «Network function placement on virtualized cellular cores.» In: *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*. 2017, pp. 259–266. DOI: 10.1109/COMSNETS.2017.7945385.

[55] R. Doost-Mohammady, O. Bejarano, L. Zhong, J. R. Cavallaro, E. Knightly, Z. M. Mao, W. W. Li, X. Chen, and A. Sabharwal. «RENEW: Programmable and Observable Massive MIMO Networks.» In: *2018 52nd Asilomar Conference on Signals, Systems, and Computers*. 2018, pp. 1654–1658. DOI: 10.1109/ACSSC.2018.8645391.

[56] S. D'Oro, L. Galluccio, S. Palazzo, and G. Schembra. «A Game Theoretic Approach for Distributed Resource Allocation and Orchestration of Softwarized Networks.» In: *IEEE Journal on Selected Areas in Communications* 35.3 (Mar. 2017), pp. 721–735. ISSN: 0733-8716. DOI: 10.1109/jsac.2017.2672278.

[57] ETSI & Network Operators. *Network Functions Virtualisation. An Introduction, Benefits, Enablers, Challenges & Call for Action*. Oct. 2012. URL: https://portal.etsi.org/NFV/NFV_White_Paper.pdf.

UNIVERSIDAD DE MÁLAGA

[58] ETSI & Network Operators. *Network Functions Virtualisation. Network Operator Perspectives on Industry Progress*. Oct. 2013. URL: https://portal.etsi.org/NFV/NFV_White_Paper2.pdf.

[59] *ETSI GR NFV-IFA 029 V3.3.1 (2019-11) - Network Functions Virtualisation (NFV) Release 3; Architecture; Report on the Enhancements of the NFV architecture towards "Cloud-native" and "PaaS"*. Group Specification. ETSI ISG NFV, 2019.

[60] ETSI ISG NFV. *ETSI White Paper No. 54: Evolving NFV towards the next decade*. May 2023. URL: https://www.etsi.org/images/files/ETSIWhitePapers/ETSI-WP-54-Evolving_NFV_towards_the_next_decade.pdf.

[61] ETSI. *GR NFV-IFA 029, version 3.3.1: Network Functions Virtualisation (NFV) Release 3; Architecture;Report on the Enhancements of the NFV architecture towards "Cloud-native" and "PaaS"*. Tech. rep. Nov. 2019. URL: https://www.etsi.org/deliver/etsi_gr/NFV-IFA/001_099/029/03.03.01_60/gr_NFV-IFA029v030301p.pdf.

[62] ETSI. *TR 103 582, version 1.1.1: EMTEL; Study of use cases and communications involving IoT devices in provision of emergency situations*. Tech. rep. July 2019. URL: https://www.etsi.org/deliver/etsi_tr/103500_103599/103582/01.01.01_60/tr_103582v010101p.pdf.

[63] European Commission. *5G PPP – 5G innovations for verticals with third party services*. https://ec.europa.eu/info/funding-tenders/opportunities/portal/screen/opportunities/topic-details/ict-41-2020. 2019.

[64] T. Faber and J. Wroclawski. «A federated experiment environment for emulab-based testbeds.» In: *2009 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities and Workshops*. IEEE, 2009. DOI: 10.1109/tridentcom.2009.4976238.

[65] F. Farina, P. Szegedi, and J. Sobieski. «GÉANT world testbed facility: Federated and distributed testbeds as a service facility of GÉANT.» In: *2014 26th International Teletraffic Congress (ITC)*. IEEE, Sept. 2014. DOI: 10.1109/itc.2014.6932972.

[66] A. Fendt, S. Lohmuller, L. C. Schmelz, and B. Bauer. «A Network Slice Resource Allocation and Optimization Model for End-to-End Mobile Networks.» In: *2018 IEEE 5G World Forum (5GWF)*. 2018, pp. 262–267. DOI: 10.1109/5GWF.2018.8517075.

[67] R. Ford, A. Sridharan, R. Margolies, R. Jana, and S. Rangan. «Provisioning low latency, resilient mobile edge clouds for 5G.» In: *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 2017, pp. 169–174. DOI: 10.1109/INFCOMW.2017.8116371.

UNIVERSIDAD DE MÁLAGA

[68] X. Foukas, M. M. Nikaein N.and Kassem, M. K. Marina, and K. Kontovasilis. «FlexRAN: A Flexible and Programmable Platform for Software-Defined Radio Access Networks.» In: *Proceedings of the 12th International on Conference on Emerging Networking EXperiments and Technologies*. CoNEXT '16. Irvine, California, USA: ACM, 2016, pp. 427–441. ISBN: 978-1-4503-4292-6. DOI: 10.1145/2999572.2999599. URL: http://doi.acm.org/10.1145/2999572.2999599.

[69] G. Frick, A. P. Tchinda, U. Trick, A. Lehmann, G. Frick, A. P. Tchinda, and B. Ghita. «Distributed NFV Orchestration in a WMN-Based Disaster Network.» In: *2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*. 2018, pp. 168–173. DOI: 10.1109/ICUFN.2018.8436705.

[70] *GS NFV-MAN 001 V1.1.1 Network Function Virtualisation (NFV); Management and Orchestration*. Group Specification. ETSI ISG NFV, 2014.

[71] GSM Association. *An Introduction to Network Slicing*. 2017. URL: https://www.gsma.com/futurenetworks/wp-content/uploads/2017/11/GSMA-An-Introduction-to-Network-Slicing.pdf.

[72] GSM Association. *Network Slicing Use Case Requirements*. 2018.

[73] C. Garcia-Perez, A. Diaz-Zayas, A. Rios, P. Merino, K. Katsalis, C. Chang, S. Shariat, N. Nikaein, P. Rodriguez, and D. Morris. «Improving the efficiency and reliability of wearable based mobile eHealth applications.» In: *Pervasive and Mobile Computing* 40 (2017), pp. 674–691. DOI: 10.1016/j.pmcj.2017.06.021.

[74] C. García-Pérez and P. Merino. «Experimental evaluation of fog computing techniques to reduce latency in LTE networks.» In: *Transactions on Emerging Telecommunications Technologies* 29.4 (2017), e3201. DOI: 10.1002/ett.3201.

[75] A. Giorgetti, F. Paolucci, and P. Castoldi. «Connectivity orchestration in multi-provider elastic optical networks (Invited paper).» In: *2017 9th International Workshop on Resilient Networks Design and Modeling (RNDM)*. 2017, pp. 1–5. DOI: 10.1109/RNDM.2017.8093026.

[76] D. Gkounis, N. Uniyal, A. S. Muqaddas, R. Nejabati, and D. Simeonidou. «Demonstration of the 5GUK Exchange: A Lightweight Platform for Dynamic End-to-End Orchestration of Softwarized 5G Networks.» In: *2018 European Conference on Optical Communication (ECOC)*. 2018, pp. 1–3. DOI: 10.1109/ECOC.2018.8535288.

[77] A. Gosain and I. Seskar. «GENI wireless testbed: An open edge ecosystem for ubiquitous computing applications.» In: *2017 IEEE International Conference on Pervasive Computing and*

UNIVERSIDAD DE MÁLAGA

Communications Workshops (PerCom Workshops). 2017, pp. 54–56. DOI: 10.1109/PERCOMW.2017.7917520.

[78] D. Griffin, M. Rio, P. Simoens, P. Smet, F. Vandeputte, L. Vermoesen, D. Bursztynowski, and F. Schamel. «Service oriented networking.» In: *2014 European Conference on Networks and Communications (EuCNC)*. 2014, pp. 1–5. DOI: 10.1109/EuCNC.2014.6882684.

[79] A. Gudipati, D. Perry, L. E. Li, and S. Katti. «SoftRAN: Software Defined Radio Access Network.» In: *Proceedings of the Second ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking*. HotSDN '13. Hong Kong, China: ACM, 2013, pp. 25–30. ISBN: 978-1-4503-2178-5. DOI: 10.1145/2491185.2491207. URL: http://doi.acm.org/10.1145/2491185.2491207.

[80] T. Guo and R. Arnott. «Active LTE RAN Sharing with Partial Resource Reservation.» In: *2013 IEEE 78th Vehicular Technology Conference (VTC Fall)*. 2013, pp. 1–5. DOI: 10.1109/VTCFall.2013.6692075.

[81] M. Gupta, R. Legouable, M. M. Rosello, M. Cecchi, J. R. Alonso, M. Lorenzo, E. Kosmatos, M. R. Boldi, and G. Carrozzo. «The 5G EVE End-to-End 5G Facility for Extensive Trials.» In: *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*. 2019, pp. 1–5. DOI: 10.1109/ICCW.2019.8757139.

[82] S. Gutz, A. Story, C. Schlesinger, and N. Foster. «Splendid Isolation: A Slice Abstraction for Software-defined Networks.» In: *Proceedings of the First Workshop on Hot Topics in Software Defined Networks*. HotSDN '12. Helsinki, Finland: ACM, 2012, pp. 79–84. ISBN: 978-1-4503-1477-0. DOI: 10.1145/2342441.2342458. URL: http://doi.acm.org/10.1145/2342441.2342458.

[83] I. Harjula, L. Panizo, B. Valera-Muros, J. Pinola, M. Hoppari, A. Flizikowski, and M. Safianowska. «Dynamic Spectrum Management for European-Wide Research Network.» In: *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*. IEEE, May 2020. DOI: 10.1109/vtc2020-spring48590.2020.9129017.

[84] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen. «Vehicular Fog Computing: A Viewpoint of Vehicles as the Infrastructures.» In: *IEEE Transactions on Vehicular Technology* 65.6 (2016), pp. 3860–3873. DOI: 10.1109/TVT.2016.2532863.

[85] D. Hutchison, D. Pezaros, J. Rak, and P. Smith. «On the Importance of Resilience Engineering for Networked Systems in a Changing World.» In: *IEEE Communications Magazine* 61.11 (Nov. 2023), pp. 200–206. ISSN: 1558-1896. DOI: 10.1109/mcom.001.2300057.

UNIVERSIDAD DE MÁLAGA

[86] Intel. *End-to-End Service Instantiation Using Open-Source Management and Orchestration Components*. Mobile World Congress 2016.

[87] M. Jiang, M. Condoluci, and T. Mahmoodi. «Network slicing in 5G: An auction-based model.» In: *2017 IEEE International Conference on Communications (ICC)*. 2017, pp. 1–6. DOI: 10.1109/ICC.2017.7996490.

[88] G. Juve and E. Deelman. «Automating Application Deployment in Infrastructure Clouds.» In: *2011 IEEE Third International Conference on Cloud Computing Technology and Science*. 2011, pp. 658–665. DOI: 10.1109/CloudCom.2011.102.

[89] A. Kaloxylos. «A Survey and an Analysis of Network Slicing in 5G Networks.» In: *IEEE Communications Standards Magazine* 2.1 (Mar. 2018), pp. 60–65. ISSN: 2471-2833. DOI: 10.1109/mcomstd.2018.1700072.

[90] A. T. Z. Kasgari and W. Saad. «Stochastic optimization and control framework for 5G network slicing with effective isolation.» In: *2018 52nd Annual Conference on Information Sciences and Systems (CISS)*. 2018, pp. 1–6. DOI: 10.1109/CISS.2018.8362271.

[91] J. Kempf and P. Yegani. «OpenRAN: a new architecture for mobile wireless Internet radio access networks.» In: *IEEE Communications Magazine* 40.5 (2002), pp. 118–123. ISSN: 0163-6804. DOI: 10.1109/35.1000222.

[92] Y. Khettab, M. Bagaa, D. L. C. Dutra, T. Taleb, and N. Toumi. «Virtual security as a service for 5G verticals.» In: *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. 2018, pp. 1–6. DOI: 10.1109/WCNC.2018.8377298.

[93] A. Khichane, I. Fajjari, N. Aitsaadi, and M. Gueroui. «Cloud Native 5G: an Efficient Orchestration of Cloud Native 5G System.» In: *IEEE/IFIP Network Operations and Management Symp. (NOMS)*. 2022, pp. 1–9. DOI: 10.1109/NOMS54207.2022.9789856.

[94] H. Kim, Y. el Khamra, I. Rodero, S. Jha, and M. Parashar. «Autonomic Management of Application Workflows on Hybrid Computing Infrastructure.» In: *Scientific Programming* 19 (Jan. 2011). DOI: 10.3233/SPR-2011-0319.

[95] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan. «NVS: A Substrate for Virtualizing Wireless Resources in Cellular Networks.» In: *IEEE/ACM Transactions on Networking* 20.5 (2012), pp. 1333–1346. ISSN: 1063-6692. DOI: 10.1109/TNET.2011.2179063.

UNIVERSIDAD DE MÁLAGA

[96] Z. Kotulski, T. W. Nowak, M. Sepczuk, and M. A. Tunia. «Graph-Based Quantitative Description of Networks' Slices Isolation.» In: *2018 Federated Conference on Computer Science and Information Systems (FedCSIS)*. 2018, pp. 369–379.

[97] Z. Kotulski, T. Nowak, M. Sepczuk, M. Tunia, R. Artych, K. Bocianiak, T. Osko, and J. Wary. «On end-to-end approach for slice isolation in 5G networks. Fundamental challenges.» In: *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*. 2017, pp. 783–792. DOI: 10.15439/2017F228.

[98] H. Koumaras et al. «5GENESIS: The Genesis of a flexible 5G Facility.» In: *2018 IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*. 2018, pp. 1–6. DOI: 10.1109/CAMAD.2018.8514956.

[99] A. Ksentini and N. Nikaein. «Toward Enforcing Network Slicing on RAN: Flexibility and Resources Abstraction.» In: *IEEE Communications Magazine* 55.6 (2017), pp. 102–108. ISSN: 0163-6804. DOI: 10.1109/MCOM.2017.1601119.

[100] H. Kukkalli, S. Maheshwari, I. Seskar, and M. Skorupski. «Evaluation of Multi-operator dynamic 5G Network Slicing for Vehicular Emergency Scenarios.» In: *2020 IFIP Networking Conference (Networking)*. 2020, pp. 761–766.

[101] J. Lee and Y. Kim. «A Design of MANO System for Cloud Native Infrastructure.» In: *2021 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, Oct. 2021. DOI: 10.1109/ictc52510.2021.9620858.

[102] E. A. Lemamou, P. Galinier, and S. Chamberland. «A Hybrid Iterated Local Search Algorithm for the Global Planning Problem of Survivable 4G Wireless Networks.» In: *IEEE/ACM Transactions on Networking* 24.1 (Feb. 2016), pp. 137–148. ISSN: 1558-2566. DOI: 10.1109/tnet.2014.2362356.

[103] X. Li, M. Samaka, H. A. Chan, D. Bhamare, L. Gupta, C. Guo, and R. Jain. «Network Slicing for 5G: Challenges and Opportunities.» In: *IEEE Internet Computing* 21.5 (2017), pp. 20–27. DOI: 10.1109/MIC.2017.3481355.

[104] C. Liang and F. R. Yu. «Wireless Network Virtualization: A Survey, Some Research Issues and Challenges.» In: *IEEE Communications Surveys Tutorials* 17.1 (2015), pp. 358–380. ISSN: 1553-877X. DOI: 10.1109/COMST.2014.2352118.

[105] L. Liang, Y. Wu, G. Feng, X. Jian, and Y. Jia. «Online Auction-Based Resource Allocation for Service-Oriented Network Slicing.» In: *IEEE Transactions on Vehicular Technology* 68.8 (2019), pp. 8063–8074. ISSN: 0018-9545. DOI: 10.1109/TVT.2019.2924456.

UNIVERSIDAD DE MÁLAGA

[106] H. Lim and Y. Kim. «A Design of Service Function Chaining with VNF and CNF on Cloud Native Environment.» In: *Int. Conf. on Information and Communication Technology Convergence (ICTC)*. 2021, pp. 1467–1469. DOI: 10.1109/ICTC52510.2021.9620867.

[107] C. Liu, Y. Mao, J. Van der Merwe, and M. Fernandez. «Cloud resource orchestration: A data-centric approach.» In: *Proceedings of the biennial Conference on Innovative Data Systems Research (CIDR)*. Citeseer. 2011, pp. 1–8.

[108] J. K. Liu, C. Chu, S. S. M. Chow, X. Huang, M. H. Au, and J. Zhou. «Time-Bound Anonymous Authentication for Roaming Networks.» In: *IEEE Transactions on Information Forensics and Security* 10.1 (2015), pp. 178–189. ISSN: 1556-6013. DOI: 10.1109/TIFS.2014.2366300.

[109] Q. Liu and T. Han. «DIRECT: Distributed Cross-Domain Resource Orchestration in Cellular Edge Computing.» In: *Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. Mobihoc '19. Catania, Italy: ACM, 2019, pp. 181–190. ISBN: 978-1-4503-6764-6. DOI: 10.1145/3323679.3326516. URL: http://doi.acm.org/10.1145/3323679.3326516.

[110] R. Mahindra, M. A. Khojastepour, Honghai Zhang, and S. Rangarajan. «Radio Access Network sharing in cellular networks.» In: *2013 21st IEEE International Conference on Network Protocols (ICNP)*. 2013, pp. 1–10. DOI: 10.1109/ICNP.2013.6733595.

[111] K. Mahmood, P. Grønsund, A. Gavras, D. Kennedy, D. Warren, C. Tranoris, A. F. Cattoni, E. Cau, and P. Muschamp. «On the Design of 5G End-to-End Facility for Performance Evaluation and Use Case Trailing.» In: (2018). DOI: 10.5281/ZENODO.2585492.

[112] Y. Mao, C. Liu, J. Merwe, and M. Fernández. «Cloud Resource Orchestration: A Data-Centric Approach.» In: Jan. 2011, pp. 241–248.

[113] G. Margetis, B. Valera-Muros, K. C. Apostolakis, A. Díaz Zayas, L. Panizo, P. Tomás, L. Cordeiro, J. Henriques, and C. Stephanidis. «Validation of NFV management and orchestration on Kubernetes-based 5G testbed environment.» In: *2022 IEEE Globecom Workshops (GC Wkshps)*. IEEE, Dec. 2022. DOI: 10.1109/gcwkshps56602.2022.10008690.

[114] E. Marku, G. Biczók, and C. Boyd. «Towards protected VNFs for multi-operator service delivery.» In: *2019 IEEE Conference on Network Softwarization (NetSoft)*. 2019, pp. 19–23. DOI: 10.1109/NETSOFT.2019.8806681.

UNIVERSIDAD DE MÁLAGA

[115] P. Merino, L. Panizo, and A. Díaz. «EuWireless: design of a pan-European mobile network operator for research.» In: *Proc. of European Conference on Networks and Communications, EuCNC 2018, Ljubljana, Slovenia*. IEEE Computer Society, Jan. 2018.

[116] G. Miranda, J. Haxhibeqiri, N. Slamnik-krijestorac, X. Jiao, J. Hoebeke, I. Moerman, D. F. Macedo, and J. M. Marquez-Barja. «The Quality-Aware and Vertical-Tailored Management of Wireless Time-Sensitive Networks.» In: *IEEE Internet of Things Magazine* 5.4 (Dec. 2022), pp. 142–148. ISSN: 2576-3199. DOI: 10.1109/iotm.001.2200214.

[117] NFV ETSI ISG. *Group Specification NFV Architectural Framework 002*. Oct. 2013. URL: https://www.etsi.org/deliver/etsi_gs/nfv/001_099/002.

[118] NFV ETSI ISG. *Group Specification NFV Use Cases 001*. Oct. 2013. URL: https://www.etsi.org/deliver/etsi_gs/nfv/001_099/001.

[119] J. Ni, X. Lin, and X. S. Shen. «Efficient and Secure Service-Oriented Authentication Supporting Network Slicing for 5G-Enabled IoT.» In: *IEEE Journal on Selected Areas in Communications* 36.3 (2018), pp. 644–657. ISSN: 0733-8716. DOI: 10.1109/JSAC.2018.2815418.

[120] D. Nojima, Y. Katsumata, T. Shimojo, Y. Morihiro, T. Asai, A. Yamada, and S. Iwashina. «Resource Isolation in RAN Part While Utilizing Ordinary Scheduling Algorithm for Network Slicing.» In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. 2018, pp. 1–5. DOI: 10.1109/VTCSpring.2018.8417638.

[121] B. Nour, A. Ksentini, N. Herbaut, P. A. Frangoudis, and H. Moungla. «A Blockchain-Based Network Slice Broker for 5G Services.» In: *IEEE Networking Letters* 1.3 (2019), pp. 99–102. DOI: 10.1109/LNET.2019.2915117.

[122] R. F. Olimid and G. Nencioni. «5G Network Slicing: A Security Overview.» In: *IEEE Access* 8 (2020), pp. 99999–100009. DOI: 10.1109/access.2020.2997702.

[123] J. Ordonez-Lucena, O. Adamuz-Hinojosa, P. Ameigeiras, J. J. Ramos-Munoz, P. Munoz, J. Folgueira Chavarria, and D. Lopez. «The Creation Phase in Network Slicing: From a Service Order to an Operative Network Slice.» In: *2018 European Conference on Networks and Communications (EuCNC)*. IEEE, June 2018. DOI: 10.1109/eucnc.2018.8443255.

[124] T. Ouyang, Z. Zhou, and X. Chen. «Follow Me at the Edge: Mobility-Aware Dynamic Service Placement for Mobile Edge Computing.» In: *IEEE Journal on Selected Areas in Communications* 36.10 (2018), pp. 2333–2345. DOI: 10.1109/JSAC.2018.2869954.

UNIVERSIDAD DE MÁLAGA

[125] L. Panizo, A. Díaz, and B. García. «Model-based testing of apps in real network scenarios.» In: *International Journal on Software Tools for Technology Transfer* 22.2 (Apr. 2019), pp. 105–114. ISSN: 1433-2787. DOI: 10.1007/s10009-019-00518-2.

[126] C. Patachia-Sultanoiu, I. Bogdan, G. Suciu, A. Vulpe, O. Badita, and B. Rusti. «Advanced 5G Architectures for Future NetApps and Verticals.» In: *2021 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*. IEEE, May 2021. DOI: 10.1109/blackseacom52164.2021.9527889.

[127] E. Pateromichelakis, D. Dimopoulos, and A. Salkintzis. «NetApps Enabling Application-Layer Analytics for Vertical IoT Industry10.3390/drones6020039.» In: *IEEE Internet of Things Magazine* 5.4 (Dec. 2022), pp. 130–135. ISSN: 2576-3199. DOI: 10.1109/iotm.001.2200212.

[128] P. Peloso, D. T. Bui, and M. Boussard. «Enforcing users' constraints in dynamic, software-defined networks of devices.» In: *2017 19th Asia-Pacific Network Operations and Management Symposium (APNOMS)*. 2017, pp. 106–111. DOI: 10.1109/APNOMS.2017.8094187.

[129] A. Pino, P. Khodashenas, X. Hesselbach, E. Coronado, and S. Siddiqui. «Validation and Benchmarking of CNFs in OSM for pure Cloud Native applications in 5G and beyond.» In: *Int. Conf. on Computer Communications and Networks (ICCCN)*. 2021, pp. 1–9. DOI: 10.1109/ICCCN52240.2021.9522356.

[130] N. Pustchi, F. Patwa, and R. Sandhu. «Multi Cloud IaaS with Domain Trust in OpenStack.» In: *Proceedings of the Sixth ACM Conference on Data and Application Security and Privacy*. CODASPY '16. New Orleans, Louisiana, USA: ACM, 2016, pp. 121–123. ISBN: 978-1-4503-3935-3. DOI: 10.1145/2857705.2857745. URL: http://doi.acm.org/10.1145/2857705.2857745.

[131] O. Queseth et al. *5G PPP Architecture Working Group: View on 5G Architecture (Version 2.0, December 2017)*. English. Belgium: European Commission, Dec. 2017.

[132] T. Rakotoarivelo, M. Ott, G. Jourjon, and I. Seskar. «OMF: A control and management framework for networking testbeds.» In: *ACM SIGOPS Operating Systems Review* 43.4 (2010), p. 54. DOI: 10.1145/1713254.1713267.

[133] R. Ranjan, B. Benatallah, S. Dustdar, and M. P. Papazoglou. «Cloud Resource Orchestration Programming: Overview, Issues, and Directions.» In: *IEEE Internet Computing* 19.5 (2015), pp. 46–56. DOI: 10.1109/mic.2015.20.

UNIVERSIDAD DE MÁLAGA

[134] Á. Rios, B. Valera-Muros, P. Merino-Gomez, and J. Sobieski. «Expanding GÉANT Testbeds Service to Support Pan-European 5G Network Slices for Research in the EuWireless Project.» In: *Mobile Information Systems* 2019 (Apr. 2019), pp. 1–13. ISSN: 1875-905X. DOI: 10.1155/2019/6249247.

[135] P. Rost et al. «Network Slicing to Enable Scalability and Flexibility in 5G Mobile Networks.» In: *IEEE Communications Magazine* 55.5 (2017), pp. 72–79. ISSN: 0163-6804. DOI: 10.1109/MCOM.2017.1600920.

[136] C. Rotsos et al. «Network service orchestration standardization: A technology survey.» In: *Computer Standards & Interfaces* 54 (2017), pp. 203 –215. ISSN: 0920-5489. DOI: https://doi.org/10.1016/j.csi.2016.12.006.

[137] M. Baker S. Sesia I. Toufik. *LTE: The UMTS Long Term Evolution. From theory to practice.* Wiley, 2011.

[138] G. Saadon, Y. Haddad, and N. Simoni. «A survey of application orchestration and OSS in next-generation network management.» In: *Computer Standards & Interfaces* 62 (2019), pp. 17–31. DOI: 10.1016/j.csi.2018.07.003.

[139] O. Sallent, J. Perez-Romero, R. Ferrus, and R. Agusti. «On Radio Access Network Slicing from a Radio Resource Management Perspective.» In: *IEEE Wireless Communications* 24.5 (2017), pp. 166–174. ISSN: 1536-1284. DOI: 10.1109/MWC.2017.1600220WC.

[140] K. Samdanis, X. Costa-Perez, and V. Sciancalepore. «From network sharing to multi-tenancy: The 5G network slice broker.» In: *IEEE Communications Magazine* 54.7 (2016), pp. 32–39. DOI: 10.1109/MCOM.2016.7514161.

[141] I. Sarrigiannis, E. Kartsakli, K. Ramantas, A. Antonopoulos, and C. Verikoukis. «Application and Network VNF migration in a MEC-enabled 5G Architecture.» In: *2018 IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*. IEEE, Sept. 2018. DOI: 10.1109/camad.2018.8514943.

[142] V. N. Sathi, M. Srinivasan, P. K. Thiruvasagam, and S. R. M. Chebiyyam. «A Novel Protocol for Securing Network Slice Component Association and Slice Isolation in 5G Networks.» In: *Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. MSWIM '18. Montreal, QC, Canada: ACM, 2018, pp. 249–253. ISBN: 978-1-4503-5960-3. DOI: 10.1145/3242102.3242135. URL: http://doi.acm.org/10.1145/3242102.3242135.

UNIVERSIDAD DE MÁLAGA

[143] D. Sattar and A. Matrawy. «Optimal Slice Allocation in 5G Core Networks.» In: *IEEE Networking Letters* 1.2 (2019), pp. 48–51. ISSN: 2576-3156. DOI: 10.1109/LNET.2019.2908351.

[144] D. Sattar and A. Matrawy. «Towards Secure Slicing: Using Slice Isolation to Mitigate DDoS Attacks on 5G Core Network Slices.» In: *2019 IEEE Conference on Communications and Network Security (CNS)*. 2019, pp. 82–90. DOI: 10.1109/CNS.2019.8802852.

[145] D. Sattar and A. Matrawy. «Proactive and Dynamic Slice Allocation in Sliced 5G Core Networks.» In: *2020 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE, 2020. DOI: 10.1109/isncc49221.2020.9297185.

[146] B. Sayadi et al. «Network Applications: Opening up 5G and beyond networks.» In: (2022). DOI: 10.5281/ZENODO.7123918.

[147] P. Schneider and G. Horn. «Towards 5G Security.» In: *2015 IEEE Trustcom/BigDataSE/ISPA*. Vol. 1. 2015, pp. 1165–1170. DOI: 10.1109/Trustcom.2015.499.

[148] P. Schneider, C. Mannweiler, and S. Kerboeuf. «Providing strong 5G mobile network slice isolation for highly sensitive third-party services.» In: *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. 2018, pp. 1–6. DOI: 10.1109/WCNC.2018.8377166.

[149] V. Sciancalepore, K. Samdanis, X. Costa-Perez, D. Bega, M. Gramaglia, and A. Banchs. «Mobile traffic forecasting for maximizing 5G network slicing resource utilization.» In: *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*. 2017, pp. 1–9. DOI: 10.1109/INFOCOM.2017.8057230.

[150] S. D. A. Shah, M. A. Gregory, S. Li, and R. Dos Reis Fontes. «SDN Enhanced Multi-Access Edge Computing (MEC) for E2E Mobility and QoS Management.» In: *IEEE Access* 8 (2020). DOI: 10.1109/ACCESS.2020.2990292.

[151] H. Shen and G. Liu. «An Efficient and Trustworthy Resource Sharing Platform for Collaborative Cloud Computing.» In: *IEEE Transactions on Parallel and Distributed Systems* 25.4 (2014). DOI: 10.1109/TPDS.2013.106.

[152] A. P. Silva et al. «5GinFIRE: An end-to-end open5G vertical network function ecosystem.» In: *Ad Hoc Networks* 93 (2019), p. 101895. ISSN: 1570-8705. DOI: https://doi.org/10.1016/j.adhoc.2019.101895.

[153] A. A. Simiscuka, A. Yaqoob, and G. Muntean. «FRADIS: A Machine Learning-based Multipath Solution for Differentiated Services in a Network Slicing-enhanced Delivery Environment.» In: *2024 IEEE International Symposium on Broadband Multimedia*

UNIVERSIDAD DE MÁLAGA

Systems and Broadcasting (BMSB). IEEE, June 2024, pp. 1–6. DOI: 10.1109/bmsb62888.2024.10608237.

[154] P. Simoens, L. Van Herzeele, F. Vandeputte, and L. Vermoesen. «Challenges for orchestration and instance selection of composite services in distributed edge clouds.» In: *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. 2015, pp. 1196–1201. DOI: 10.1109/INM.2015.7140466.

[155] N. Slamnik-Krijestorac, G. M. Yilma, M. Liebsch, F. Z. Yousaf, and J. Marquez-Barja. «Collaborative orchestration of multi-domain edges from a Connected, Cooperative and Automated Mobility (CCAM) perspective.» In: *IEEE Transactions on Mobile Computing* (2021), pp. 1–1. DOI: 10.1109/TMC.2021.3118058.

[156] J. Sobieski, S. Naegele-Jackson, B. Pietrzak, F. Farina, K. Kramaric, and M. Hazlinsky. «GÉANT Testbed Service External Domain Ports: A demo on multiple domain connectivity.» In: (Jan. 2015).

[157] B. Sonkoly, F. Nemeth, L. Csikor, L. Gulyas, and A. Gulyas. «SDN based testbeds for evaluating and promoting multipath TCP.» In: *2014 IEEE Int. Conf. on Communications (ICC)*. IEEE, 2014. DOI: 10.1109/icc.2014.6883788.

[158] W. Stallings. *Foundations of Modern Networking. SND, NFV, QoE, IoT and Cloud*. Pearson, 2016.

[159] F. Tabatabaei, H. Khalili, M. Requena, S. Kahvazadeh, and J. Mangues-Bafalluy. «Dynamic Service Placement in 6G Multi-Cloud Scenarios.» In: *2023 23rd International Conference on Transparent Optical Networks (ICTON)*. IEEE, July 2023. DOI: 10.1109/icton59386.2023.10207547.

[160] F. Tabatabaeimehr, M. Ruiz, C. Liu, X. Chen, R. Proietti, S. J. B. Yoo, and L. Velasco. «Cooperative Learning for Disaggregated Delay Modeling in Multidomain Networks.» In: *IEEE Transactions on Network and Service Management* 18.3 (Sept. 2021), pp. 3633–3646. ISSN: 2373-7379. DOI: 10.1109/tnsm.2021.3077736.

[161] T. Taleb, I. Afolabi, K. Samdanis, and F. Z. Yousaf. «On Multi-Domain Network Slicing Orchestration Architecture and Federated Resource Control.» In: *IEEE Network* 33.5 (2019), pp. 242–252. DOI: 10.1109/MNET.2018.1800267.

[162] T. Taleb, A. Ksentini, and B. Sericola. «On Service Resilience in Cloud-Native 5G Mobile Systems.» In: *IEEE Journal on Selected Areas in Communications* 34.3 (Mar. 2016), pp. 483–496. ISSN: 0733-8716. DOI: 10.1109/jsac.2016.2525342.

UNIVERSIDAD DE MÁLAGA

[163] A. Thantharate, R. Paropkari, V. Walunj, and C. Beard. «DeepSlice: A Deep Learning Approach towards an Efficient and Reliable Network Slicing in 5G Networks.» In: *2019 IEEE 10th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*. IEEE, Oct. 2019. DOI: 10.1109/uemcon47517.2019.8993066.

[164] A. Thantharate, R. Paropkari, V. Walunj, P. Kankariya, and C. Beard. «Secure5G: A Deep Learning Framework Towards a Secure Network Slicing in 5G and Beyond.» In: *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2020. DOI: 10.1109/ccwc47524.2020.9031158.

[165] D. Tipper, A. Babay, B. Palanisamy, and P. Krishnamurthy. «Network Connectivity Resilience in Next Generation Backhaul Networks: Challenges and Future Opportunities.» In: *IEEE Transactions on Network and Service Management* 21.5 (Oct. 2024), pp. 5321–5334. ISSN: 2373-7379. DOI: 10.1109/tnsm.2024.3392857.

[166] K. Toczé and S. Nadjm-Tehrani. «ORCH: Distributed Orchestration Framework using Mobile Edge Devices.» In: *2019 IEEE 3rd International Conference on Fog and Edge Computing (ICFEC)*. 2019, pp. 1–10. DOI: 10.1109/CFEC.2019.8733152.

[167] P. Tsai, N. Xia, T. Fang, H. Huang, and C. Yang. «Using Software-Defined Tenant to Improve Network Adaptation in 5G Core Networks.» In: *2024 IEEE 16th International Conference on Advanced Infocomm Technology (ICAIT)*. IEEE, Aug. 2024, pp. 19–24. DOI: 10.1109/icait62580.2024.10808058.

[168] B. Valera-Muros and P. Merino-Gomez. «Is GÉANT Testbeds Service compliant with ETSI MANO?» In: *2019 IEEE 2nd 5G World Forum (5GWF)*. IEEE, Sept. 2019. DOI: 10.1109/5gwf.2019.8911622.

[169] B. Valera-Muros, L. Panizo, A. Rios, and P. Merino-Gomez. «An Architecture for Creating Slices to Experiment on Wireless Networks.» In: *Journal of Network and Systems Management* 29.1 (2020). DOI: 10.1007/s10922-020-09571-8.

[170] S. Vittal, S. Sarkar, and A. A. Franklin. «Revamping the Resilience and High Availability of 5G Core for 6G Ready Network Slices.» In: *IEEE Transactions on Network and Service Management* 21.2 (Apr. 2024), pp. 2287–2302. ISSN: 2373-7379. DOI: 10.1109/tnsm.2023.3348137.

[171] G. Wang, G. Feng, W. Tan, S. Qin, R. Wen, and S. Sun. «Resource Allocation for Network Slices in 5G with Network Resource Pricing.» In: *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*. 2017, pp. 1–6. DOI: 10.1109/GLOCOM.2017.8254074.

UNIVERSIDAD DE MÁLAGA

[172] Q. Wang et al. «SliceNet: End-to-End Cognitive Network Slicing and Slice Management Framework in Virtualised Multi-Domain, Multi-Tenant 5G Networks.» In: *2018 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. 2018, pp. 1–5. DOI: 10.1109/BMSB.2018.8436800.

[173] J. Wettinger, V. Andrikopoulos, S. Strauch, and F. Leymann. «Enabling Dynamic Deployment of Cloud Applications Using a Modular and Extensible PaaS Environment.» In: *2013 IEEE Sixth International Conference on Cloud Computing*. IEEE, 2013. DOI: 10.1109/cloud.2013.68.

[174] J. Wu, Z. Zhang, Y. Hong, and Y. Wen. «Cloud radio access network (C-RAN): a primer.» In: *IEEE Network* 29.1 (2015), pp. 35–41. DOI: 10.1109/MNET.2015.7018201.

[175] X. Wang and P. Krishnamurthy and D. Tipper. «Wireless network virtualization.» In: *2013 International Conference on Computing, Networking and Communications (ICNC)*. 2013.

[176] M. Yampolskiy and M.K. Hamm. «Management of multidomain end-to-end links - a federated approach for the pan-European research network Geant 2.» In: *2007 10th IFIP/IEEE Int. Symposium on Integrated Network Management*. IEEE, 2007. DOI: 10.1109/inm.2007.374783.

[177] N. Yang, L. Wang, G. Geraci, M. Elkashlan, J. Yuan, and M. D. Renzo. «Safeguarding 5G wireless communication networks using physical layer security.» In: *IEEE Communications Magazine* 53.4 (2015), pp. 20–27. ISSN: 0163-6804. DOI: 10.1109/MCOM.2015.7081071.

[178] X. Yang, T. Lehman, R. Kettimuthu, L. Winkler, and E. Jung. «A Model Driven Intelligent Orchestration Approach to Service Automation in Large Distributed Infrastructures.» In: *Proceedings of the 1st International Workshop on Autonomous Infrastructure for Science*. AI-Science'18. Tempe, AZ, USA: ACM, 2018, 5:1–5:8. ISBN: 978-1-4503-5862-0. DOI: 10.1145/3217197.3217207. URL: http://doi.acm.org/10.1145/3217197.3217207.

[179] V. Yazici, U. C. Kozat, and M. O. Sunay. «A new control plane for 5G network architecture with a case study on unified handoff, mobility, and routing management.» In: *IEEE Communications Magazine* 52.11 (2014), pp. 76–85. ISSN: 0163-6804. DOI: 10.1109/MCOM.2014.6957146.

[180] M. Ye, M. B. Cohen, W. Srisa-an, and S. Wei. «EvoIsolator: Evolving Program Slices for Hardware Isolation Based Security.» In: *Search-Based Software Engineering*. Ed. by Thelma Elita Colanzi and Phil McMinn. Cham: Springer International Publishing, 2018, pp. 377–382.

UNIVERSIDAD DE MÁLAGA

[181] A. Yegin, J. Park, K. Kweon, and J. Lee. «Terminal-centric distribution and orchestration of IP mobility for 5G networks.» In: *IEEE Communications Magazine* 52.11 (2014), pp. 86–92. DOI: 10.1109/MCOM.2014.6957147.

[182] E. Yigitoglu, L. Liu, M. Looper, and C. Pu. «Distributed Orchestration in Large-Scale IoT Systems.» In: *2017 IEEE International Congress on Internet of Things (ICIOT)*. 2017, pp. 58–65. DOI: 10.1109/IEEE.ICIOT.2017.16.

[183] H. Yu, F. Musumeci, J. Zhang, M. Tornatore, and Y. Ji. «Isolation-Aware 5G RAN Slice Mapping Over WDM Metro-Aggregation Networks.» In: *Journal of Lightwave Technology* 38.6 (2020). DOI: 10.1109/jlt.2020.2973311.

[184] J. Yu, T. Chen, C. Gutterman, S. Zhu, G. Zussman, I. Seskar, and D. Kilper. «COSMOS: Optical Architecture and Prototyping.» In: *Optical Fiber Communication Conference (OFC) 2019*. Optical Society of America, 2019, M3G.3. DOI: 10.1364/OFC.2019.M3G.3. URL: http://www.osapublishing.org/abstract.cfm?URI=OFC-2019-M3G.3.

[185] W. Zhang, Z. Zhang, and H. Chao. «Cooperative Fog Computing for Dealing with Big Data in the Internet of Vehicles: Architecture and Hierarchical Resource Management.» In: *IEEE Communications Magazine* 55.12 (2017), pp. 60–67. DOI: 10.1109/MCOM.2017.1700208.

[186] M. Zhao et al. «Verification and validation framework for 5G network services and apps.» In: *2017 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. 2017, pp. 321–326. DOI: 10.1109/NFV-SDN.2017.8169878.

[187] F. Zhou, P. Yu, L. Feng, X. Qiu, Z. Wang, L. Meng, M. Kadoch, L. Gong, and X. Yao. «Automatic Network Slicing for IoT in Smart City.» In: *IEEE Wireless Communications* 27.6 (Dec. 2020), pp. 108–115. ISSN: 1558-0687. DOI: 10.1109/mwc.001.2000069.

[188] P. Zikas et al. «MAGES 4.0: Accelerating the World's Transition to VR Training and Democratizing the Authoring of the Medical Metaverse.» In: *IEEE Computer Graphics and Applications* 43.2 (Mar. 2023), pp. 43–56. ISSN: 1558-1756. DOI: 10.1109/mcg.2023.3242686.

[189] xRAN.org. *The Mobile Access Network, beyond connectivity*. White Paper. 2016.

UNIVERSIDAD DE MÁLAGA

Part VI

APPENDIX

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

A

RESUMEN EN ESPAÑOL

A.1 INTRODUCCIÓN

A.1.1 *Motivación*

El éxito de las tecnologías de computación en la nube durante los últimos años se ha visto reflejado directamente en las redes móviles, que ahora adoptan los principios de virtualización para el diseño de la arquitectura del núcleo de las redes 5G y posteriores.

Para asegurar que la integración de estos conceptos se realice con éxito en las redes heredadas, es necesario un entorno que permita desarrollar y experimentar con nuevos diseños y servicios. Sin embargo, la experimentación en redes móviles ha estado limitada tradicionalmente por la necesidad de equipamiento específico y de licencia de espectro radio para realizar experimentos realistas fuera de entornos de laboratorio que restringen la movilidad. Esto ha llevado a la creación de distintas plataformas de experimentación en Europa y Estados Unidos, con la intención de ofrecer a la comunidad investigadora la infraestructura y los recursos necesarios para desplegar redes móviles a escala.

La Tabla 5 incluye un resumen de las plataformas de experimentación existentes más relevantes.

En este contexto, esta tesis plantea la creación de un entorno de investigación para desplegar redes 5G dedicadas que se extiendan geográficamente por múltiples ubicaciones, y cuyos recursos se gestionen mediante un sistema de orquestación distribuido entre los puntos de presencia que componen el entorno de investigación.

A.1.2 *Preguntas de investigación*

Las preguntas de investigación en las que se basa esta tesis son las siguientes

- **Pregunta 1 (P1):** ¿Cuáles son las tecnologías principales en una plataforma de experimentación extremo a extremo para desplegar redes celulares experimentales realistas?
- **Pregunta 2 (P2):** ¿Están las funciones preparadas para soportar la creación de redes temporales customizadas como elemento clave para proporcionar servicios en las redes móviles de nueva generación?

UNIVERSIDAD DE MÁLAGA

Proyecto	Cobertura	Movilidad	Espectro	Extensible	Orquestación
FIRE	Laboratorios fijos (Europa)	Por *testbed*	Por *testbed*	Por *testbed*	Independiente por *testbed*
GENI	Universidades en EEUU	Por proyecto	Banda ancha educación	Por proyecto	Independiente por proyecto
COSMOS	Nueva York	V2X	5G Comercial	Sí	OMF centralizado
Powder	Salt Lake City	V2X	5G Comercial	Sí	Emulab centralizado
5Gtango	No	No	No	Sí	OSM Centralizado
MATILDA	No	No	No	Sí	Multi-plataforma, centralizado
SliceNet	No	No	No	Sí	Multi-dominio, transversal
5GENESIS	Ciudades europeas	Sí	4G & 5G comerciales	Sí	CCentralizado, por plataforma
5G-VINNI	Ciudades europeas	Sí	4G & 5G comerciales	Sí	Centralizado, por plataforma
5G EVE	Ciudades europeas	Sí	4G & 5G comerciales	Sí	Centralizado, por plataforma
FABRIC	Ciudades en EEUU	Sí	5G Comercial	Sí	Centralizado, por *testbed*

Cuadro 5: Comparación de plataformas de experimentación existentes

UNIVERSIDAD DE MÁLAGA

- **Pregunta 3 (P3):** ¿Son suficiente las tecnologías de virtualización tradicionales (como las máquinas virtuales) para cumplir los requisitos de las redes móviles de nueva generación?
- **Pregunta 4 (P4):** ¿Cuáles son los principales retos de los sistemas de orquestación y cuáles son los beneficios y retos específicos de distribuir la orquestación?

A.1.3 *Objetivos y contribuciones de la tesis*

El objetivo principal de esta tesis es el diseño de la arquitectura de una plataforma de experimentación en redes móviles basada en tecnologías de virtualización, demostrando la posibilidad de distribuir la arquitectura tanto de los servicios proporcionados en las redes de última generación como de la orquestación de los recursos. En resumen, los objetivos son:

i La definición de la arquitectura distribuida, teniendo en cuenta en el diseño los componentes, las interfaces, los protocolos, la orquestación y la prestación de servicios.

ii La evaluación de la arquitectura diseñada mediante simulaciones y emulaciones con *software* real.

iii La creación de una plataforma de experimentación con la funcionalidad definida y su demostración mediante la aplicación de casos de uso.

Ya que la hipótesis inicial de la investigación se basa en la mejora que supone distribuir la orquestación de recursos y servicios en el contexto de las redes 5G, se esperan las siguientes contribuciones:

i La identificación de los beneficios heredados de virtualizar y distribuir los servicios.

ii La identificación de las deficiencias de los sistemas de orquestación centralizados.

iii La propuesta de una arquitectura de red celular que integra un sistema de orquestación distribuido para proporcionar servicios 5G también distribuidos, incluyendo el detalle de la arquitectura del sistema y de la comunicación entre las entidades que componen dicha arquitectura.

iv La evaluación práctica de la viabilidad de la propuesta para soportar redes de experimentación 5G mediante la implementación de testbeds.

UNIVERSIDAD DE MÁLAGA

A.2 ANTECEDENTES

En esta sección se presentan los antecedentes de alto nivel de las redes móviles y las tecnologías de virtualización.

A.2.1 *Redes móviles*

Cuando las redes LTE, comúnmente conocidas como redes 4G, se presentaron en 2008 como una evolución de la tecnología UMTS [1], trajeron consigo un cambio radical en la arquitectura de la infraestructura de telecomunicaciones. Este nuevo estándar se basaba en la conmutación de paquetes de datos, siguiendo un modelo IP para toda la comunicación entre los componentes de la red, y sustituyendo así la naturaleza basada en circuitos de las generaciones previas.

Para entender mejor esta evolución en la arquitectura, es necesario conocer los antecedentes de alto nivel de los antecesores de las redes LTE. Empezando por GSM, cuya arquitectura se muestra en Figura 67. GSM se fundamenta en la creación de interfaces abiertas para interconectar los componentes de red de distintos operadores que se integran en la misma red.

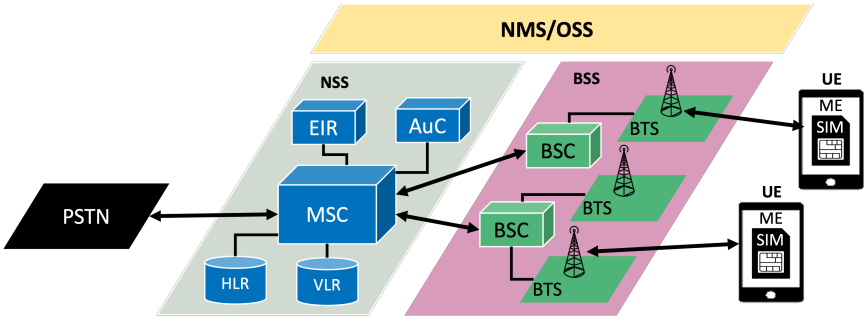

Figura 67: Arquitectura de alto nivel de GSM

Tal y como se observa en la Figura 67, la arquitectura de GSM [10] se divide en subsistemas que descentralizan la gestión de la red, a saber el equipamiento de usuario (UE), los subsistemas de estaciones base (BSS), el subsistema de conmutación de red (NSS) y los sistemas de gestión de la red (NMS) y de soporte a la operación (OSS).

La red de acceso está compuesta por las BSS, mientras que el núcleo de la red, que comunica a los usuarios finales con la red telefónica conmutada (PSTN), incluye el NSS. A su vez, el NSS incluye distintas entidades, siendo la principal la central de conmutación móvil (MSC).

Para poder incluir servicios multimedia, GSM evolucionó a GPRS introduciendo la conmutación IP de paquetes. En esta evolución, tal y como se muestra en la Figura 68, se mantiene la misma red de acceso, mientras que en el núcleo de la red se integran las entidades GPRS para comunicar las estaciones base con la red de datos [2].

UNIVERSIDAD DE MÁLAGA

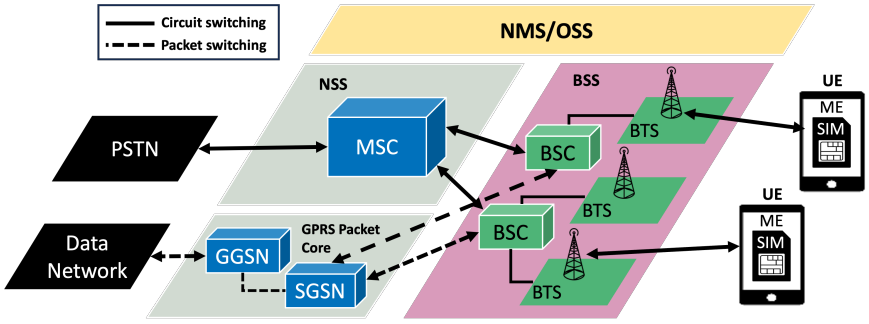

Figura 68: Arquitectura de alto nivel de GPRS

Posteriormente, GPRS fue sustituido por UMTS, que en este caso sí modificaba la red de acceso, ahora llamada UTRAN, tal y como se muestra en la Figura 69, para poder abordar la creciente demanda de tráfico por parte de los usuarios. UMTS se caracteriza por la convergencia del tráfico de datos y voz, la mejora de los servicios multimedia y el incremento de ancho de banda.

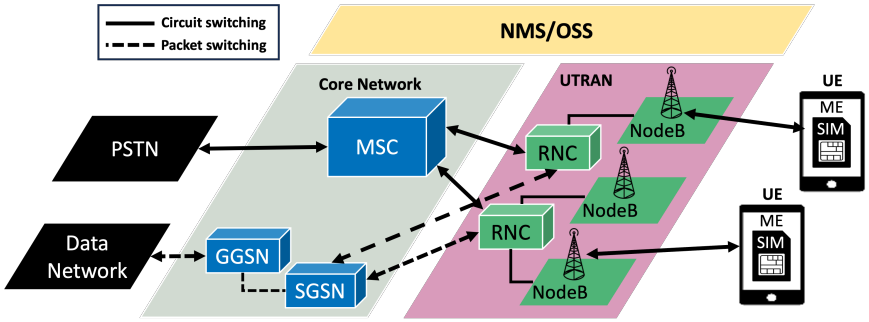

Figura 69: Arquitectura de alto nivel de UMTS

De igual forma, la arquitectura 4G también presenta tres dominios diferenciados: el equipamiento de usuario [3], la red de acceso [4] y el sistema de paquetes interno de los operadores [5]. En este caso, la evolución introducida por la arquitectura afecta tanto a la red de acceso como al núcleo interno.

La red de acceso radio (RAN) se compone de un conjunto de estaciones base llamadas eNB que conectan a los usuarios con el núcleo de la red de los operadores. En este entorno, la inteligencia de la red está descentralizada para facilitar el establecimiento de conexión y reducir las latencias en los traspasos entre celdas. Entre ellos, los eNB se comunican por una interfaz única y dedicada, mientras que la conexión con el núcleo de la red se divide para diferenciar el tráfico del plano de control y el del plano de datos, tal y como se muestra en la Figura 70.

Por su parte, el núcleo de la red 4G incluye el EPC, que también diferencia el plano de control o señalización y el de datos de usuario para simplificar el dimensionamiento de red. En lugar de seguir una arquitectura monolítica, el EPC se compone de una serie de elementos

UNIVERSIDAD DE MÁLAGA

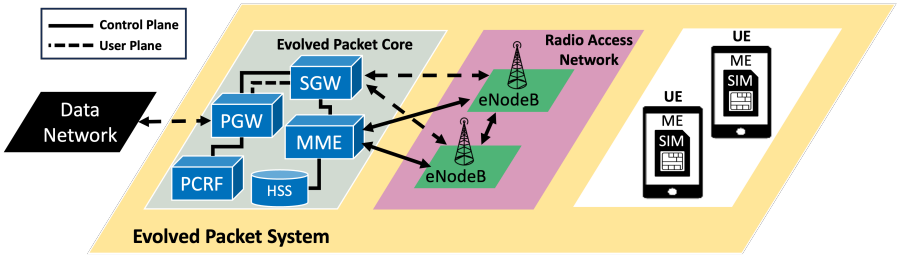

Figura 70: Arquitectura de alto nivel de 4G

o funciones de red que se encargan de realizar unas tareas específicas para proporcionar distintos servicios. De esta forma, mediante la distribución de tareas se busca maximizar el uso y adaptabilidad de cada componente.

En el caso de las redes 5G, la división de las funciones de red es aún mayor, tanto a nivel de núcleo de red como de red de acceso radio, de cara a proporcionar servicio a un creciente número de usuarios con dispositivos de tecnologías heterogéneas, lo que aumenta la complejidad de la red. De nuevo se incluyen tres subsistemas, que suponen una evolución a los susbsistemas 4G. La descripción completa de la arquitectura de las redes 5G se encuentra en las siguientes especificaciones técnicas del 3GPP: TS 23.501 [7], TS 23.502 [8] y TS 23.503[9].

En este contexto, de cara a cumplir con las especificaciones y requisitos de las nuevas verticales a las que las redes 5G pretenden proporcionar servicio, se recurre a nuevos paradigmas de diseño de la arquitectura, introduciendo el concepto de *slicing* de red. Los *slices* de red son básicamente redes virtuales privadas y aisladas que comparten la misma infraestructura física.

La implementación de los *slices* de red se basa principalmente en la virtualización de las funciones de red o NFV, las redes definidas por software o SDN, y la computación en el *Edge* o MEC. En la Figura 71 se muestra la arquitectura de alto nivel de una red 5G desplegada sobre una infraestructura que integra estas tecnologías de virtualización para proporcionar el núcleo de red y la red de transporte.

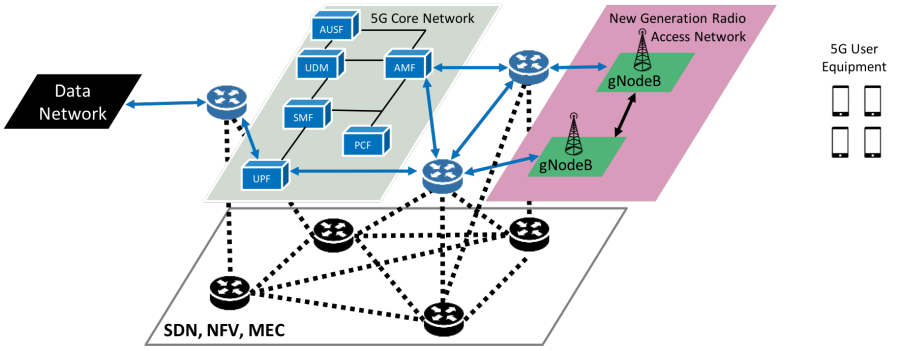

Figura 71: Despliegue 5G sobre una arquitectura SDN

UNIVERSIDAD DE MÁLAGA

Los *slices* de red 5G [34, 71], por tanto, se componen de las funciones de red que requiere un servicio determinado para alcanzar un rendimiento específico, utilizando una infraestructura común que se puede configurar de manera dinámica para situar algunas funciones de red más cerca de los usuarios finales.

En la Figura 72 se muestra un ejemplo de *slicing* de red, donde uno de los *slices* (numerado como 1) representa un despliegue dedicado en el que cada función de red proporciona servicio al mismo *slice*. Por otra parte, los *slices* de red 2 y 3 comparten las funciones de red relacionadas con el plano de control, pero disponen de distintas funciones de red para el plano de usuario, siendo así redes diferenciadas, aunque proporcionen servicio al mismo usuario (UE_b). Los *slices* 2 y 3 por tanto tienen un plano de usuario completamente independiente y aislado, lo que permite una configuración fina de los *slices* asociada a los requisitos específicos de los servicios que proporcionan.

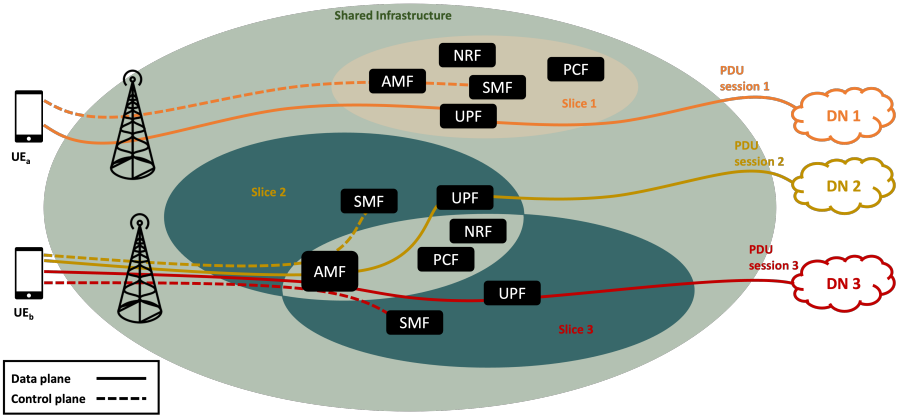

Figura 72: *Slicing* de red 5G

A.2.2 *Tecnologías de virtualización*

El concepto de virtualización agrupa una gran variedad de tecnologías para la gestión de recursos, proporcionando una capa de abstracción entre el *hardware* físico y el *software* que transforma los recursos físicos en lógicos. Mediante un uso más eficiente de los recursos y la separación de las funcionalidades de la infraestructura que las soportan, los costes asociados al despliegue y mantenimiento de los sistemas se ven disminuidos con el uso de las tecnologías de virtualización.

En el contexto de las redes móviles, la virtualización pretende adaptar las redes existente a los cambios estructurales que suponen las nuevas arquitecturas. Así, se plantea que múltiples redes móviles virtuales operadas por distintos proveedores de servicio compartan de manera dinámica el mismo sustrato físico operado por el operador comercial.

UNIVERSIDAD DE MÁLAGA

La virtualización de las funciones de red o NFV se define en [158] como la implementación *software* de las funciones para su ejecución en forma de máquinas virtuales. En las redes móviles tradicionales, esas funciones de red se entendían como *cajas negras* que actuaban como plataformas propietarias con *hardware* dedicado, que de no ser usado permanecía ocioso. Sin embargo, en el paradigma NFV, esas funciones de red se entienden como aplicaciones independientes que se despliegan de manera flexible sobre una infraestructura compartida, y la separación entre el *hardware* y el *software* permite que cada aplicación aumente o disminuya los recursos virtuales en uso en función de su necesidad.

En 2012, la ETSI definió en [57] el concepto de NFV, describiendo la especificación y la arquitectura de referencia para cualquier plataforma NFV. El objetivo de esta estandarización era alcanzar un consenso con la industria en los requisitos técnicos y de negocio de la tecnología NFV para establecer una infraestructura común única. Desde ese momento, NFV se presenta como la solución para satisfacer las necesidades presentes y futuras de proveedores, compañías y usuarios de las redes de comunicación, siendo los casos de uso principales la provisión de infraestructura y red como servicio, IaaS y NaaS, respectivamente. En [168] se identifican los siguientes requisitos como base de la tecnología NFV: portabilidad, rendimiento, elasticidad, resiliencia, seguridad, continuidad de servicio, operación y gestión, eficiencia energética y migración y co-existencia.

A partir del concepto de NFV, se definen las funciones de red virtualizadas (VNF) como los bloques utilizados para proporcionar servicios de red extremo a extremo siguiendo estos tres principios:

- **Encadenamiento de servicios:** Las VNF son componentes modulares que proporcionan una funcionalidad limitada, por lo que para poder proporcionar una función de red determinada se encadena el tráfico a través de múltiples VNFs combinadas.
- **Gestión y Orquestación, o MANO:** Se refiere al despliegue y gestión del ciclo de vida de las instancias NFV, así como a la gestión de los elementos que componen la infraestructura virtualizada.
- **Arquitectura distribuida:** Cada VNF puede estar compuesta a su vez por distintas VNF combinadas, de cara a proporcionar redundancia y escalabilidad.

De esta forma, se establecen los requisitos para cualquier despliegue o implementación NFV. La arquitectura de referencia, definida en [117] por la ETSI, incluye los bloques funcionales y puntos de referencia necesarios para adaptar cualquier arquitectura de red a la tecnología NFV. Esta arquitectura, tal y como muestra la Figura 73, diferencia los siguientes bloques:

UNIVERSIDAD DE MÁLAGA

- El bloque de la infraestructura NFV (NFVI), que incluye los componentes *hardware* de red, computación y almacenamiento, la capa de virtualización que los abstrae, y los recursos virtuales proporcionados por esa abstracción que se necesitan para dar soporte a las VNF.
- El bloque de las VNF, que incluye también los EMS, y donde la VNF representa la implementación *software* de una función de red y el EMS el elemento de gestión de la funcionalidad de esa VNF.
- El bloque del MANO NFV responsable de la orquestación y gestión del entorno NFV al completo, que a su vez incluye el Orquestador, el Gestor de las VNF (VNF) y el Gestor de la Infraestructura Virtual (VIM).
- El bloque con la descripción del servicio, VNF e infraestructura, con la información de platillas, modelos y servicios.
- El bloque OSS/BSS del operador o proveedor de servicio para interactuar con el MANO.

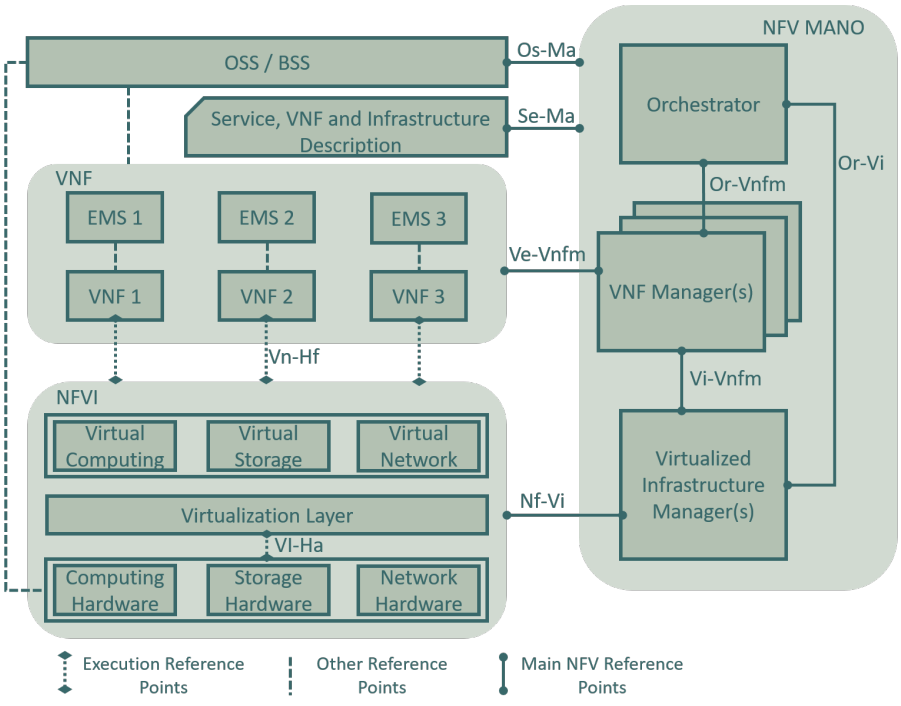

Figura 73: Arquitectura de referencia de NFV [168]

En la práctica, se ha identificado la solución *OpenStack* como posible VIM que cumple con las especificaciones de la referencia definida por la ETSI. *OpenStack* es una plataforma de computación en la nube de código abierto que por su éxito se ha convertido en el estándar *de*

UNIVERSIDAD DE MÁLAGA

facto para despliegues *software* en la nube. *OpenStack* incluye un conjunto de proyectos, cada uno asociado a un servicio, que interactúan a través de un API para integrar y gestionar los recursos disponibles de red, computación y almacenamiento y agilizar la provisión de recursos virtuales bajo demanda.

Esta arquitectura modular se basa principalmente en los siguientes seis proyectos:

- **Keystone:** Para gestionar los servicios de identidad.
- **Nova:** Para gestionar los servicios de computación.
- **Neutron:** Para gestionar los servicios de red.
- **Glance:** Para gestionar los servicios de imágenes.
- **Cinder:** Para gestionar los servicios de almacenamiento de bloques.
- **Swift:** Para gestionar los servicios de almacenamiento de objetos.

Adicionalmente, hay otros proyectos desarrollados para proporcionar servicios asociados a la telemetría (Ceilometer), paneles de *dashboard* (Horizon), sistemas de compartición de archivos (Manila), servicios de alarma (Aodth) y orquestación (Heat).

Respecto a las implementaciones del MANO NFV disponibles, se han desarrollado distintas soluciones de código abierto para cumplir con el estándar de la ETSI, entre las que destacan OSM y *Open Baton*, principalmente por el apoyo que reciben de la industria.

OSM es la implementación de código abierto que proporcionó la ETSI en 2014 tras la publicación del estándar. Es una implementación ampliamente respaldada por la ETSI y la comunidad formada por los operadores comerciales [138], ya que es altamente interoperable con con diversos VIM. OSM sigue una arquitectura basada en unos principios de capas, abstracciones, modularidad y simplicidad, e incluye, tal y como muestra la Figura 74, los siguientes componentes [86]:

- **La interfaz gráfica de usuario o GUI** para que los usuarios accedan al sistema.
- **Los paquetes** con los descriptores y plantillas disponibles en el GUI para que los usuarios los utilicen.
- **El orquestador del servicio o SO** que recibe y procesa los paquetes del GUI con las descripciones de las VNF y de los servicios de red (NS). El SO gestiona el ciclo de vida de los servicios.
- **El orquestador de los recursos o RO** que procesa las peticiones del SO y opera sobre el VIM para asignar los recursos y crear las instancias necesarias para cumplir con los requisitos definidos en los descriptores.

UNIVERSIDAD DE MÁLAGA

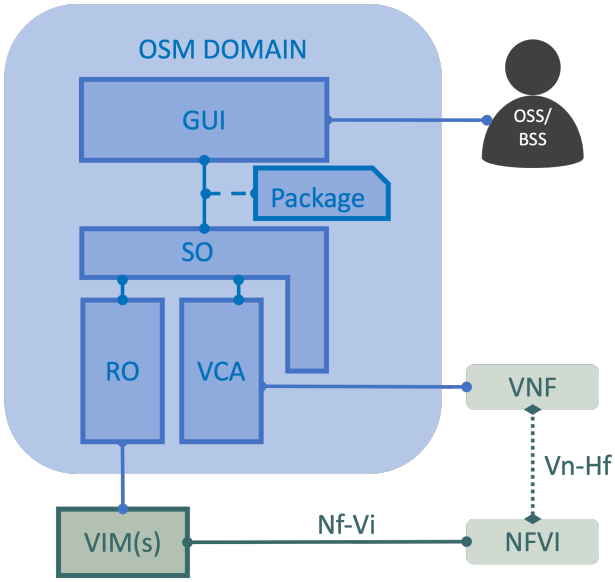

Figura 74: Arquitectura de OSM

- **El componente de abstracción y configuración de las VNF o VCA** que recibe la configuración de los descriptores y las peticiones del SO cuando las VNF están operativas, actuando como VNFM.

Otra alternativa es *Open Baton*, que fue desarrollado por Fraunhofer FOKUS and TU Berlin en 2015 basándose en la especificación de la ETSI. Al igual que con OSM, los usuarios acceden al sistema por la interfaz gráfica o GUI, que se conecta con el orquestador NFV o NFVO, que en este caso incluye también componentes OSS [40]. Tal y como muestra la Figura 75, *Open Baton* incluye múltiples *drivers* para interactuar con los distintos VIM y un VNFM genérico, y los flujos de trabajo son similares a los de la implementación de OSM.

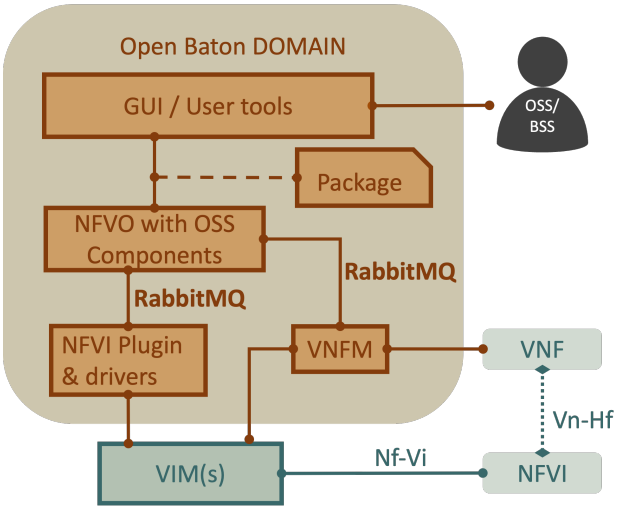

Figura 75: Arquitectura *Open Baton*

UNIVERSIDAD DE MÁLAGA

En [168], identificamos los criterios a evaluar en la comparación de cualquier orquestador de red con el estándar del MANO de la ETSI, proponiendo también la evaluación de un tercer orquestador en el marco del estándar, en este caso GTS.

GTS es un servicio creado en 2013 para ofrecer a la comunidad investigadora *testbeds* virtuales para experimentación que se ejecutan sobre una infraestructura compartida y con los que los investigadores interactúan a través de APIs dedicadas. Así, se pueden definir recursos tanto de la propia infraestructura de GÉANT como de terceros, mediante un lenguaje específico llamado DSL [156][157][176]. Lo que hace de GTS una implementación candidata del MANO de acuerdo al estándar de la ETSI, es que una vez se define el *testbed*, el propio sistema automatiza la asignación de recursos, de manera que los experimentadores reciben el control del *slice* que se ejecuta de manera aislada en una infraestructura real compartida con otros *testbeds*. Debido a que su implementación es anterior a la definición del estándar MANO de la ETSI, y a que su foco principal es la comunidad académica e investigadora, no se considera GTS como un MANO en sí.

Tras un análisis del cumplimiento de los requisitos NFV por parte del servicio GTS, se concluye que sí es una opción viable y se continúa con el estudio de su arquitectura y la comparación con la arquitectura de referencia del estándar.

GTS se compone de dos elementos principales: el CSF, que es responsable de la gestión y operación del servicio; y los puntos de distribución o PoDs, que se localizan a lo largo de Europa e incluyen los recursos *hardware* disponibles para ser virtualizados y usados por los experimentadores. El servicio sigue un modelo de virtualización generalizado (GVM) para controlar el ciclo de vida de los recursos y abstraer los procesos de asignación y provisionamiento internos mediante el uso de agentes de control de los recursos (RCA). El mapeo de los componentes del servicio con la arquitectura de referencia se muestra en la Figura 76.

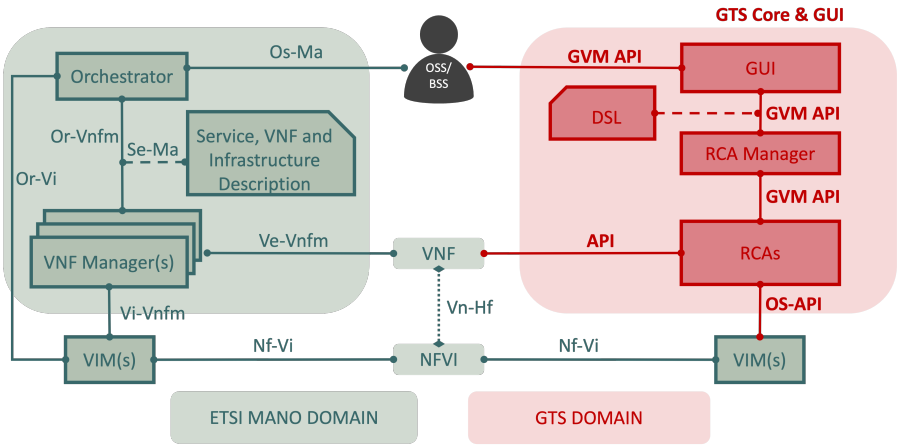

Figura 76: Mapeo de la arquitectura GTS-ETSI MANO [168]

UNIVERSIDAD DE MÁLAGA

En el caso de GTS, el equivalente al NFVI con los recursos del sistema es el conjunto de PoDs desplegados, teniendo en cuenta que en cada uno de esos PoDs se incluye el tejido de red para proporcionar circuitos virtuales y los nodos de cómputo para dar servicio a las máquinas virtuales. Como GTS se despliega sobre una plataforma *OpenStack*, el VIM en este caso es la propia plataforma *OpenStack*. Respecto a las VNF, GTS incluye una gran variedad de recursos virtualizados disponibles para la experimentación en su catálogo, que además se puede extender, tal y como propusimos en [134], para desarrollar nuevos RCA que gestionen recursos móviles de las redes 5G. Los puntos de referencia entre la infraestructura GTS y el despliegue *OpenStack* son idénticos al estándar NFV, mientras que el GVM proporciona a través de las API la comunicación entre las entidades del servicio.

El objetivo de esta comparación es el bloque funcional del MANO, que incluye los descriptores, el orquestador y el VNFM, tal y como muestra la Figura 76. En el sistema GTS, el componente CSF incluye el núcleo que se encarga de la orquestación, los RCA para controlar los recursos y la interfaz gráfica o GUI para acceder al servicio. Mapeando estos componentes a la arquitectura de referencia, el investigador desplegando un *testbed* es a GTS lo que un proveedor de servicio u operador a una red móvil desplegando un *slice* de red. En el núcleo del sistema se encuentra el RCA Manager para controlar los RCA, y un componente de red que intercambia información con los *routers* del plano de datos de los PoDs. El equivalente a los descriptores en GTS es el DSL que utilizan los experimentadores para definir los *testbeds* en el GUI. Dicho DSL incluye las características específicas de los recursos, así como la topología de los escenarios, y esa descripción es la que utiliza el RCA Manager para crear los RCA necesarios, actuando como VNFM.

Teniendo en cuenta este mapeo, queda demostrado que GTS, o cualquier otra implementación de sistema de orquestación, se puede evaluar en el contexto de la referencia del MANO de la ETSI siempre y cuando cumpla con los requisitos de NFV establecidos por la ETSI y siga una arquitectura como la definida en el estándar.

Toda esta virtualización de funciones de red en forma de máquinas virtuales es el primer paso hacia un entorno completamente virtualizado. Sin embargo, cuando se sustituyen los elementos de red físicos por VNF que incluyen todas las funcionalidades del elemento original, se tiende a la creación de soluciones monolíticas y pesadas que, a pesar de ser virtuales, no están optimizadas y resultan costosas en términos de gestión y mantenimiento. Por esto, se considera que los mapeos 1:1 entre las funciones de red físicas y virtualizadas son ineficientes, y se plantea la evolución de las VNF hacia las funciones de red nativas a la nube, o CNF, que siguen un modelo más ligero de virtualización basado en contenedores.

UNIVERSIDAD DE MÁLAGA

Las CNF se diseñan e implementan para ejecutarse empaquetadas en contenedores, de manera que tienen acceso a los recursos y al sistema operativo del anfitrión. Cada contenedor es una unidad de aplicación *software* que incluye tanto el código como las dependencias para ejecutarse de manera independiente y poder ser transferido de manera fiable entre entornos y nubes sin perder funcionalidad, lo que agiliza los despliegues y el consumo de recursos [60].

La adopción del modelo basado en CNF implica por tanto no solo el encapsulado de funciones en contenedores, sino también el rediseño de dichas funciones y su distribución en unidades más pequeñas que actúan como bloques de montaje desacoplados para mejorar la escalabilidad de los sistemas y disminuir su complejidad. El nivel de abstracción aumenta por tanto con la virtualización en contenedores, que abandonan las funciones monolíticas y permiten dinamizar los sistemas mediante el uso de recursos lógicos que comparten el sistema operativo y proporcionan mayor eficiencia en la utilización de los mismos.

En este nuevo paradigma, una aplicación o servicio se compone de diversos contenedores cuya gestión, despliegue y escalado son controlados por un orquestador de contenedores que asume el papel del ETSI MANO en entornos nativos de la nube. *Kubernetes*, también conocido como K8s, es el proyecto de código abierto impulsado por *Google* que se ha convertido en el estándar *de facto* para la orquestación de contenedores.

La ETSI presentó en [61] la adaptación de la referencia NFV a los principios nativos a la nube. Sin embargo, este mapeo de la arquitectura resultó demasiado genérico, además de limitarse al VIM y VNFM, por lo que *Amazon Web Services* presentó en [13] el mapeo de las plataformas de contenedores a entornos gestionados por K8s.

Resumiendo la arquitectura de K8s, una plataforma equivale a un cluster compuesto por un conjunto de nodos, que son las máquinas trabajadoras, así como un plano de control. Los nodos trabajadores alojan los *Pods*, que son la unidad de cómputo desplegable más pequeña en el ecosistema K8s. Los *Pods* corresponden a los microservicios containerizados, e incluyen uno o más contenedores. El plano de control gestiona los *Pods* y nodos, tomando decisiones globales que aplican a todo el cluster. Para exponer las aplicaciones que corren en un conjunto de *Pods*, se definen los objetos de tipo servicio, que actúan como una capa de abstracción que define las políticas de acceso a dicha aplicación. Al igual que los *Pods*, estos servicios se controlan mediante primitivas a través de la API. Contrario al ETSI MANO, K8s define las aplicaciones sin importar su función, sino como despliegues, de manera que esta definición incluye los objetos que componen la aplicación y K8s se encarga de la interacción y del correcto funcionamiento de esos objetos.

UNIVERSIDAD DE MÁLAGA

La principal diferencia entre la referencia del ETSI MANO y K8s reside en la exposición hacia niveles superiores, ya que el VNFM mantiene una visión detallada de sus VNF y las expone al NFVO, mientras que K8s es opaco con sus procesos internos y las operaciones son controladas simplemente a través de definiciones de los objetos y primitivas. De esta forma, en lugar de utilizar como el ETSI MANO autorizaciones para la gestión del ciclo de vida de los objetos, como es el caso en el que el VNFM solicita permisos al NFVO, en entornos K8s el NFVO solo especifica el estado operativo final deseado, y son el planificador y los constructos los encargados de garantizar la operación dentro de ese rango deseado, localizando los *Pods*, escalándolos y realizando una gestión eficiente de los recursos bajo demanda.

A.3 ESTADO DEL ARTE DE LOS SISTEMAS DE ORQUESTACIÓN

Una característica común que se observa en las plataformas de experimentación existentes y motiva esta tesis es la centralización de la orquestación de su arquitectura, lo que supone diversas limitaciones en términos de escalabilidad, automatización y resiliencia de la red. Distribuyendo la orquestación se puede optimizar el tráfico de la red y aligerar la carga del orquestador y de los propios enlaces, además de evitar que los sistemas tengan un punto único de fallo.

Sin embargo, distribuir la orquestación también aumenta la complejidad de los sistemas, así como el tráfico inherente a las propias tareas de orquestación. A esto se suma en el caso de las redes móviles la dificultad de virtualizar y distribuir entre los diferentes *slices* recursos celulares.

Así, se identifican distintos retos asociados a la orquestación, clasificados como retos consecuencia de la centralización del sistema, de la distribución del mismo, o relacionados con la naturaleza celular de los recursos siendo orquestados.

A.3.1 *Retos de la orquestación centralizada*

Los sistemas de orquestación centralizados presentan distintos retos, especialmente cuando la red siendo orquestada es distribuída, como suele ser el caso de las redes móviles que se extienden a lo largo de amplios territorios. Los retos principales identificados son:

- **Escalabilidad:** La escalabilidad de la red es fundamental para asegurar su continuidad cuando la carga del sistema es cambiante temporal y espacialmente. Como solución, distintas investigaciones se centran en dinamizar el *slicing* de red y localización de los recursos, como [18] y [37]. Sin embargo, la estrategia más común en la literatura se inclina hacia la distribución como solución a los problemas de escalabilidad [41][166][42][167].

UNIVERSIDAD DE MÁLAGA

- **Automatización:** La automatización en el proceso de creación de *slices* es crucial para disminuir la interacción humana en entornos donde el número de usuarios y dispositivos crece continuamente, aumentando consigo la complejidad de la red. Para abordar esta automatización, las propuestas se basan principalmente en la implementación de modelos basados en datos para la orquestación inteligente [178] [163] [44]; *machine learning* [187] [27] [153]; o la distribución de la orquestación para minimizar la complejidad [123] [160].

- **Resiliencia:** La resiliencia es esencial en el despliegue de sistemas con alta disponibilidad y fiabilidad, como las redes 5G, donde el fallo de una función de red virtualizada puede impactar a la operación de la red al completo. Así, se presentan distintos mecanismos, como los de restauración en [162], y [14] y [15], que no buscan evitar el fallo, sino predecirlo y restaurar el servicio con el menor impacto posible; la orquestación federada y cooperativa en [46] y [165]; o los algoritmos para mejorar la supervivencia de la red en [102] y [170].

A.3.2 *Retos de la orquestación distribuida*

Por otra parte, si bien distribuir la orquestación supone diversas ventajas en comparación con los sistemas centralizados, también implica múltiples retos a abordar:

- **Asignación de recursos:** Se necesitan mecanismos ágiles y dinámicos para realizar la asignación de recursos e instanciación de *slices* de red, para lo que se aplica teoría de gamificación [56]; predicción de tráfico de red celular [149]; formulaciones basadas en ILP [66][54][143][67]; VNE [31]; modelos basados en subasta [87][105][171]; y usando un modelo de sistema basado en colas [17].

- **Creación y gestión dinámica de *slices*:** La creación de *slices* incluye la optimización de la asignación de recursos de cara a maximizar la cantidad de servicios distintos que la red puede alojar. Así, algunos planteamientos se basan en los acuerdos de servicio [21]; en la creación de protocolos para orquestar los recursos dinámicamente a través de distintos dominios [109]; en los propios datos [112]; en la gestión de infraestructuras de la nube [88]; en TOSCA [38] [173]; o en la creación de un gestor de tareas que actúa como agente maestro [94].

- **Compartición de recursos:** La compartición de recursos dinámica permite optimizar el uso de la red. Sin embargo, se necesita un mecanismo de planificación para asignar correctamente los recursos entre los *slices* que los van a compartir, lo que

UNIVERSIDAD DE MÁLAGA

generalmente dificulta otros retos como la seguridad y el aislamiento. Además, en las redes móviles los recursos a compartir no son solo computacionales, sino también radio. Existen distintos mecanismos para abordar este problema, que se centran específicamente en la compartición radio, como es el caso del NVS [95][110][80][104]; mientras que otros pretenden actuar como un mediador de *slices* de red para servicios 5G, como los basados en *blockchain*[121] o el propuesto por Samdanis et al. [140][75][39]; y otros centrados en las plataformas de computación en la nube colaborativas [151].

- **Seguridad en escenarios distribuidos:** En los despliegues multidominio los riesgos de seguridad son aún más complejos, por lo que se necesitan mecanismos de seguridad y coordinación entre dominios que no se planteaban en arquitecturas de generaciones previas [25]. Los escenarios distribuidos se pueden abordar de manera similar a como se hace en infraestructuras multi-nube, tal y como se propone en SafeLib [114], con SECaaS en entornos multi-nube [92] y con modelos de dominios de confianza [130]. De las generaciones previas, se toman como ejemplo escenarios de *roaming* [108].

A.3.3 *Retos de la orquestación de redes móviles*

Parte de los retos de la orquestación vienen heredados de la naturaleza celular de la red, por lo que son comunes a los sistemas de orquestación, independientemente de su grado de distribución [103]:

- **Virtualización de recursos celulares:** La mayor parte de la investigación en la virtualización de redes se centra en el núcleo de la red. Sin embargo, los enlaces celulares varían con el tiempo y pueden sufrir interferencias, lo que evita que se puedan virtualizar con los mismos métodos que se aplican sobre los recursos cableados. Existen múltiples propuestas para proporcionar la abstracción de los recursos celulares, como la implementación y diseño del NVS [95][110][80][104]; el protocolo FlexRAN basado en OAI [68][99]; el plano de control SoftRAN [79]; la arquitectura OpenRAN [91]; la iniciativa xRAN [189]; la arquitectura CMaaS [179]; y la propuesta de C-RAN [174].
- **Aislamiento:** El aislamiento en el contexto del *slicing* de red es clave para la seguridad, dado que en caso contrario cualquier ataque o fallo que afecte a un *slice* podría impactar sobre la red al completo. En el caso de las redes móviles, hay que tener también en cuenta el aislamiento de los recursos RAN. Así, entre las soluciones para abordar este problema, se pueden distinguir las orientadas a aislar recursos RAN [120]; el uso de una infraestructura privada [148]; el aislamiento *hardware* [180]; protocolos para

UNIVERSIDAD DE MÁLAGA

securizar el aislamiento [142]; aislamiento centrado en plano de control [30]; y métodos para realizar *slicing* de red que directamente proporcionan aislamiento [90], que cuantifican el nivel de aislamiento alcanzado [96] y proporcionan aislamiento mediante la disminución de la utilización ineficiente de recursos que normalmente implica [183].

- **Gestión de la movilidad:** La industria del automóvil es una de las verticales principales de las redes 5G y posteriores, lo que supone una necesidad de soporte a la movilidad y la gestión de la misma. Además, los servicios en tiempo real requieren un traspaso móvil rápido e inapreciable. Estos retos se han abordado desde un punto de vista centralizado [150] [124], y desde uno distribuido [76] [19] [155]. Adicionalmente, existe la investigación centrada en las redes vehiculares y de computación en la niebla para atajar el problema del soporte a la movilidad [84] [185] [52].

- **Seguridad en el *slicing* de red:** El *slicing* de red es crucial en las redes 5G y posteriores, por lo que la seguridad en la orquestación del *slicing* es una de las preocupaciones principales, independientemente de la distribución de los sistemas. Entre la literatura disponible en seguridad en el *slicing* de red, cabe destacar el trabajo relacionado con securizar la capa física [177]; el aislamiento en los *slices* [144][36]; la comunicación *intra-slice* [142]; y el *micro-slicing* [36][128][35].

A.4 ARQUITECTURA DE REDES MÓVILES PARA SERVICIOS DISTRIBUIDOS

En esta sección se presentan los principios de la arquitectura definida en esta tesis para proporcionar *slices* de red, entendidos los *slices* como redes privadas que se pueden crear de manera dinámica con distintos niveles de configuración y control. La infraestructura resultante se compone de un conjunto de Puntos de Presencia o PoP distribuidos, que se encargan de gestionar de manera transparente recursos locales y remotos.

Esta infraestructura se diseñó y desarrolló en el contexto del proyecto EuWireless con la intención de proporcionar un operador virtual para la investigación en redes en Europa que integre nuevas tecnologías con la investigación en redes actuales.

En este diseño, los puntos de presencia o PoP son el componente principal o núcleo de la infraestructura. Cada PoP incluye el conjunto de *hardware* y *software* necesario para configurar y gestionar un *slice* de red, y puede funcionar como un nodo único o como parte de un conjunto de PoPs si el *slice* está distribuido geográficamente.

UNIVERSIDAD DE MÁLAGA

Así, cada PoP sigue la misma arquitectura de capas, cuya arquitectura a alto nivel es la siguiente, tal y como muestra la Figura 77:

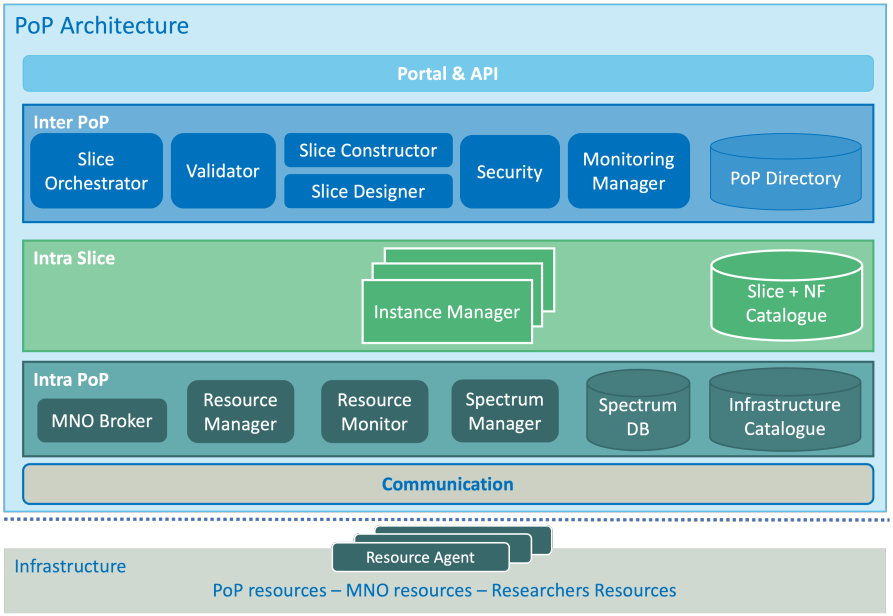

Figura 77: Architectura del Punto de Presencia (PoP)

- Portal & API: La infraestructura está diseñada para ser accesible a través de un portal web o una API donde se pueda diseñar *slices* mediante la personalización o extensión de alguna plantilla de *slice* disponible. Las plantillas son *slices* genéricos que incluyen los atributos configurables de cada uno de los recursos, de manera que pueden surgir distintas subplantillas mediante el establecimiento de ciertos rangos de valores para esos parámetros. Para que cada *slice* se adapte mejor a cada caso de uso específico, esos valores se pueden modificar y afinar.

- Inter PoP: Esta capa dispone de una visión global de la infraestructura y se encarga de las tareas que involucran distintos PoPs y *slices*. Para ello, almacena información del resto de PoPs que componen la infraestructura, incluyendo sus recursos locales y los usuarios autorizados a usarlos, pero excluyendo la disponibilidad de los recursos.

- Intra *Slice*: Esta capa se ocupa de gestionar los *slices* que han sido previamente diseñados y desplegados, y que se pueden extender a través de uno o más PoPs. Así, almacena el mapeado de la descripción abstracta del *slice* a los recursos para acelerar la construcción del *slice* cuando se reutiliza alguna plantilla o descripción de *slice*.

- Intra PoP: Esta capa interactúa con los recursos locales del PoP, que pueden tener distinto dominio administrativo, ya que pue-

UNIVERSIDAD DE MÁLAGA

den ser propiedad de la infraestructura, del operador comercial que comparte sus recursos con la infraestructura, o de cualquier usuario de la misma que quiera incorporar un recurso propietario en un *slice*. Esta capa sí incluye la información de la disponibilidad de los recursos locales, así como de los componentes de compartición del espectro que interactúan con el repositorio LSA externo.

- Comunicación: Esta capa proporciona los servicios de conectividad a las capas superiores para que se comuniquen con otros PoPs estableciendo conexiones lógicas extremo a extremo, con los recursos locales de la infraestructura, con las plataformas de experimentación y recursos de los investigadores y con la infraestructura de los operadores comerciales. Además, abstrae la topología de la red e integra los protocolos de bajo nivel requeridos para comunicarse con los operadores, los investigadores, y otros PoPs.

Tras la definición de un *slice*, los recursos incluídos deben mapearse primeramente a los recursos de la infraestructura, para después ser reservados. La Figura 78 muestra este proceso, donde se observa en la parte superior la definición del *slice* como un conjunto de recursos virtuales, ya sean funciones de red, máquinas virtuales o enlaces; la parte central muestra las entidades del PoP que gestionan los recursos de la infraestructura a los que se mapean esos recursos virtuales; y en la parte inferior se presenta el *slice* desplegado en términos de recursos. En la imagen se puede ver que los recursos pueden estar específicamente localizados en un área geográfica determinada, como es el caso de los UE, eNB, y SGW en la ubicación #1, mientras que el resto de funciones del EPC y servicios externos se distribuyen en localizaciones indiferentes al experimento.

Para gestionar los recursos, la capa Intra PoP incluye dos entidades específicas, el Gestor de Recursos y los Agentes de Recursos o RAs. Cada RA se encarga de la virtualización y control del ciclo de vida de un recurso concreto, de forma que basa la gestión de cualquier recursos en seis primitivas: Reservar, para asignar el recurso físico y crear la instancia; Activar, para configurar la instancia y ponerla en servicio; Reconfigurar, para modificar la configuración de algún parámetro o atributo tras la primera configuración; Consultar, para obtener información del estado del recurso; Desactivar, para parar la instancia manteniendo reservado el recurso; y Liberar, para eliminar la instancia y devolver el recurso a la lista de disponibles. Este ciclo de vida se muestra en la Figura 79, y permite realizar la abstracción de cualquier tipo de recurso mediante la implementación de las primitivas.

Como consecuencia de esta abstracción, la infraestructura es lo suficientemente flexible para integrar recursos nuevos, de manera que

UNIVERSIDAD DE MÁLAGA

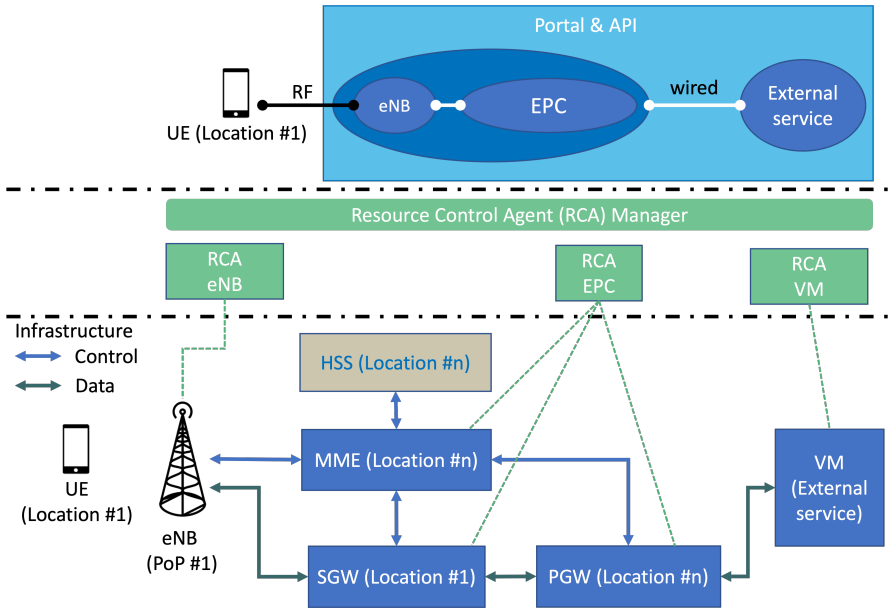

Figura 78: Diseño y mapeado de un *slice*

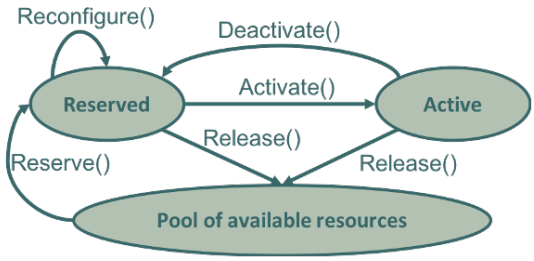

Figura 79: Máquina de estados del ciclo de vida de los recursos

se puede extender para adoptar nuevas tecnologías y tipos de recursos. La tecnología que inspira este diseño es el servicio GTS, creado por GÉANT en 2013 para que los investigadores fuesen capaces de crear redes de computación virtuales para experimentar en una infraestructura compartida.

A.5 EVALUACIÓN DE LA PROPUESTA

Para evaluar la arquitectura propuesta, se diseñan, desarrollan y despliegan tres plataformas de experimentación en redes móviles, basadas en distintas técnicas de virtualización. Concretamente, la plataforma EuWireless, que se orienta a la virtualización de las funciones de red como máquinas virtuales o VNFs orquestadas por un sistema distribuido; una primera aproximación a la plataforma 5G-EPICENTRE con una orquestación monolítica centralizada pero abstrayendo los componentes de la red en forma de contenedores o CNFs; y la versión final de 5G-EPICENTRE, que combina el despliegue de CNFs con un sistema de orquestación distribuido.

UNIVERSIDAD DE MÁLAGA

A.5.1 *Testbed EuWireless*

El planteamiento de la plataforma EuWireless, tal y como muestra la Figura 80, consiste en una arquitectura descentralizada y distribuida entre los PoPs para proporcionar la mayor cobertura posible y asegurar la escalabilidad de la plataforma. Estos PoPs tienen capacidad para conectarse con otras plataformas de experimentación disponibles, así como con otros PoPs para extender su cobertura y recursos disponibles para componer un *slice* de red para experimentación.

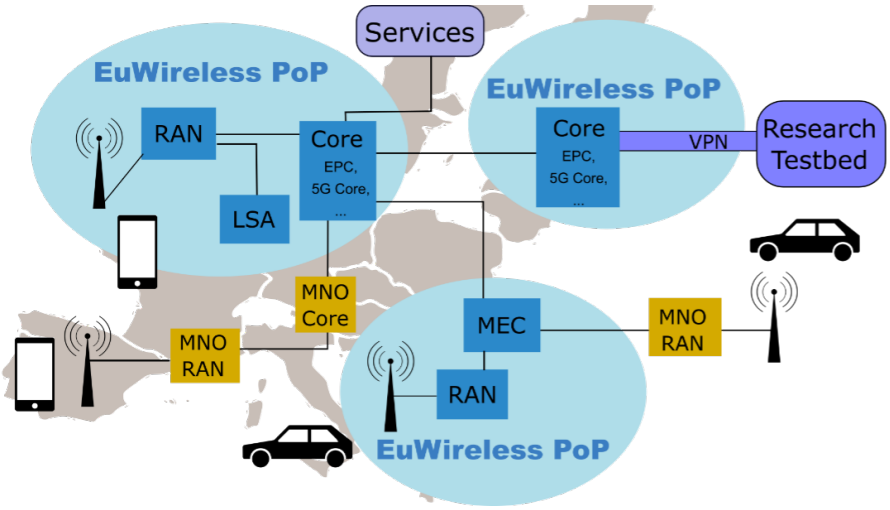

Figura 80: Visión de alto nivel de la plataforma EuWireless [115]

Para la implementación de este testbed, se combinan la tecnología de redes definidas por *software* o SDN con la herramienta GTS de GÉANT. GTS sigue una arquitectura con un modelo generalizado de virtualización que proporciona una abstracción de la infraestructura tal que los recursos resultan opacos para los usuarios. De esta forma, cada recurso se distingue por una serie de atributos y puertos que el experimentador especifica al definir el recurso. Los puertos se utilizan para definir la topología de la red, conectándose con los puertos del resto de recursos. Así, definir un *slice* de red se reduce a especificar el conjunto de recursos que lo componen y la adyacencia entre sus puertos. La capa de coordinación de EuWireless, por tanto, extiende el entorno GTS para incluir componentes de redes celulares mediante la definición e implementación de sus atributos y primitivas para controlar su ciclo de vida.

Como prueba de concepto, se realiza el despliegue de un PoP en las instalaciones de la Universidad de Málaga sobre el que se instancias distintos *slices* de red para demostrar que se cumplen los distintos objetivos de diseño: soporte a *slices* concurrentes y aislados, acceso remoto garantizado para los usuarios de la plataforma, proceso automatizado para suministrar *slices*, y la posibilidad de integrar distintas tecnologías y dominios celulares.

UNIVERSIDAD DE MÁLAGA

A.5.2 *Testbed 5G-EPICENTRE Málaga (fase inicial)*

La segunda plataforma de experimentación desarrollada es el prototipo del testbed 5G-EPICENTRE, orientado a la experimentación en redes 5G centradas en dar soporte a servicios de misión crítica. En este caso, las funciones de red se virtualizan como contenedores, de manera que se sigue un enfoque nativo de la nube con una arquitectura orientada a servicios que facilita los desarrollos y despliegues, manteniendo su compatibilidad con una amplia variedad de tecnologías y soluciones para las distintas verticales.

La implementación en este caso toma como punto de partida la arquitectura de referencia del estándar de la ETSI, añadiendo nuevas entidades orientadas a dar soporte a las funciones contenerizadas. Como sistema orquestador en esta implementación, se despliega una plataforma *Kubernetes*, ya que se trata del orquestador *de facto* en entornos de contenedores, con un único máster y dos nodos trabajadores. El clúster de *Kubernetes*, tal y como muestra Figura 81, conecta con las redes externas y con un núcleo de red 5G que conecta a su vez con los recursos radio de la plataforma de experimentación 5GENESIS, proporcionando así los recursos necesarios para desplegar servicios 5G como contenedores y realizar experimentos en escenarios realistas.

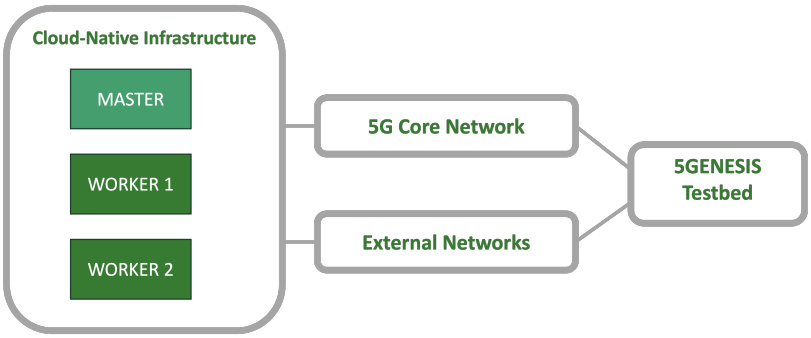

Figura 81: Arquitectura basada en *Kubernetes* de la plataforma de Málaga

La prueba de concepto en este caso, dado que se trata de una primera aproximación a lo que será la plataforma de 5G-EPICENTRE, se centra simplemente en demostrar la viabilidad del enfoque de contenedores en el contexto de las redes móviles, y más concretamente, de la vertical 5G de servicios críticos. Así, se realiza sobre el cluster *Kubernetes* el despliegue en tiempo real de una solución de vídeo portátil para aplicaciones de telemedicina, probando así la viabilidad de la infraestructura diseñada.

UNIVERSIDAD DE MÁLAGA

A.5.3 *Testbed 5G-EPICENTRE*

Probada la viabilidad de aplicar los conceptos de la tecnología de contenedores en el contexto de las redes móviles, la implementación definitiva de la plataforma de Málaga del testbed 5G-EPICENTRE incluye un sistema de orquestación distribuido, de manera que la distribución del sistema no se limite a los servicios desplegados.

En el estudio y diseño de esta plataforma, se concluye que para crear una red robusta y controlable para proporcionar recursos redundantes y de alta disponibilidad, las funciones de red que la componen deben dividirse en entidades más pequeñas, de manera que al aumentar su granularidad aumente también su adaptabilidad a los distintos escenarios. Estas entidades reciben el nombre de microservicios y normalmente se encapsulan en contenedores, combinándose en cadenas para componer las funciones de red al completo.

En esta plataforma, de nuevo se recurre a *Kubernetes* como sistema de orquestación, ya que simplifica la gestión de sistemas dinámicos a gran escala y proporciona una plataforma sobre la que desplegar sistemas distribuidos, escalables, resilientes y con tolerante frente a fallos. A diferencia de la plataforma anterior, en este caso el despliegue de *Kubernetes* es distribuido en una arquitectura multi-master multi-nodo, tal y como se observa en Figura 82, que proporciona una alta disponibilidad y fiabilidad características de los sistemas de misión crítica, mientras procesa la alta demanda de peticiones típica de las aplicaciones y tecnologías 5G.

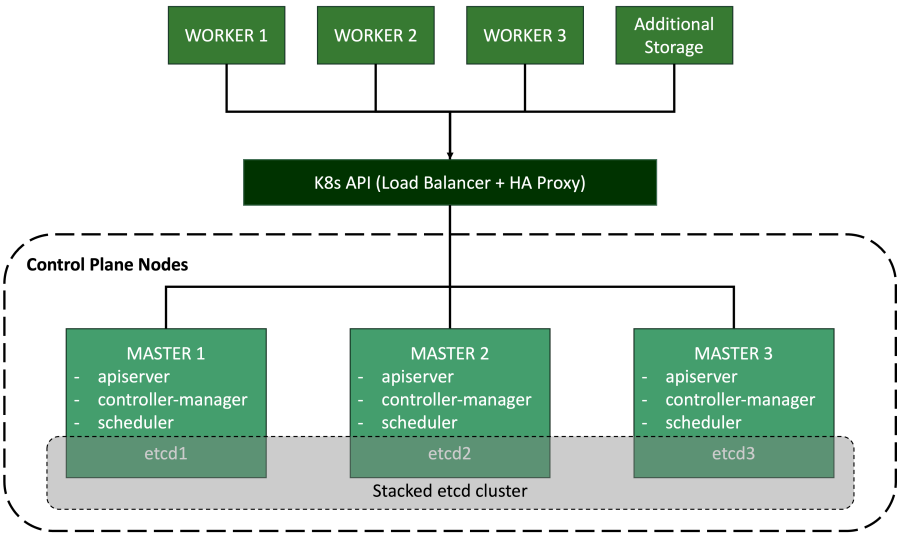

Figura 82: Infraestructura distribuida desplegada basada en *Kubernetes*

Como prueba de concepto para esta plataforma, se realiza el despliegue de un sistema de comunicaciones críticas 5G llamado *Mobitrust* que incluye un equipamiento de usuario móvil que se encarga de recoger y monitorizar datos del entorno a través de distintos sen-

UNIVERSIDAD DE MÁLAGA

sores y dispositivos para transmitirlos a un centro de control que se encarga de procesarlos. En la Figura 83 se muestra la descomposición de la aplicación *Mobitrust* en microservicios desplegados sobre el cluster *Kubernetes* de la plataforma.

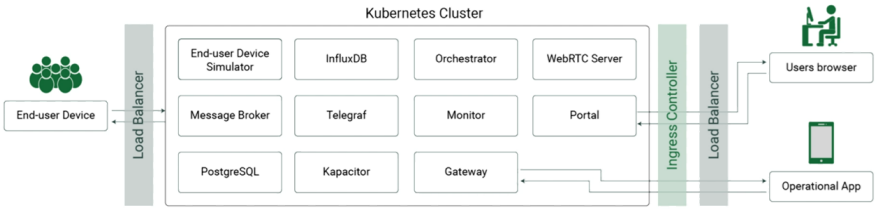

Figura 83: Aplicación Mobitrust desplegada

Los resultados de este experimento, basados en el estudio de KPIs característicos de las aplicaciones 5G de misión crítica, demuestran las ventajas de usar un sistema de orquestación distribuido para gestionar funciones basadas en tecnologías nativas de la nube y desplegar aplicaciones de este tipo.

A.6 AUMENTO DE LA DISTRIBUCIÓN MEDIANTE MICROSERVICIOS

El uso de soluciones de virtualización y componentes *software* en la definición de las redes 5G y posteriores ha sido motivado por la evolución de los servicios para adaptarse a nuevos tipos de infraestructuras y sistemas, ya que permiten mediante la automatización y la programación montar y configurar redes orientadas a casos de uso específicos. De igual forma, el *slicing* de red permite que una única infraestructura aloje múltiples redes, cada una con su configuración y requisitos específicos asociados a la vertical a la que dan servicio, lo que a su vez implica la necesidad de procedimientos de reconfiguración bajo demanda que resulten eficientes para los operadores de red. Todo esto es posible gracias a la adopción del paradigma NFV y de la descomposición de las VNF en microservicios.

En este contexto, las soluciones nativas a la nube proporcionan la abstracción de la infraestructura requerida para adaptarse fácilmente a nuevas tecnologías manteniendo la compatibilidad con las soluciones heredadas, mientras que la containerización de servicios permite realizar una virtualización ligera para la creación y el encadenado de microservicios, de manera que combinadas permiten manejar más fácilmente estos servicios distribuidos y mejorar el rendimiento de las redes. En conclusión, distribuir los servicios incrementando su granularidad se plantea como una alternativa a la mera distribución de los sistemas de orquestación, que también puede combinarse con ésta para aunar los beneficios de los esquemas distribuidos.

UNIVERSIDAD DE MÁLAGA

El concepto de aplicación de red o *NetApp* se define como una cadena de VNFs orientada a abordar los requisitos específicos de una vertical determinada de la red 5G [63]. Inicialmente, se plantearon como funciones de red que cumplían con la especificación NFV de la ETSI, aunque es posible combinarlas para desplegar servicios de red complejos donde el tráfico transita entre las funciones como si se se tratara de una cadena. Así, las *NetApp*s ofrecen una capa de abstracción para permitir a las aplicaciones de las verticales consumir recursos de las distintas entidades que componen la arquitectura de la red 5G.

La plataforma de experimentación del proyecto 5G-EPICENTRE permite desplegar desarrollos basados en el modelo *NetApp* orientados a la vertical PPDR de las redes 5G. La infraestructura, a alto nivel, incluye cuatro plataformas de experimentación desplegadas en cuatro instalaciones diferentes e independientes, con una arquitectura de gestión basada en *Kubernetes* para orquestar las aplicaciones en forma de contenedor, y con una capa de federación para el plano de control basada en *Karmada*, tal y como muestra la Figura 84.

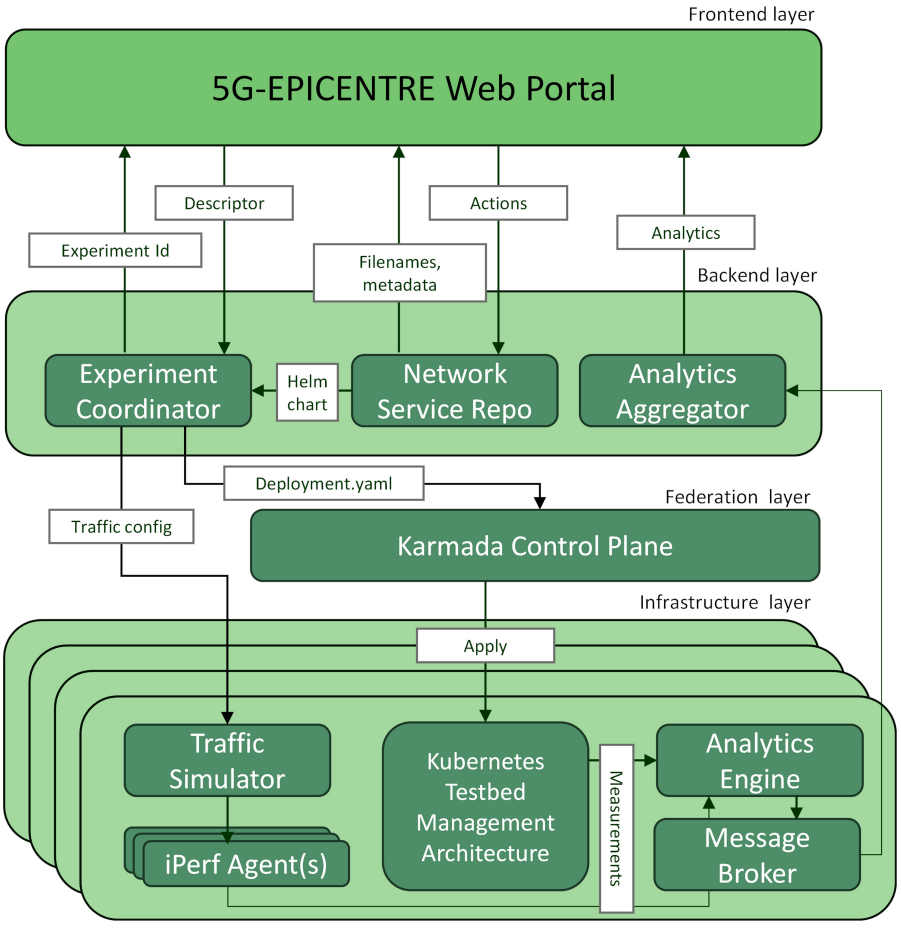

Figura 84: Estructura en capas de alto nivel de la plataforma 5G-EPICENTRE [23]

UNIVERSIDAD DE MÁLAGA

Los prototipos de esta plataforma evolucionan desde un ecosistema basado puramente en *OpenStack* que sigue la referencia del MANO de la ETSI hacia la inclusión de *Kubernetes* como el VNFM. De esta forma, la plataforma 5G-EPICENTRE recurre a las tecnologías de virtualización basadas en contenedores para proporcionar seguridad y robustez en el despliegue de servicios, así como alta disponibilidad de dichos servicios para cumplir con los requisitos de la vertical PPDR, adaptando este modelo basado en NetApps.

La evolución de las infraestructuras 5G existentes hacia soluciones orientadas a los microservicios queda reflejada en esta adopción del modelo de NetApps. En este contexto, la plataforma de 5G-EPICENTRE aborda la fragmentación de recursos a través de la federación de distintas plataformas de experimentación basadas en *Kubernetes* mediante el despliegue de contenedores en un entorno multicluster, que es posible mediante la combinación de las herramientas de federación con la gestión de *Kubernetes* de *Karmada*, que permiten el despliegue y la ejecución de NetApps de manera concurrente.

Como resultado, la plataforma 5G orientada a las NetApps es capaz de integrar tanto la distribución de los servicios como de la orquestación, con el objetivo de aumentar la granularidad y la distribución de la arquitectura al completo, mejorando así la propuesta original de la arquitectura de esta tesis, ya que esta granularidad en los servicios facilita la adaptabilidad de la red a entornos y requisitos que son constantemente cambiantes.

Un entorno multi-nube combinado con una solución MANO multiplataforma refleja esta distribución de servicios y orquestación, permitiendo la definición y ejecución de microservicios compuestos por NetApps y diseminados a través de distintos PoPs. El MANO multiplataforma interactúa con los clusters desplegados en cada una de las plataformas 5G individuales a través del API de *Kubernetes*, mientras que *Karmada* proporciona un modelo de información único que encapsula los aspectos diferenciados del API de cada plataforma, de manera que los propietarios de las plataformas mantengan el control de sus recursos 5G y haya una adaptación generalizada a las particularidades de cada cluster. La arquitectura de alto nivel de esta federación multiplataforma se muestra en la Figura 85, en la que se observa la posibilidad de desplegar cadenas de NetApps distribuidas en distintas plataformas desde una única interfaz.

Respecto a los experimentos realizados para demostrar las mejoras reales de aplicar este planteamiento basado en NetApps en los ecosistemas 5G y en los servicios de las verticales, se realizan los tres siguientes cumpliendo con el estándar 3GPP de aplicaciones MCX para la vertical PPDR:

Aplicación de comunicaciones de grupo en tiempo real: Demuestra que los *slices* de red parametrizados para comunicaciones de misión crítica no se ven afectados por la carga de tráfico

UNIVERSIDAD DE MÁLAGA

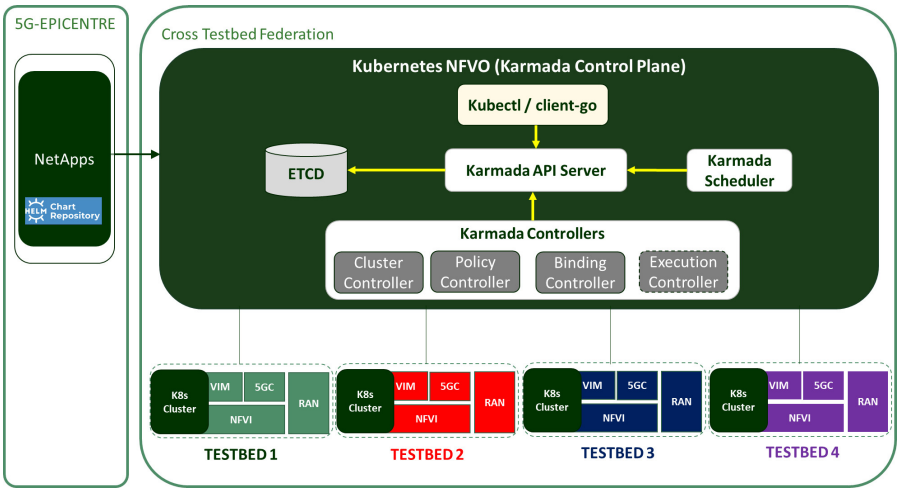

Figura 85: Arquitectura de alto nivel de la federación multiplataforma [23]

de la red subyacente, asegurando así que los requisitos de estas comunicaciones se cumplen incluso en las condiciones más exigentes. Este experimento también muestra cómo la adopción de soluciones ágiles de despliegue mejora significativamente el proceso de instanciación en comparación con otros modelos de despliegue, y cómo dos sistemas de la vertical MCX localizados en dos clusters separados se pueden comunicar entre ellos tan pronto como se despliegue la NetApp, lo que agiliza también la instanciación de servicios en el *Edge*.

Plataforma de conocimiento de la situación: Este experimento sitúa no solo a los sistemas IoT sino también al paradigma de computación en el *Edge* como factores clave en los despliegues PPDR, ya que permiten ubicar los microservicios lo más cerca posible de los usuarios finales para obtener el mayor conocimiento de las operaciones en campo, mejorando la calidad de las transmisiones y reduciendo la latencia de los sensores. Se demuestran también los beneficios de combinar el modelo de NetApp con las tecnologías 5G en escenarios con mecanismos de *slicing* dedicado implementados.

Solución de realidad aumentada: Sirve para validar la capacidad del modelo de NetApp para dar soporte a despliegues de aplicaciones de verticales con requisitos altamente exigentes en términos de ancho de banda y latencia estricta. Además, se pone en valor la posibilidad de conectar cualquier sistema de vertical con las funcionalidades 5G a través de las APIs de servicio de las NetApps, teniendo en cuenta que la mayoría de las aplicaciones se diseñaron para redes heredadas y sin considerar los despliegues en la nube. Mediante la implementación de sistemas de verticales, se muestra también la optimización de rendimiento alcanzada.

UNIVERSIDAD DE MÁLAGA

En conclusión, los resultados de los experimentos evidencian la viabilidad de la integración de las comunicaciones de misión crítica en las redes 5G siguiendo el modelo de NetApp y superando las limitaciones impuestas a estas comunicaciones por la banda estrecha.

A.7 CONCLUSIONES Y TRABAJO FUTURO

En esta tesis se presenta la creación de un entorno de experimentación para desplegar *slices* de red 5G con recursos distribuidos en múltiples localizaciones y controlados por un sistema de orquestación igualmente distribuido. La motivación principal tras este planteamiento es la limitación actual en el campo de la experimentación en redes celulares debida a las exigencias de los requisitos para plantear experimentos realistas.

Teniendo en cuenta la integración del paradigma del *cloud computing* y de la virtualización de redes en los diseños y arquitecturas de las redes móviles de última generación, esta tesis propone una solución para ofrecer redes móviles temporales para realizar experimentos que actúan como *slices* de red. Dicha solución incluye el diseño de una arquitectura basada en la implementación de *Puntos de Presencia* distribuidos, conectados y que incluyen una variedad heterogénea de recursos móviles.

Para evaluar la viabilidad y madurez de la propuesta, se definen, implementan y prueban tres opciones basadas en diferentes técnicas de virtualización y con un nivel distinto de distribución y granularidad de los servicios y su orquestación. La conclusión principal respecto a la evaluación de la viabilidad reside en la naturaleza de la distribución que se aplica. A pesar de la hipótesis inicial que plateaba distribuir la orquestación de los recursos, un análisis más profundo y la experimentación realizada demuestran que distribuir los servicios que se van a proporcionar aumentando su granularidad mejora la eficiencia de la infraestructura y agiliza los despliegues en tiempo real. Es más, la plataforma que presenta mejores resultados en términos de rendimiento es la última plataforma de experimentación desplegada para evaluar la propuesta, que se basa en un sistema de orquestación distribuido basado en *Kubernetes* para gestionar funciones de red nativas de la nube, distribuidas y de grano fino, combinando así la distribución tanto de los servicios como de la orquestación de los recursos.

Así, la última parte de la tesis estudia los beneficios de crear una infraestructura de experimentación distribuida y nativa de la nube que sea capaz de adaptar los servicios móviles 5G al modelo de las aplicaciones de red y los microservicios para mejorar la capacidad de la red, los mecanismos de *slicing*, y la gestión de picos de tráfico y usuarios en la misma.

UNIVERSIDAD DE MÁLAGA

Como líneas de trabajo futuras, durante la revisión de la literatura para realizar el estado del arte se identificaron múltiples retos asociados a la orquestación que aún están por resolver. Por otra parte, si bien se considera que desplegar una plataforma de experimentación como las propuestas a nivel regional, nacional o global podría tener un gran impacto en la investigación en redes móviles, es necesario implementar componentes de red 5G para extender el alcance de las plataformas. Se propone también continuar con la investigación del impacto de la interacción del modelo de las aplicaciones de red con las redes 5G. Por último, dado que el análisis de la viabilidad de las plataformas presentadas se centra en el sector de los servicios de comunicaciones críticas, se propone extender el análisis al resto de verticales 5G.

UNIVERSIDAD DE MÁLAGA