

UNIVERSIDAD POLITÉCNICA DE MADRID
Escuela Técnica Superior de Ingenieros de Telecomunicación



Contribución al diseño y despliegue de gemelos digitales de red

TESIS DOCTORAL

Presentada para optar al título de Doctor por:

Mario Sanz Rodrigo

Ingeniero de Sistemas de Comunicaciones

Madrid, 2024



UNIVERSIDAD POLITÉCNICA DE MADRID
Escuela Técnica Superior de Ingenieros de
Telecomunicación

Doctorado en Ingeniería de Sistemas Telemáticos

Contribución al diseño y despliegue de gemelos digitales de red

TESIS DOCTORAL

Presentada para optar al título de Doctor por:

Mario Sanz Rodrigo

Ingeniero de Sistemas de Comunicaciones

Bajo la dirección de:

Dr. José Ignacio Moreno Novella (Director)

Dr. Diego Rivera Pinto (Codirector)

Madrid, 2024

Título: Contribución al diseño y despliegue de gemelos digitales de red

Autor: Mario Sanz Rodrigo

Programa de Doctorado: Doctorado en Ingeniería de Sistemas Telemáticos

Dirección de tesis:

Dr. José Ignacio Moreno Novella, Catedrático de Universidad, UPM (Director)

Dr. Diego Rivera Pinto, Profesor Contratado Doctor, UPM (Codirector)

Tribunal de tesis:

Fecha de defensa:

A mis queridos Ana, Dana y Adrián,
Por todos los buenos momentos pasados, presentes y futuros.

Agradecimientos

A todas las personas que en algún momento han prestado parte de su tiempo para aguantar las divagaciones, frustraciones, logros o alegrías, sin importar si entendían o no lo que les estaba contando.

En particular, las gracias a mis directores de tesis, José Ignacio y Diego, que se prestaron desde el primer momento a recorrer este camino y ayudarme en todo lo posible. Atrás quedan los cafés en los que exponía los problemas o las ideas trasnochadas que han ido surgiendo durante todo este proceso, pero seguro que no serán los últimos.

Agradecer también a todo el grupo de investigación Redes y Servicios de Telecomunicación e Internet, es difícil realizar la asociación de lo que ha aportado cada uno de ellos, pero estoy seguro durante los casi ocho años que pertenezco al equipo, todos han ido aportando su granito de arena.

Especialmente gracias a mi familia, en primer lugar, a mis padres por darme la oportunidad, hace ya muchos años de empezar mis estudios universitarios, ellos saben que en esos tiempos no era mi primera opción. En segundo lugar, a mi mujer Ana, probablemente si no fuera por ella, no estaría hoy donde esto, ya que fue la que me insistió y motivó para comenzar la universidad. Finalmente, a mis hijos, Dana y Adrián, aunque ahora mismo sois muy pequeños, y no seréis conscientes del aporte que habéis hecho, muchos días sumabais al cansancio acumulado, pero otros días con vuestras sonrisas y risotadas me dabais la fuerza necesaria para afrontar los días malos y recordar que lo importante está en casa.

En definitiva, esto es solo un paso más del camino, y estaré encantado de que los que están sigan estando a mi lado.

Abstract

Although the concept of Digital Twins has been present in the literature for more than two decades, it is only in recent years, thanks to technological advancements, that its implementation in multidisciplinary fields such as Industry 4.0, cybersecurity, healthcare, and next-generation communication networks has become feasible. However, it is still far from being a consolidated and regulated paradigm, primarily due to the presence of numerous unaddressed challenges, such as security, deployment, management, and standardization.

This doctoral thesis focuses on the applicability of the digital twin paradigm to the specific context of communication networks, known as Network Digital Twins. This approach combines the characteristics and inherent requirements of the general paradigm with the specific challenges related to modeling communication networks. While technological advancements are facilitating the use and development of digital twin-based solutions, they are also reshaping the way networks are designed and deployed, thereby complicating the attainment of global solutions due to factors such as heterogeneity in network equipment or virtualization technologies.

Although there are currently various approaches of the paradigm applied in areas such as the Internet of Things, augmented reality, healthcare, or unmanned autonomous vehicles, these cannot be directly extrapolated to the realm of communication networks due to the high dependence of modeling on the specific application context. Therefore, the main objective of this doctoral thesis is to contribute to the development of a methodology enabling the creation and deployment of network digital twins, considering the limitations and inherent challenges within this field.

Through the analysis of the state of the art, the characteristics and requirements associated with Digital Twins (DT) are initially identified, followed by a review of the specific case of Network Digital Twins (NDT), focusing on the identification of unresolved needs and issues. As a result of this study, a methodology is proposed that seeks to standardize the creation and deployment of Network Digital Twins, as well as their integration with Physical Twins, through various phases.

After formally defining the methodology, the first phase leads to the second contribution, which focuses on the problem associated with acquiring the necessary data for modeling both the topology and the behavior of network devices. This issue stems from the heterogeneity of equipment and virtualization technologies, as well as the various network topologies. In response to this, the design and development of the DANA system (Digital Agent for Network Data Acquisition) is proposed, capable of collecting data

regardless of the type of end device or network in the physical twin. This system is a key component of the NDT ecosystem, present in both the physical twin environment and the digital twin environment after its creation. This allows, in subsequent phases, intervention on the twins to facilitate the joint coevolution of the ecosystem, a fundamental characteristic of the paradigm.

In relation to the need to carry out the implementation of the proposed methodology and based on the enabling technologies reviewed during the study of the state of the art, the design and development of a platform capable is performed, allowing both the creation and deployment of the complete NDT ecosystem, such as the implementation of a bidirectional communication channel capable of guaranteeing both data collection and information exchange, in order to enable the joint co-evolution of both environments.

Resumen

Pese a que el concepto de Gemelos Digitales lleva presente en la literatura desde hace más de dos décadas, es en los últimos años, cuando gracias a los avances tecnológicos se está pudiendo llevar a cabo su implementación en campos multidisciplinarios como la industria 4.0, ciberseguridad, salud o redes de comunicación de nueva generación. No obstante, aún está lejos de ser un paradigma consolidado y regulado, debido a la existencia de múltiples desafíos sin abordar, como la seguridad, el despliegue, la gestión o la estandarización.

Esta tesis doctoral se enfoca en la aplicabilidad del paradigma de gemelos digitales al contexto específico de las redes de comunicación, conocido como Network Digital Twins. Este enfoque combina las características y requisitos inherentes al paradigma general con los desafíos específicos relacionados con el modelado de las redes de comunicación. Si bien los avances tecnológicos están facilitando el uso y desarrollo de soluciones basadas en gemelos digitales, también están transformando la forma en que se diseñan y despliegan las redes, lo que dificulta la obtención de soluciones globales debido a factores como la heterogeneidad en el equipamiento o en tecnologías de virtualización de red.

Aunque actualmente existen diversas aproximaciones del paradigma aplicadas en áreas como el Internet de las Cosas, realidad aumentada, salud o vehículos autónomos no tripulados, estas no pueden extrapolarse directamente al ámbito de las redes de comunicación debido a la alta dependencia del modelado con el contexto específico de aplicación. Por tanto, el objetivo principal de esta Tesis Doctoral es contribuir al desarrollo de una metodología que permita la creación y despliegue de gemelos digitales de redes, considerando las limitaciones y desafíos inherentes a este campo.

A través del análisis del estado del arte se identifican inicialmente las características y requisitos asociados a los Gemelos Digitales (DT), seguido por una revisión del caso específico de Gemelos Digitales de Red (NDT), con un enfoque en la identificación de necesidades y problemáticas no resueltas. Como resultado de este estudio, se propone una metodología cuyo objetivo es tratar de estandarizar, a través de diversas fases, la creación y despliegue de los gemelos digitales de red, así como su integración con los gemelos físicos.

Tras definir formalmente la metodología, la primera fase conduce a la segunda contribución, centrada en la problemática asociada a la adquisición de los datos necesarios para el modelado tanto de la topología como del comportamiento de los equipos de la red. Esta problemática radica de la heterogeneidad de equipos y tecnologías de virtualización, así como las diversas topologías de red. En respuesta a esto, se propone el diseño y desarrollo del sistema DANA (Digital Agent for Network Data Acquisition),

capaz de recolectar datos independientemente del tipo de dispositivo final o red del gemelo físico. Este sistema es un componente clave del ecosistema NDT, presente tanto en el entorno del gemelo físico como en el del gemelo digital tras su creación. Esto permite, en fases posteriores, la actuación sobre los gemelos para facilitar la coevolución conjunta del ecosistema, una característica fundamental del paradigma.

En relación con la necesidad de llevar a cabo la implementación de la metodología propuesta y basado en las tecnologías habilitadoras revisadas durante el estudio del estado del arte, se realiza el diseño y desarrollo de una plataforma capaz de llevarla a cabo, permitiendo tanto la creación y despliegue del ecosistema NDT completo, como la implementación de un canal de comunicación bidireccional capaz de garantizar, tanto la recolección de datos como el intercambio de información, con el fin de posibilitar la coevolución conjunta de ambos entornos.

Tabla de Contenido

Capítulo 1 Introducción y objetivos	1
1.1. Contexto y motivación.....	2
1.2. Objetivos	6
1.3. Metodología de trabajo	8
1.4. Actividades desarrolladas en el marco de la presente tesis doctoral.....	8
1.4.1. Participación en Proyectos de investigación competitivos:.....	8
1.4.2. Publicaciones relacionadas con el trabajo de investigación:.....	12
1.4.3. Otras publicaciones:	12
1.4.4. Dirección de Trabajos de fin de titulación:.....	13
1.5. Estructura del documento.....	14
2. Capítulo 2 Estado del Arte	17
2.1. Gemelos Digitales. Conceptos, principios y habilitadores	19
2.1.1. Conceptos.....	22
2.1.2. Principios.....	25
2.1.3. Tecnologías habilitadoras.....	27
2.1.4. Trabajos relacionados.....	29
2.2. Gemelos Digitales de Red (NDT). Conceptos, principios y habilitadores.	31
2.2.1. Conceptos.....	32
2.2.2. Principios.....	33
2.2.3. Habilitadores tecnológicos.....	37
2.2.4. Trabajos relacionados.....	38
2.2.5. Retos.....	41
2.3. Tecnologías de soporte al NDT.....	42
2.3.1. Tecnología SDN.....	42
2.3.2. Tecnología NFV.....	44
2.3.3. Tecnologías de comunicación	58
2.3.4. Trabajos relacionados.....	62
2.4. Tecnologías de Redes Móviles Avanzadas.....	65
2.4.1. Conceptos.....	66
2.4.2. Arquitectura basada en servicios	68
2.4.3. Implementaciones 5G.....	70
2.4.4. Especificación Gemelo Digital en 6G.....	71
2.4.5. Trabajos relacionados.....	73
3. Capítulo 3 Metodología para el modelado y despliegue de gemelos digitales de red..	73
3.1. Diseño de la metodología.....	75
3.1.1. Fase I. Adquisición de datos	79
3.1.2. Fase II. Modelado DTs.....	96
3.1.3. Fase III. Adaptación.....	97
3.1.4. Fase IV. Despliegue NFV.....	108

3.1.5.	Fase V. Aprovechamiento del NDT	110
3.1.6.	Fase VI. Interconexión de gemelos	112
3.1.7.	Fase VII. Realimentación del sistema NDT	116
4.	Capítulo 4 Plataforma para el soporte del NDT.....	117
4.1.	Diseño de la plataforma.....	120
4.1.1.	Requisitos de la plataforma.....	120
4.1.2.	Diseño de la arquitectura	123
5.	Capítulo 5 Casos de estudio	137
5.1.	Caso I. Despliegue de NDT a través de red generada en GNS3.....	139
5.1.1.	Conclusiones.....	147
5.2.	Caso II. Despliegue NDT red 5G.....	149
5.2.1.	Conclusiones.....	162
5.3.	Caso III. Interconexión entre gemelos y securización del canal de comunicaciones	165
5.3.1.	Conclusiones.....	177
6.	Capítulo 6 Conclusiones y líneas futuras.....	181
6.1.	Análisis de objetivos.....	183
6.2.	Limitaciones	186
6.3.	Conclusiones	187
6.4.	Trabajos futuros.....	191
	Bibliografía	193

Lista de Figuras

Figura 1. Pilares gemelo digital de red	33
Figura 2. Arquitectura referencia NDT propuesta por el IETF	35
Figura 3. Capas de la arquitectura SDN	43
Figura 4. Arquitectura NFV propuesta por ETSI	46
Figura 5. Diagrama de interconexión de elementos dentro de la unidad NFV-MANO.....	47
Figura 6. Interacción entre componentes OSM.....	48
Figura 7. Arquitectura en detalle OSM	50
Figura 8. Diagrama VNF.....	52
Figura 9. Diagrama NS	53
Figura 10. Acciones primitivas OSM	58
Figura 11. Jerarquía de topics MQTT	59
Figura 12. Funcionamiento protocolo MQTT a través de Mosquitto MQTT.....	60
Figura 13. Arquitectura general Broker Mosquitto MQTT	60
Figura 14. Arquitectura general Apache Kafka.....	61
Figura 15. Funcionamiento básico RabbitMQ.....	62
Figura 16. Escenarios de uso en 5G.....	66
Figura 17. Arquitectura core 5G basada en funciones de red	68
Figura 18. Escenarios de utilización y aspectos generales IMT-2030.....	71
Figura 19. Metodología propuesta	75
Figura 20. Relación entre estados y metodología.....	79
Figura 21. Diagrama de bloques funcionamiento del agente DANA.....	81
Figura 22. Diagrama de flujo funcionamiento del sistema DANA	82
Figura 23. Diagrama de bloques funcionamiento agente (centralizado y descentralizado) ...	84
Figura 24. Ejemplo del mapa generado por el protocolo ALTO	85
Figura 25. Ejemplo del mapa de coste generado por el protocolo ALTO.....	86
Figura 26. Ejemplo de módulo de reporte agente satélite.....	87

Figura 27. Arquitectura del modelo de datos.....	90
Figura 28. Diagrama de bloques flujos de comunicación hacia el gestor	96
Figura 29. Modelado de la información recolectada	97
Figura 30. Adaptación del formato de datos a plantillas NFV	98
Figura 31. Ejemplo de fichero NSD y VNFD	101
Figura 32. Estructura de carpetas generada por el subsistema de adaptación	108
Figura 33. Interconexión de gemelos basada en flujos de comunicación	115
Figura 34. Diagrama bloque I de la plataforma de soporte al NDT.....	124
Figura 35. Plataforma para el soporte del NDT.....	127
Figura 36. Diagrama bloque III de la plataforma de soporte al NDT	135
Figura 37. Caso I - Red gemelo físico.....	139
Figura 38. Caso I - Ficheros VNFD cargados en OSM.....	142
Figura 39. Caso I - Representación gráfica fichero NSD.....	142
Figura 40. Caso I - Despliegue de NDT sobre OpenStack	143
Figura 41. Caso I - Detalle despliegue de NDT sobre OpenStack.....	144
Figura 42. Caso I - Listado de subredes asociadas al NDT desplegado en OpenStack.....	145
Figura 43. Caso I - Listado de instancias asociadas al NDT desplegado en OpenStack	147
Figura 44. Caso II - Red gemelo físico.....	150
Figura 45. Caso II - Ficheros VNFD cargados en OSM.....	153
Figura 46. Caso II - Representación gráfica del fichero NSD desde OSM	154
Figura 47. Caso II - Gemelo Digital lanzado como instancia NS lanzada desde OSM	154
Figura 48. Caso II - Representación gráfica del Gemelo Digital de red 5G desplegado sobre el VIM OpenStack conectado a OSM.....	156
Figura 49. Caso II - Listado de subredes asociadas al NDT desplegado en OpenStack	157
Figura 50. Caso II - Listado de instancias asociadas al NDT desplegado en OpenStack.....	157
Figura 51. Core5G monitorizado con la herramienta ntop-ng	158
Figura 52. Detalle de la monitorización de cada VNF del core 5G	159
Figura 53. Estadísticas de uso asociadas al UPF.....	160

Figura 54. Estadísticas de uso asociadas al conector STGUTG	160
Figura 55. Ejemplo del Dataset obtenido desde el NDT vía ntop	161
Figura 56. Detalle de la auto suscripción a los topics MQTT de comportamiento por parte del sistema de gestión	166
Figura 57. Detalle de la auto suscripción a los topics MQTT de comportamiento por parte del agente inteligente del NDT.....	166
Figura 58. Visual sistema de gestión para la información topológica reportada por el escenario del caso I.....	167
Figura 59. Filtrado de datos topológicos reportados por el caso I.....	168
Figura 60. Modelo de datos JSON almacenado para el caso I.....	169
Figura 61. Visual sistema de gestión para la información topológica reportada por el escenario del caso II	170
Figura 62. Filtrado de datos topológicos reportados por el caso II.....	170
Figura 63. Modelo de datos JSON almacenado para el caso I.....	171
Figura 64. Visual sistema de gestión para la información de comportamiento reportada por el escenario del caso II.....	172
Figura 65. Filtrado de datos de comportamiento reportados por el caso II	173
Figura 66. Consulta de la evolución en el consumo de recursos de NF específicas del caso II	174

Lista de Tablas

Tabla 1. Definiciones de gemelo digital.....	21
Tabla 2. Soluciones comerciales basadas en Digital Twin.....	27
Tabla 3. Trabajos relacionados a la aplicabilidad del paradigma Digital Twin.....	30
Tabla 4. SDN - NFV aplicado a Digital Twin	37
Tabla 5. Servicios OpenStack.....	55
Tabla 6. Implementaciones de componentes 5G	70
Tabla 7. Trabajos relacionados a NDT.....	73
Tabla 8. Subgrupos de datos dentro del modelo	88
Tabla 9. Descripción de los campos del modelo de datos.....	91
Tabla 10. Elementos básicos del descriptor NSD.....	99
Tabla 11. Elementos básicos del descriptor VNFD.....	100
Tabla 12. Requisitos relacionados al paradigma de gemelos digitales.....	120
Tabla 13. Requisitos relacionados a la aplicabilidad	121
Tabla 14. Requisitos relacionados con los flujos de comunicación	121
Tabla 15. Requisitos relacionados con la seguridad	122
Tabla 16. Otros requisitos	122
Tabla 17. Flujos de comunicación presentes en el sistema de gestión NDT	129
Tabla 18. Listado de Object IDentifier utilizados por el agente	140

Abreviaturas y acrónimos

3GPP	Thrid Generation Partnership Project
5GC	5G Core
AF	Application Function
AKS	Azure Kubernetes Service
AMF	Access and Mobility Management Function
API	Application Programming Interface
AUSF	Authentication Server Function
AWS	Amazon Web Service
BSS	Business Support System
CAPEX	Capital Expenditure
CLI	Common Line Interface
CP	Connection Points
CPS	Cyber Physical System
DHCP	Dynamic Host Configuration Protocol
DM	Digital Model
DNS	Domain name Server
DS	Digital Shadow
DT	Digital Twin
DTN	Digital Twin Network
EC2	Elastic Compute Cloud
eMBB	Enhanced Mobile Broadband
eNB	Evolved Node B
EPC	Evolved Packet Core
ETSI	European Telecommunications Standards Institute
FCAPS	Fault, Configuration, Accounting, Performance, Security
FDMA	Frequency Division Multiple Access
gNB	gNode B
GNN	Graph Neural Networks
GSM	Global system for Mobile Communications
HNF	Hybrid Network Function

IA	Inteligencia Artificial
IaaS	Infrastructure as a Service
IoDT	Internet of Digital Twin
IoT	Internet of Things
IP	Internet Protocol
ITU	International Telecommunications Union
KPI	Key Performance Indicators
LCM	Life-cycle Management
LTE	Long Term Evolution
LXD	Linux Container Daemon
MEC	Multi-access Edge Computing
ML	Machine Learning
mMTC	Massive Machine-Type Communications
MNDT	Mobile Network Digital Twin
MON	Monitoring
MQTT	Message Queuing Telemetry Transport
N2VC	Network Service to VNF Communication
NB	Node B
NBI	Nortbound Interface
NDT	Network Digital Twin
NEF	Network Exposure Function
NF	Network Function
NFV	Network Function Virtualization
NFV	Network Function Virtualization Management and Orchestration
MANO	
NFVI	Network Function Virtualization Infrastructure
NFVO	NFV Orchestrator
NR	New Radio
NRF	NF Repository Function
NS	Network Service
NSA	Non-Stand Alone
NSD	Network Service Descriptor
NSI	Network Slices Instance

NSSF	Network Slice Selection Function
NTT	Nippon Telegraph and Telephone
OAI	Open Air Interface
OAM	Operation, Administration and Management
OPEX	Operational Expenditure
OSI	Open Systems Interconnection
OSM	Open-Source MANO
OSS	Operator Support System
OWL	Web Ontology Language
P2P	Physical to Physical
P2V	Physical to Virtual
PaaS	Platform as a Service
PCF	Policy Control Function
PLM	Product Life-cycle Management
PNF	Physical Network Function
POL	Policy Module
PQC	Post-Quantum Cryptography
RAN	Radio Access Network
RO	Resource Orchestrator
SA	Stand Alone
SCADA	Supervisory Control and Data Acquisition
SDN	Software Defined Networks
SLA	Service Level Agreement
SMF	Session Management Function
SNMP	Simple Network Management Protocol
STGUTG	Signaling Traffic Generator and User Traffic Generator
TIC	Tecnología de la Información y Comunicaciones
TMN	Telecommunication Management Network
UDM	Unified Data Management
UI	User Interface
UMTS	Universal Mobile Telecommunications System
UPF	User Plane Function
uRLLC	Ultra-Reliable and Low Latency Communications

V2V	Virtual to Virtual
VCA	VNF Configuration and Abstraction
VDU	Virtual Deployment Unit
VIM	Virtualized Infrastructure Manager
VL	Virtual Link
VNF	Virtualized Network Function
VNF	Virtualized Network Function
VNFD	Virtualized Network Function Descriptor
VPC	Virtual Private Cloud
YANG	Yet Another Next Generation

Capítulo 1

Introducción y objetivos

En este primer capítulo se presenta la línea de investigación, así como el alcance de la presente Tesis Doctoral. En primer lugar, la sección 1.1 contextualiza el marco tecnológico actual, centrándose en la evolución de las redes de comunicación móviles, así como las distintas modificaciones como resultado de los diferentes avances tecnológicos.

En concreto, se aborda la motivación de la Tesis Doctoral, centrándose en la aplicación del paradigma de gemelos digitales a redes de comunicación, detallando las capacidades que podrían alcanzarse al implementar este paradigma en términos como la mejora de gestión, optimización de recursos o ciberseguridad.

Tras el contexto y la motivación, la sección 1.2, recoge los objetivos propuestos para la Tesis, alineados con las contribuciones desarrolladas. Seguidamente la sección 1.3 presenta la metodología de trabajo aplicada a la investigación realizada, combinando la investigación documental y experimental para alcanzar los objetivos planteados. A Continuación, la sección 1.4 se presentan las actividades realizadas durante el desarrollo de la tesis doctoral. Finalmente, la sección 1.5 enuncia la organización de la estructura del documento.

1.1. Contexto y motivación

Desde su aparición, las redes de telecomunicaciones, específicamente las redes de comunicación móvil han sufrido una evolución continua, modificando sus arquitecturas con las diferentes especificaciones y estándares, así como con la adopción de nuevas tecnologías.

La primera de estas generaciones de comunicaciones móviles, liberada de manera comercial por la *Nippon Telegraph and Telephone Corporation* (NTT) en Japón en 1979, consistía en una red puramente analógica que ofrecía exclusivamente servicios de voz y planteando problemas como la nula seguridad en las comunicaciones, la baja capacidad de enlace basado en *Frequency Division Multiple Access* (FDMA) o la imprecisión a la hora de realizar la transferencia entre celdas. En 1982, el *European Telecommunications Standards Institute* (ETSI) desarrolla el estándar *Global System for Mobile Communications* (GSM), considerándose un estándar de segunda generación o 2G, cuyo objetivo consistía en mejorar evitar los problemas heredaos de las redes analógicas de telefonía móvil. No obstante, esta segunda generación tuvo varios estándares e implementaciones incompatibles con la primera generación. Debido a estas incompatibilidades, en 1998 se crea la *Third Generation Partnership Project* (3GPP), cuyo objetivos inicial era especificar un sistema global de comunicaciones móviles de tercera generación evolucionando el sistema GSM existente.

Desde su creación, el 3GPP ha sido el organismo encargado de liderar las posteriores generaciones de comunicaciones móviles, desde la especificación en 3G *Universal Mobile Telecommunications System* (UMTS) liberada en el 2000 a través de la *Release 99*, la especificación de 4G o *Long Term Evolution* (LTE) en 2008 a través de la *Release 8*, a la reciente quinta generación 5G en 2018 desde la *Release 15*. Durante estos 25 años, la arquitectura de red ha sufrido numerosos cambios y modificaciones para llevar a cabo la implementación de las distintas generaciones.

Desde su especificación y posterior implantación, todas estas generaciones comparten un elemento común, la gestión de red, de vital importancia para garantizar su correcto funcionamiento. En la década de 1990 la *International Telecommunications Union* (ITU) junto a la *International Organization for Standardization* (OSI) publicaron la series de estándares X.700 (*Management Framework for Open System Interconnection*) [1] enfocados a la gestión de redes y sistemas [X701, 1989], [X710, 1991], [X711, 1991]

[X720, 1992], conocido actualmente como el modelo de referencia OSI. Este modelo de referencia establece cinco áreas funcionales para la gestión definidas por sus siglas en inglés como FCAPS (*fault, configuration, accounting, performance, and security*).

Dado que las redes de comunicación presentan arquitecturas complejas que abarcan no solo redes, sino también servicios, usuarios y entidades comerciales, el modelo OSI no proporciona una visión troncal completa de la gestión en este tipo de redes. Con el objetivo de suplir esa carencia surge la iniciativa para el establecimiento del framework Telecommunication Management Network (TMN), el cual tuvo su primera versión en 1996 con la recomendación M3010 [2]. Este framework define una arquitectura lógica en capas que abarca la gestión de elementos, gestión de redes, gestión de servicios y la gestión empresarial o de negocio.

Esta gestión de red engloba una serie de actividades como son la operación, la administración y el mantenimiento de la red, OAM (Operation, Administration and Management). El cual a través de diversas tecnologías se encarga de gestionar tareas como la actualización de configuraciones, la detección y resolución de problemas en la red, la implementación de nuevas funcionalidades o equipos, la actualización de software, la monitorización y la seguridad en las comunicaciones.

Durante este proceso de evolución continua en las redes de comunicación, cabe destacar un concepto, Network Function Virtualization (NFV), propuesto en 2012 [3], el cual supuso un cambio significativo en la forma de entender, gestionar y utilizar este tipo de redes. Este enfoque de red basado en la tecnología NFV ha planteado un cambio radical en la industria, liderando la transición del dispositivo de red físico tradicional basado en hardware hacia una nueva era basada en *Virtualized Network Functions* (VNF) implementadas vía software. Esta tecnología permite la virtualización de funciones de red, lo cual significa que tareas realizadas tradicionalmente por hardware especializado, pasan a ejecutarse en servidores virtualizados, permitiendo a los operadores de telecomunicación mejorar la flexibilidad, ajustando, actualizando o implementando funciones de red de forma más rápida y sencilla, consiguiendo adaptarse con mayor facilidad a los diversos cambios y demandas por parte de nuevos servicios y aplicaciones. Otra ventaja clave de la adopción de esta tecnología recae en la optimización de recursos, permitiendo a los operadores eliminar la dependencia con hardware específico, reduciendo la inversión en equipos propietarios aprovechando soluciones de hardware

estándar y software OpenSource, traducido directamente en la reducción de los costes operativos [4].

Pese a que la tecnología NFV lleva presente más de una década, ha sido su convergencia con la tecnología de quinta generación de redes móviles la que ha supuesto un hito clave en la evolución de las comunicaciones. La tecnología 5G, desde el lanzamiento de la Release 15 en 2017 por parte del 3GPP, establece la definición para el core de red basado íntegramente en funciones de red virtualizadas ligadas al plano de control. La unión de estas dos tecnologías, sumadas a la tecnología de Software Defined Networks (SDN) ligada al plano de datos, permite a los operadores alcanzar todo el potencial de ambas, implementando, gestionando y escalando estas funciones de red de manera eficiente y dinámica en entornos 5G.

No obstante, pese a las numerosas ventajas que pueda presentar la unión de estas tecnologías, también implica una serie de retos que se traducen en la necesidad de realizar una gestión avanzada debido a la complejidad introducida en la gestión de recursos, la seguridad o la orquestación de servicios. Tareas como la automatización, la monitorización en tiempo real o la capacidad de gestión dinámica de los recursos asignados a la red se vuelven elementos críticos para garantizar un rendimiento óptimo y una alta disponibilidad.

Siguiendo esta evolución tecnológica, es reseñable destacar que, durante los últimos años, ha surgido un nuevo concepto conocido como Digital Twin (DT) el cual plantea un nuevo paradigma para el diseño, implementación y operación en las redes de próxima generación. La base de esta tecnología se centra en modelar de forma precisa un elemento físico para crear su gemelo digital estableciendo un enlace bidireccional entre ambos permitiendo así una coevolución conjunta a lo largo de su ciclo de vida. Este nuevo paradigma se está aplicando en diversos campos como la industria 4.0, la ciberseguridad, medicina, industria aeroespacial, sistemas inteligentes, entre otros [5]. No obstante, en el contexto de la presente tesis doctoral, se aborda el concepto de gemelos digitales aplicado a redes móviles, denominado Mobile Network Digital Twin (MNDT), como una tecnología habilitadora para aprovechar el máximo potencial al integrar los últimos avances en tecnología 5G, redes definidas por software y funciones de red virtualizadas. Esta integración crearía un ecosistema tecnológico completo el cual abordaría áreas como la ciberseguridad, el mantenimiento preventivo de la red, el control de la calidad de servicio,

la optimización de la red y la autoconfiguración o planificación de acciones de red teniendo en cuenta el comportamiento esperado de los usuarios.

A pesar de las numerosas ventajas que plantea este nuevo paradigma, uno de sus principales desafíos radica en la heterogeneidad del equipamiento del que se componen las nuevas redes de operadores, lo que desemboca en la necesidad de definir y establecer nuevas metodologías centradas en la recolección de datos de una naturaleza específica con el objetivo de obtener una representación semántica de la red, así como caracterizar su comportamiento para el posterior modelado en forma de MNDT. Estos dos aspectos son críticos dado que el éxito de la tecnología basada en gemelos digitales depende en gran medida de su adecuado modelado del elemento físico que representa, lo que a su vez requiere una amplia recopilación de datos para su posterior procesamiento, filtrado y generación de conocimiento. Adicionalmente a la problemática asociada a la adquisición de datos y el modelado, existe otro desafío aun mayor relacionado con las comunicaciones entre el gemelo digital y su contraparte física, las cuales son el componente que da sentido a este tipo de paradigma. Debido principalmente a dos factores, el primero de ellos está relacionado con la sensibilidad de los datos intercambiados entre los dos gemelos dentro de un ecosistema MNDT, lo que implica que este tipo de comunicación requiere el uso de mecanismos de seguridad que garanticen la confidencialidad, integridad y autenticidad de los datos. El segundo está relacionado a los distintos flujos de información que pueden intercambiar los gemelos.

Durante estos veinticinco años la red de telecomunicaciones ha experimentado un enorme crecimiento y una evolución tecnológica reseñable, pasando de una infraestructura cerrada basada en conmutación de circuitos a una infraestructura digital capaz de ofrecer diversos servicios y aplicaciones. Por ello existe una necesidad urgente de que la gestión en este tipo de redes sea capaz de hacer frente a la heterogeneidad tecnológica, la movilidad, la integración de nuevos sistemas tanto hardware como software, asegurar la calidad de servicio en la red, así como ampliar el alcance de la gestión para cubrir nuevas aplicaciones, servicios y usuarios.

1.2. Objetivos

El objetivo general de la tesis doctoral es formular propuestas que contribuyan al diseño, despliegue e interconexión de gemelos digitales de red aplicados a redes de nueva generación. Dicho objetivo nace de la hipótesis de que las distintas tecnologías asociadas a las redes de nueva generación, como son las redes 5G y la futura estandarización de las redes 6G, van a integrar los paradigmas tecnológicos relacionados con Network Function Virtualization (NFV) y Software Defined Networking (SDN) los cuales combinados con el paradigma de gemelos digitales permitirían facilitar su óptima integración de manera progresiva. Sobre esta hipótesis se plantea que esta adopción permitirá crear soluciones de gestión de red y optimización de recursos avanzadas aprovechando las ventajas ofrecidas por la tecnologías basadas en gemelos digitales.

Este objetivo general se puede descomponer en los siguientes objetivos específicos:

1. Realizar un análisis exhaustivo del estado del arte, el cual permita obtener una visión completa en referencia a la unión del paradigma de gemelos digitales aplicado a la tecnología 5G, centrándose especialmente en el estado actual de los diferentes mecanismos de recolección de información para su posterior modelado y despliegue en forma de gemelo digital, con el objetivo de identificar los retos y limitaciones que plantea la aplicación de tecnología basada en gemelos digitales en este tipo de redes de comunicación de última generación.
2. Proponer y diseñar una metodología que permita, a partir de una infraestructura física, modelar y desplegar gemelos digitales de red. Esta metodología abarca distintas fases, desde la captación de información en los gemelos físicos, hasta la interconexión y realimentación con los gemelos digitales, siguiendo los diferentes modelos de framework propuestos por la ETSI.

Específicamente se abordará:

- a. Diseño basado en la identificación de retos y limitaciones de la tecnología de Gemelos Digitales aplicada a redes de nueva generación tras el análisis de la literatura realizada en el capítulo anterior.
- b. Diseño de principios de implementación y funcionamiento de una metodología que permita la creación e interconexión de gemelos digitales

- c. Diseño de arquitectura y mecanismos que permitan llevar a cabo la implementación técnica de la metodología asociada al paradigma de gemelos digitales aplicado a redes de comunicación de última generación.
- Esta metodología servirá como guía para la creación de un ecosistema NDT capaz de permitir la implementación técnica del paradigma de gemelos digitales aplicado a redes de nueva generación.
- 3. Proponer y diseñar una arquitectura, alineada con el framework NFV establecido por la ETSI, que permita la implementación de la metodología diseñada satisfaciendo los distintos requisitos y requerimientos de las tecnologías implicadas. Específicamente se llevará a cabo:
 - a. Identificación de las tecnologías habilitadoras capaces de cubrir los requisitos específicos asociados al paradigma de Gemelos Digitales, haciendo especial énfasis en la gestión del ciclo de vida de los gemelos, así como la comunicación bidireccional entre ellos.
 - b. Diseño de subsistemas modulares capaces de llevar a cabo la implementación del NDT, así como la orquestación de las comunicaciones, tanto la asociada a los propios gemelos, como la asociada a la propia gestión de los elementos que conforman el ecosistema NDT.
 - c. Validación, tanto de la metodología como la plataforma diseñada, aplicadas a diferentes casos de estudio.

Esta plataforma pretende ser un punto inicial capaz de llevar a cabo la implementación del paradigma basado en gemelos digitales aplicado al caso específico de redes de comunicación de nueva generación.

- 4. Proponer y diseñar agentes software encargados de la recolección e intercambio de información, así como la definición de un sistema encargado de la gestión e interconexión de los distintos flujos de información presentes en el ecosistema dotando a los mismos de un nivel de seguridad apropiado para la tipología de datos intercambiada entre los gemelos.
- 5. Validar las propuestas presentadas en la presente tesis doctoral mediante la implementación de escenarios virtualizados abordando diversos ámbitos de importancia identificados en el estado del arte como puede ser la ciberseguridad o la gestión de red.

1.3. Metodología de trabajo

La metodología de investigación realizada en la presente tesis doctoral se aborda desde dos perspectivas. La primera, documental, realizando una revisión exhaustiva de conceptos, así como propuestas desarrolladas por la academia e industria con el objetivo de identificar problemas, carencias o necesidades con el fin de proponer nuevas soluciones que permitan contribuir y superar el actual estado del arte. La segunda, experimental, se centra en diseñar, evaluar, validar y discutir los beneficios de las propuestas planteadas en la tesis doctoral con casos de uso específicos. Dicha metodología tiene como objetivo el cumplimiento de cada uno de los objetivos citados anteriormente, para ello se definen las siguientes tareas:

- **Tarea 1.** Identificación de las necesidades, retos y problemas asociadas al concepto de gemelo digital en el caso específico de redes de comunicación.
- **Tarea 2.** Estudio del estado del arte NFV, 5G, gemelos digitales, monitorización, aprovisionamiento, etc, identificando publicaciones, software OpenSource y soluciones propuestas por organismos de estandarización.
- **Tarea 3.** Estudio e identificación de publicaciones y software OpenSource relacionado con NDT con el objetivo de llevar a cabo del objetivo 2.
- **Tarea 4.** Desarrollo de una metodología que permita llevar a cabo el NDT
- **Tarea 5.** Especificación de arquitectura y componentes que permitan la implementación software de la metodología.
- **Tarea 6.** Prototipado de las soluciones y su aplicabilidad a distintas pruebas de concepto / casos de estudio-uso.

1.4. Actividades desarrolladas en el marco de la presente tesis doctoral

1.4.1. Participación en Proyectos de investigación competitivos:

- Plan Nacional 5G.
 - Proyecto Piloto 5G en el puerto de Algeciras (Retevisión S.AU, 2020)
 - Colaboración para la empresa Cellnex(Retevisión) en el desarrollo de dos casos de uso en el ámbito de la seguridad ciudadana dentro del proyecto piloto 5G liderado por Vodafone España y Huawei en Andalucía seleccionado en la primera convocatoria de solicitudes de subvenciones

públicas para promover la tecnología 5G por el Ministerio de Economía y Empresa de España (Ref. MINECO/Red.es, 2018/C007/00043535).

- La participación del grupo RSTI se centra en el desarrollo y despliegue sobre una plataforma MEC (Multi-acces Edge Computing) de una serie de módulos software para dos aplicaciones: 1) Sistema de guiado automático de vehículos para el Puerto de Algeciras mediante reconocimiento de matrículas y paneles luminosos y 2) Generación de alertas policiales de identificación de personas y vehículos sospechosos en el paso fronterizo del puerto.
- Beyond 5G Enhanced Management through digital twins based on artificial Intelligence (B5GEMINI).
 - El proyecto B5GEMINI, pretende el desarrollo de una red gemela digital aplicada a un entorno de núcleo 5G y su evolución hacia 6G donde se desarrollen casos de uso de interés para el despliegue de escenarios avanzados tales como la ciberseguridad o la gestión de red, en las cuales los operadores de la red en coordinación con proveedores de servicio y de equipos de telecomunicación puedan validar las nuevas funcionalidades con carácter previo a su incorporación a la red o bien emular situaciones existentes en la red y analizar el efecto de aplicar distintas políticas en la red.
 - Trabajo coordinado formado por dos proyectos UNICO-5G I+D, B5GEMINI-AIUC y B5GEMINI-INFRA financiados por el Ministerio de Asuntos Económicos y Transformación Digital del Gobierno de España y NextGenerationEU (Plan de Recuperación, Transformación y Resiliencia - RETC)
 - El grupo RSTI lidera el proyecto B5GEMINI-INFRA cuyo objetivo es el diseño, desarrollo, despliegue, operación y soporte de una infraestructura Digital Twin Network (NDT) para redes B5G que permita el desarrollo de casos de uso y experimentos sobre la misma.

- 5G Verticals INNOvation Infrastructure (5GVinni).
 - El concepto 5G-VINNI es desarrollar una instalación E2E 5G que pueda usarse para demostrar primero la implementación práctica de la infraestructura para respaldar los KPI clave de 5G y luego permitir que las industrias verticales prueben y validen aplicaciones específicas que dependen de esos KPI. Sin embargo, 5G-VINNI no pretende ser simplemente un grupo de sitios de instalaciones de prueba interconectados: está respaldado por principios que permitirán arquitecturas de red, implementación de servicios y pruebas altamente dinámicas y flexibles, que crearán una nueva implementación de servicios técnicos y comerciales.
 - El grupo RSTI de la Universidad Politécnica de Madrid participó en la última fase del proyecto europeo 5G-VINNI (H2020-ICT-2018-20/H2020-ICT-2018-1) como Third Party de Telefónica I+D desde noviembre de 2020 hasta el final del proyecto.
- ACROSS (Automated zero-touch cross-layer provisioning framework for 5G and beyond vertical services)
 - ACROSS es una plataforma segura de gestión de servicios y redes de extremo a extremo que tiene como objetivo abordar los requisitos dinámicos y complejos de los servicios modernos y futuros en un entorno de procesamiento en la nube. Proporciona una capa de orquestación de servicios altamente escalable, administrada por un orquestador de servicios nativo de la nube multidominio, para administrar infraestructuras multifacéticas en implementaciones de extremo a núcleo distribuidas geográficamente. Aumenta la seguridad y la confianza sobre los dispositivos subyacentes y las fuentes de datos mientras explota conjuntamente los datos, la programabilidad y la IA para mejorar el rendimiento y burlar la orquestación de servicios existente.
 - El proyecto ACROSS ha recibido financiación del programa de investigación e innovación Horizonte Europa de la Unión Europea en virtud del Acuerdo de subvención nº 101097122, así como de la Empresa Conjunta de Servicios y Redes Inteligentes (SNS JU).

- XTREME (X-Treme edge network innovations for advanced multi-segment services)
 - El proyecto XTREME tiene como objetivo abordar la integración de dispositivos presentes en el Edge en el ámbito de la gestión y orquestación de las redes 6G. En relación con ello, XTREME propone una transformación arquitectónica fundamental en 6G, donde se contempla tanto una extensión horizontal como una extensión vertical. (A) Extensión horizontal: los recursos de tipo Extreme Edge , es decir, los recursos del lado de la UE/terminal más allá del borde de la infraestructura fija, se integran funcionalmente con, y extienden, el continuo de cómputo-conexión 6G, es decir, se convierten en recursos totalmente visibles y configurables sujetos a M&O. La transformación buscada permitirá no solo la extensión del continuo de cómputo, sino que también dará lugar a topologías de servicios flexibles que permitan la integración dinámica y sin problemas de segmentos NTN y/o comunicaciones localizadas, sujetas a condiciones operativas en el extreme Edge. (B) Extensión vertical: basándose en los fundamentos de la extensión horizontal, se diseñan mecanismos para respaldar la provisión de servicios holísticos aprovechando la capacidad para monitorear y controlar estos dispositivos/recursos, junto con componentes de servicio vertical (aplicación), de manera verdaderamente de extremo a extremo, dirigidos tanto a los indicadores clave de rendimiento (KPI) de red como de aplicación. relacionadas con la gestión de dispositivos en el Edge.
 - El grupo RSTI lidera el paquete de trabajo 4, en relación con gemelos digitales de red. Esta tarea tiene como objetivo abordar los desafíos en relación con toma de decisiones, aprovechando el paradigma de en gemelos digitales aplicado al extreme Edge para ofrecer una provisión holística, proactiva y eficiente de servicios.

1.4.2. Publicaciones relacionadas con el trabajo de investigación:

- M. Sanz-Rodrigo, D. Rivera, J.I. Moreno, M. Álvarez-Campana, D.R. López. ***“Digital Twins for 5G Networks: A Modeling and Deployment Methodology”***. IEEE Access (2023). April 2023. DOI: <https://doi.org/10.1109/access.2023.3267548>
- A. Mozo, Amit Karamchandani, S. Gómez-Canaval, M. Sanz, J. I. Moreno Novella, and A. Pastor. ***“B5GEMINI: AI-Driven Network Digital Twin”***. Sensors 2022, 22, 4106. DOI: <https://doi.org/10.3390/s22114106>
- D. Rivera Pinto, J.I. Moreno Novella, M. Sanz, D. Lopez, A. Mozo. ***“Providing Heterogeneous Signaling and User Traffic for 5G Core Network Functional Testing”***. IEEE Access 2023. Vol 11. ISSN: 2169-3536. DOI: <https://doi.org/10.1109/ACCESS.2022.3233412>
- Amit Karamchandani, Mario Sanz, Angela Burgaleta, Luis de la Cal, Alberto Mozo, Jose Ignacio Moreno, Antonio Pastor, Diego Lopez. ***“Design of an AI-driven Network Digital Twin for advanced 5G-6G network management”***. NOMS TNT 2024 – 3rd International Workshop on Technologies for Network Twins. Paper ID 2024068852
- M. Sanz Rodrigo, D. Rivera Pinto, J.I. Moreno Novella and X. Larriva Novo (2022). ***“Diseño de un ecosistema DTN basado en NFV para el análisis de ciberseguridad en redes 5G”***. Actas de las VII Jornadas Nacionales de Investigación en Ciberseguridad (JNIC 2022). España. 27-29 Junio 2022. ISBN 978-84-88734-13-6.

1.4.3. Otras publicaciones:

- M. Sanz, J.I. Moreno Novella, G. López, J. Matanza and J. Berrocal Colmenarejo, ***“Web-Based Toolkit for Performance Simulation and Analysis of Power Line Communication Networks”***. Energies 2021, 14, 6475. DOI: <https://doi.org/10.3390/en14206475>
- Xavier Larriva-Novo, Víctor A. Villagrà, Mario Vega-Barbas, Diego Rivera, Mario Sanz Rodrigo. ***“An IoT-Focused Intrusion Detection System Approach Based on Preprocessing Characterization for Cybersecurity Datasets”***. Sensors 2021, 21, 656. DOI: <https://doi.org/10.3390/s21020656>

- Mario Vega-Barbas, Manuel Álvarez-Campana, Diego Rivera, Mario Sanz, Julio Berrocal. *“AFOROS: A Low-Cost Wi-Fi-Based Monitoring System for Estimating Occupancy of Public Spaces”*. Sensors 2021, 21, 3863. DOI: <https://doi.org/10.3390/s21113863>
- Carmen Sánchez-Zas, Xavier Larriva-Novo, Víctor A.Villagrà, Mario Sanz Rodrigo, José Ignacio Moreno. *“Design and Evaluation of Unsupervised Machine Learning Models for Anomaly Detection in Streaming Cybersecurity Logs”*. Mathematics 2022, 10, 4043. DOI: <https://doi.org/10.3390/math10214043>

1.4.4. Dirección de Trabajos de fin de titulación:

- **Implementación de cores de red 5G sobre arquitecturas MANO a través de la adaptación de descriptores NFV.**
 - Autor: Wang, Xinzhou
 - Calificación: 9
 - Fecha de lectura: 14 de julio de 2023
- **Desarrollo e implementación de una metodología para la caracterización y despliegue de gemelos digitales de red 5G.**
 - Autor: Martínez Gonzalez, Ana
 - Calificación: 9
 - Fecha de lectura: 14 de julio de 2023
- **Desarrollo de un sistema para el despliegue automatizado de redes 5G sobre arquitecturas NFV mediante Open Source MANO.** Fecha de lectura: 14 de julio de 2023
 - Autor: Remezov, Sergiy
 - Calificación: 6.5
 - Fecha de lectura: 14 de julio de 2023
- **Diseño e implementación de tecnología basada en gemelos digitales aplicado a entornos 5G.** Fecha de lectura: 17 de julio de 2023
 - Autor: Diez Cubillo, Covadonga
 - Calificación: 8
 - Fecha de lectura: 14 de julio de 2023

1.5. Estructura del documento

La investigación realizada durante el desarrollo de la tesis doctoral está estructurada en el presente documento por capítulos, los cuales abordan los conceptos teóricos, tecnologías, necesidades, retos, propuestas, desarrollos y evaluaciones desarrolladas a lo largo del trabajo. A continuación, se describe la estructura de la memoria:

- Capítulo 1. *Introducción*

En este capítulo se da una visión general, contextualizando el área de investigación de la tesis doctoral, presentando la problemática que motiva las distintas propuestas y trabajos llevados a cabo. Para ello se describe la metodología utilizada durante la investigación documental y experimental aplicada permitiendo dar una solución al problema principal. Adicionalmente, se presenta una lista de trabajos y artículos realizados en el contexto de la investigación, así como una serie de objetivos que derivan en contribuciones concretas de esta tesis doctoral.

- Capítulo 2. *Estado del Arte*

Este capítulo presenta una revisión del estado del arte sobre diferentes líneas de investigación relacionadas con la tesis doctoral, así como las distintas tecnologías facilitadoras. En primer lugar, el capítulo presenta las bases teóricas que permiten comprender el paradigma basado en gemelos digitales, identificando sus características, sus ventajas y sus limitaciones o retos identificados. Mas adelante se abordan las tecnologías habilitadoras para la aplicabilidad del paradigma de gemelos digitales sobre redes de comunicación, en concreto sobre redes de nueva generación, identificando una serie de áreas de contribución como son el modelado de gemelos, la interconexión, los flujos de comunicación necesarios en este tipo de entornos, así como la necesidad de mecanismos de securización en las comunicaciones. Seguidamente se realiza una revisión de las diferentes tecnologías y framework disponibles para llevar a cabo dicha aproximación, estableciendo los elementos indispensables, así como la identificación de nuevos elementos necesarios para alcanzar el funcionamiento esperado. Finalmente, el capítulo aborda la problemática asociada al aprovisionamiento de este tipo de redes en forma de gemelos digitales, así como la tecnología necesaria para realizar una gestión en la fase de interconexión e intercambio de información entre los elementos físicos y su contraparte digital.

- Capítulo 3. *Metodología para el modelado y despliegue de gemelos digitales de red*

Este capítulo incluye la contribución principal de esta tesis doctoral. Describe el diseño de la metodología alineada con el framework NFV propuesto por la ETSI, así como con los recientes drafts con relación a gemelos digitales de red realizados por dicho organismo. Detalla el diseño y funcionamiento de los componentes vinculados a cada fase específica de la metodología, así como las interfaces definidas para la interacción entre los diversos componentes. Adicionalmente, se presenta el diseño de agentes recolectores de información, los cuales permitirán modelar los gemelos físicos en un primer instante, así como reconfigurar el gemelo digital en el caso de detectar algún cambio significativo en la configuración o su funcionamiento. En relación con los datos recogidos por estos agentes, se presenta el modelo de datos establecido en esta tesis doctoral, el cual permite llevar a cabo el modelado y despliegue de los elementos como gemelos digitales.

- Capítulo 4. *Plataforma para el soporte del NDT*

Este capítulo se aborda la arquitectura software diseñada para llevar a cabo la implementación práctica de metodología descrita en el capítulo 3. Esta arquitectura unifica todos los elementos asociados a cada una de las fases de la metodología, tomando como elemento base las VNFs, las cuales se desplegarán sobre entornos completamente virtualizados. Además, presenta la definición de un nuevo elemento encargado de definir, gestionar y securizar los flujos de información intercambiados entre el mundo físico y su contraparte digital.

- Capítulo 5. *Casos de estudio*

Este capítulo se presentan los distintos métodos de evaluación llevados a cabo con el fin de cumplir los objetivos definidos en la presente tesis doctoral. Dicha evaluación se puede diferenciar en dos aproximaciones. La primera de ellas consiste en la evaluación individual de cada uno de los módulos que conforman la implementación de las diferentes fases de la metodología propuesta. La segunda consiste en la evaluación completa asociada a diferentes casos de estudio como puede ser el uso de la tecnología basada en gemelos digitales para la obtención de datasets de tráfico, los cuales posteriormente puedan ser utilizados por algoritmos de machine Learning o inteligencia artificial para realizar una gestión inteligente de red o la detección automatizada de problemas de ciberseguridad en este tipo de redes.

▪ Capítulo 6. *Conclusiones y líneas futuras*

Finalmente, este capítulo presenta las conclusiones obtenidas a través del trabajo de investigación y desarrollo realizado durante la tesis doctoral. En él se analiza el cumplimiento de los objetivos propuestos en la introducción e identifica cuales son las líneas futuras para la continuación del trabajo de investigación postdoctoral.

Capítulo 2

Estado del Arte

En este capítulo se realiza un estudio de los diferentes conceptos propuestos o utilizados a lo largo de la tesis, con el fin de obtener una visión general sobre el origen, la evolución, el estado actual de las tecnologías y la motivación que surge para llevar a cabo el desarrollo de la presente tesis doctoral.

La investigación realizada durante la tesis introduce en la sección 0 el paradigma de Gemelos Digitales, especificando a lo largo del capítulo, su aplicabilidad al campo de redes de comunicación. En primer lugar, se presenta la evolución del término de gemelo digital, sus principios básicos, las tecnologías actuales que permiten llevar a cabo su implementación, así como una serie de trabajos relacionados que servirán para contextualizar los retos o carencias identificadas en este ámbito para poder ser abordadas en el desarrollo de esta investigación.

Tras la contextualización general, la sección 2.2 se centra en la aplicabilidad del paradigma de gemelos digitales sobre redes de comunicación. Tomando como referencia el draft realizado por el IETF, el cual describe de los pilares fundamentales para los NDTs, se presentan las distintas herramientas y tecnologías habilitadoras para poder llevar a cabo su implementación. Durante esta investigación, se pone de manifiesto los retos que plantean los NDTs, así como diversas carencias o problemas, los cuales serán tomados como punto de partida para las contribuciones de la presente tesis doctoral.

Una vez presentado el término NDT, se abordan en la sección 2.3 una serie de tecnologías capaces de ofrecer soporte y facilitar la implementación técnica del paradigma DT aplicado a redes.

Finalmente, la sección 2.4 presenta las últimas evoluciones en tecnología de redes móviles, las cuales pueden alcanzar su máximo potencial integrando las soluciones basadas en NDTs.

2.1. Gemelos Digitales. Conceptos, principios y habilitadores

Durante los últimos años, el concepto de gemelos digitales ha despertado gran interés tanto en la comunidad académica como en la industria debido a los beneficios que promete este tipo de tecnología aplicada a campos multidisciplinarios.

Según información aportada por Gartner en 2019 [6] sitúa a esta tecnología como un elemento relevante para muchas organizaciones. En este reporte se establece que en 2020 el 75% de las organizaciones con relación a la tecnología del internet de las cosas o Internet of Things (IoT) utilizan o tienen pensado utilizar la tecnología de gemelos digitales. Gartner estima que para el 2027, el 40% de las grandes compañías mundiales utilizarán esta tecnología en sus proyectos para mejorar sus productos y servicios [7].

Adicionalmente a los estudios realizados por Gartner, existen datos proporcionados por el Global Market Insight, los cuales valoran al mercado relacionado a gemelos digitales en aproximadamente 8 billones de dólares en 2022 [8] , previendo un crecimiento de más del 25% en el periodo comprendido entre el 2023 a 2032. En relación con la valoración del mercado, existe otro reporte realizado por Global Technology Research, el cual estima un crecimiento de alrededor de 32 billones de dólares de este mercado en el periodo comprendido entre el 2023 al 2027 [9]. Sin duda estos datos ponen de manifiesto la importancia y aceptación de esta tecnología, tanto en términos de cifras de negocio como en la aplicabilidad y creación de nuevos servicios mejorados.

No obstante, este concepto no es algo nuevo, en la **Tabla 1** se muestran las definiciones más relevantes con relación al paradigma DT. El profesor Michael Grieves, en 2002, lo presenta en el ámbito universitario aplicado a la gestión del ciclo de vida de un producto, PLM por sus siglas en inglés (Product life-cycle management), en su trabajo con título “Product lifecycle management: the new paradigm for enterprises” [10]. En esta primera aproximación, se define el concepto de gemelo digital como un equivalente virtual digital de un producto físico. Tras este esbozo inicial del concepto de gemelo digital, o entidad virtual ligada a una entidad física, han surgido diferentes propuestas o aproximaciones a lo largo de los años. Las más relevantes son las realizadas por Framling [11] en 2013, que propone una arquitectura basada en agentes, donde cada producto tiene una correspondencia con su contraparte virtual o agente asociado, siendo esto un concepto similar al concepto actual de gemelo digital. En 2010, se realiza la primera aproximación del concepto de manera práctica por parte de la NASA, la cual desarrolló dos

transbordadores Apollo [Ref] con el objetivo de poder simular y reflejar en ellos el estado durante los entrenamientos de vuelo. Siguiendo con esta línea temporal, en el periodo comprendido entre 2010 y 2012, Tuegel [12] aplica el concepto de gemelo digital de manera práctica en el campo aeroespacial, utilizando el gemelo digital como sensor virtual para tratar de predecir la vida útil de la estructura de aeronaves asegurando así su integridad estructural. Ligado al campo de la aviación, en 2013 la fuerza aérea de los estados unidos propone el concepto de amenaza digital, donde la parte digital hace referencia al framework de comunicación utilizado para gestionar todos los datos de producción de manera virtual y lógica [13]. Tras años de evolución y aplicación del concepto de manera abstracta, Grieves y Vickers dan la primera definición formal del concepto de gemelo digital [14], identificando tres aspectos fundamentales, el primero trata de un espacio real el cual contiene al objeto físico, el segundo de un espacio virtual que contiene al objeto virtual y el tercero haciendo referencia a la conexión entre ambos para habilitar el intercambio de información en ambos sentidos. Finalmente, tras esta definición formal, en 2017 Barricelli [15] propone las primeras aproximaciones de la tecnología basada en gemelos digitales aplicada al campo del IoT.

Tabla 1. Definiciones de gemelo digital

Autor	Definición
Grieves and Vickers [14]	“Equivalente virtual digital de un producto físico”
Fritzingier [16]	“Contrapartida de un objeto físico”
Negri [17]	“Representación virtual de un sistema de producción capaz de ejecutarse en diferentes disciplinas de simulación que se caracteriza por la sincronización entre el sistema virtual y el real, gracias a datos sensados y dispositivos inteligentes conectados, modelos matemáticos y elaboración de datos en tiempo real”
Zheng [18]	“Sistema integrado que puede simular, supervisar, calcular, regular y controlar el estado y el proceso del sistema”
Digital Twin Consortium [19]	“Representación virtual de entidades y procesos del mundo real, sincronizados con una frecuencia y fidelidad determinadas”
Industrial Internet Consortium [20]	“Representación digital formal de algún bien, proceso o sistema que captura atributos y comportamientos de esa entidad adecuados para la comunicación, el almacenamiento, la interpretación o el procesamiento dentro de un contexto determinado”
Centre for Digital Built Britain (CDBB) [21]	“Representación digital de activos, procesos o sistemas contruidos artificialmente”
Siemens [22]	“Es una representación virtual de un producto o proceso físico que se utiliza para comprender y predecir las características de rendimiento de su contraparte física”
GE Digital [23]	“Representaciones software de activos y procesos que se utilizan para comprender, predecir y optimizar el rendimiento con el fin de lograr mejores resultados comerciales”
Söderberg [24]	“Algoritmos de optimización más rápidos, mayor potencia informática y cantidad de datos enfocados en el área de simulación para lograr el control y la optimización en tiempo real de productos y sistemas en producción”
Tao [25]	“Simulación integrada multifísica, multiescala y probabilística compuesta por un producto físico, un producto virtual, datos, servicios y conexiones entre ellos”
Liu [26]	“Modelo vivo de un active o sistema físico el cual se adapta continuamente a los cambios operativos en función de la información y los datos recopilados en línea, pudiendo pronosticar el futuro de la contraparte física correspondiente”

2.1.1. Conceptos

Pese a que, como se ha visto en la sección anterior, el paradigma tecnológico de Digital Twin (DT) lleva presente más de una década en el ámbito académico e industrial, aún siguen surgiendo diferentes conceptos y definiciones asociadas principalmente a los casos de uso específicos para la aplicación de esta tecnología. A continuación, revisaremos los principales conceptos, dimensiones de trabajo y los tipos de comunicación asociados a este paradigma.

En primer lugar, tenemos el concepto tradicional de gemelo digital, el cual, como se ve en las definiciones presentadas en la **Tabla 1**, trata de ser una representación virtual en tiempo real de un objeto, proceso o sistema físico. Un gemelo digital está puramente basado en datos y modelos computarizados capaces de replicar las características propias, así como el comportamiento del elemento físico modelado, consiguiendo una implementación virtual computacionalmente más simple que el elemento original, evitando así caer en la paradoja de la replicación total y cumpliendo con la característica principal asociada a esta tecnología, la comunicación bidireccional necesaria para una coevolución colaborativa entre la parte digital y su contraparte física.

Ligado al concepto original de gemelo digital, surge en la literatura el concepto de *Digital Twin Network* (DTN). Este concepto hace referencia a una red de gemelos digitales, los cuales pueden colaborar y comunicarse entre sí, permitiendo replicar de esta manera entornos, procesos o sistemas reales más complejos con el objetivo de mejorar la optimización y el control.

Adicionalmente al concepto de gemelo digital, cabe destacar el concepto de *Cyber Physical System* (CPS), el cual guarda ciertos parecidos con el concepto de gemelo digital. En este tipo de sistemas está presente una parte real y otra virtual. Típicamente se trata de sistemas que integran elementos virtuales y físicos, coordinados con el objetivo de controlar y monitorizar procesos en el mundo real incluyendo sensores, actuadores, redes de comunicación y componentes software que permiten la interacción entre ambos mundos. Un ejemplo de este tipo de sistemas, presente desde hace décadas, son los sistemas *Supervisory Control and Data Acquisition* (SCADA) para el control industrial, en los cuales a través de una interfaz *Human Machine Interface* (HMI) software es posible llevar a cabo acciones sobre elementos físicos remotos, así como consultar el estado general del sistema físico.

Dentro del paradigma Digital Twin, se enfatiza en un componente base definido como Digital Model (DM) Este modelo digital es el encargado de la representación o modelado del gemelo físico, teniendo en cuenta sus características físicas, su comportamiento, configuraciones, y relaciones con el entorno. Estos modelos no solo proporcionan una representación estática del gemelo físico a modelar, sino que permiten la simulación y el análisis en tiempo real, pudiendo ser utilizados para tareas como la predicción de rendimiento, la optimización de operaciones, la detección de fallos o la toma de decisiones basada en datos reales. Este tipo de tareas permiten al gemelo digital evolucionar continuamente a medida que se recopilan nuevos datos desde la contraparte física, permitiendo así mejorar los algoritmos de modelado con el fin de reflejar en tiempo real el estado y comportamiento de los elementos físicos en el ámbito digital.

Como ya se ha comentado, el punto diferenciador del paradigma DT reside en la comunicación bidireccional entre el mundo físico y el mundo digital, la cual habilita la coevolución de los gemelos a lo largo de su ciclo de vida. No obstante, es importante remarcar otro concepto presente en el estado del arte asociado al paradigma de gemelos digitales, conocido como *Digital Shadow (DS)*, el cual en esencia guarda muchas similitudes con el concepto original de *Digital Twin*, con la única diferencia de que la información intercambiada entre los gemelos no es automática en ambos sentidos, y no se realiza en tiempo real. En este caso el intercambio de la información que se realiza desde el gemelo físico hacia el gemelo digital es automática, pero el intercambio en el otro sentido se ha de realizar de manera manual, permitiendo utilizar la parte digital del ecosistema como un campo de pruebas alimentadas con datos del entorno real.

No obstante, pese a que los conceptos vistos están asentados y aceptados tanto por la comunidad académica como por parte de la industria, con la aplicación del paradigma a diferentes casos de uso surgen conceptos y terminologías nuevas, como son el concepto de *Network Digital Twin (NDT)*, el cual se abordará en puntos posteriores y aplica el paradigma DT a redes de comunicación, y el concepto de *Mobile Network Digital Twin (MNDT)*, siendo una subcategoría dentro del concepto general NDT, aplicado a redes de comunicaciones específicas como 5G-6G o el concepto de *Internet of Digital Twin (IoDT)* que aplica el paradigma a IoT.

Independientemente del campo de aplicación de los DTs, todos ellos implementan una serie de dimensiones, las cuales fueron propuestas originalmente por Michael Grieves

[10], y posteriormente ampliadas por Tao [25] con el objetivo de establecer unas bases comunes para esta tecnología. El primero define el espacio real, donde se encuentran los objetos físicos, el espacio virtual que alberga los productos virtuales y define la interconexión entre los objetos físicos y los productos virtuales. El segundo agrega la dimensión del controlador de datos, encargado de gestionar la información de contexto del gemelo digital y define los servicios que se pueden comunicar con el gemelo digital, como pueden ser proveedores de información, aplicaciones de clientes, consumidores de datos, etc. Esta clasificación de dimensiones propuesta por Grieves y Tao ha sido utilizada como base para las distintas propuestas de arquitecturas basadas en gemelos digitales.

Dentro de las cinco dimensiones descritas, cuatro de ellas tienen una fuerte dependencia en cuanto a requisitos de funcionamiento e implementación en base al caso de uso o área de aplicabilidad. No obstante, la dimensión relacionada a la comunicación, pese a poder presentar diferencias en base a área de aplicación, permite diferenciar tres tipos de comunicaciones, *Physical to Virtual* (P2V), *Physical to Physical* (P2P) y *Virtual to Virtual* (V2V), que van a estar siempre presentes en un ecosistema DT. En el caso de las comunicaciones P2V, el objeto, proceso o entidad física envía información en tiempo real, ya sea de manera inalámbrica o cableada, a su gemelo digital, permitiendo además recibir la realimentación generada desde su gemelo digital. Este tipo de comunicación plantea necesidades como la baja latencia para garantizar el tiempo real, la alta capacidad de canal para permitir el intercambio de grandes volúmenes de datos, así como cumplir con la privacidad y seguridad de los datos intercambiados entre ambas dimensiones. En el caso de las comunicaciones P2P, se trata de comunicaciones íntegramente realizadas en la dimensión física, permitiendo que varios sistemas físicos colaboren entre sí. En este caso, la comunicación P2P no es algo nuevo ligado exclusivamente al paradigma de DT, ya que es un tipo de comunicación presente desde hace décadas en sistemas encargados de compartir información. En el caso de las comunicaciones V2V, realizadas exclusivamente en la dimensión virtual, son un reflejo de las comunicaciones P2P realizadas en la dimensión física. Adicionalmente a esto, este tipo de comunicaciones son capaces de romper las barreras temporales presentes en el mundo físico, permitiendo reflejar o simular largos periodos de comunicaciones o comportamientos con un escaso coste de tiempo.

2.1.2. Principios

2.1.2.1. Digital Twin vs simulaciones y emulaciones

Aunque el paradigma de *Digital Twin* puede parecer tener relación directa con los sistemas de simulación y emulación tradicionales, son enfoques distintos aplicados a la modelización y representación de sistemas o entidades, cumpliendo cada uno de ellos con un propósito específico.

En el caso de las simulaciones, el objetivo principal es la representación abstracta, basada en modelos, matemáticos o software, utilizadas para la evaluación de hipótesis, escenarios, o analizar el funcionamiento de un sistema en un entorno controlado, típicamente aislado, para proporcionar una visión simplificada y ayudar en la toma de decisiones o la identificación de patrones. Por su parte, las emulaciones se centran en replicar de forma más precisa el comportamiento y características de un elemento real, haciendo uso tanto de hardware como software. Siendo realmente útiles a la hora de probar y validar sistemas complejos como redes de comunicación, sistemas electrónicos, etc, tratando de generar un entorno de pruebas lo más realista posible. No obstante, los DTs van un paso más allá, ya que aúnan las características de simulación y emulación, representando de manera precisa la entidad física reflejando su comportamiento en tiempo real a través de los datos recopilados, permitiendo una característica no presente en las simulaciones o emulaciones, la realimentación por parte de los gemelos digitales, en tiempo real, hacia los gemelos físicos. Esta característica los convierte en herramientas esenciales para la gestión y toma de decisiones, la optimización, la seguridad, la detección temprana de problemas y la planificación.

2.1.2.2. Modelado

Uno de los principios fundamentales de los gemelos digitales es el modelado de la información [27], permitiendo la representación del mundo real en un entorno virtual. Principalmente se puede establecer una diferenciación en dos grandes tipos de modelos: de comportamiento y físicos. Los primeros tratan de obtener una caracterización del comportamiento de la entidad física a modelar, utilizando enfoques basados en la recopilación y análisis de datos. Este tipo de modelos son adecuados para sistemas complejos, como pueden ser las redes de comunicación. Los segundos persiguen modelar el comportamiento de la entidad física mediante ecuaciones matemáticas o modelos

computacionales. Esta aproximación, pese a ser más fiel a la realidad, suele estar asociada de manera específica al caso de uso, lo que limita su extrapolación.

Dentro de esa diferenciación inicial, el modelado se puede enfocar en varias áreas en base al requisito específico a satisfacer con la implementación del gemelo digital. En [28][\[ref\]](#) los autores hacen una diferenciación basada en optimización, seguridad, monitorización y predicción, despliegue y entrenamiento.

2.1.2.3. Arquitecturas

Pese a tener identificadas las características principales de la tecnología DT, para poder llevar a cabo su correcta implementación, es necesario identificar que requisitos básicos han de cumplir las arquitecturas encargadas de esa tarea. Actualmente existen multitud de trabajos que exponen arquitecturas de DT, sin embargo, la mayoría de ellas guardan una fuerte dependencia con el caso de uso o área de aplicación [29]. No obstante, todas estas propuestas comparten los enfoques propuestos por los autores en [30] [31], [32], donde se identifican las capas básicas que deben tener este tipo de arquitecturas:

- **Capa de adquisición de datos:** encargada de tanto de la recolección de datos para la fase de modelado del DT, como de la comunicación entre ambos mundos, físico y digital.
- **Capa de modelado y gestión de los datos:** encargada de la recepción de los datos para su homogeneización y posteriormente uso en la fase de modelado. Pudiendo utilizar formatos de homogeneización como JSON, XML o YANG.
- **Capa de procesamiento de información:** en ella se realiza el procesamiento de la información homogeneizada, a través de técnicas como Machine Learning, permite a los DT la obtención de conclusiones o favorecer la toma de decisiones.
- **Capa de consumo de resultados:** esta capa puede ser utilizada por diversos sistemas, como aplicaciones de gestión que eleven la información generada por el DT a sistemas avanzados o simplemente realimentar a la entidad física para tratar de optimizar o mejorar su funcionamiento.

Existe también un gran esfuerzo por tratar de llevar la tecnología DT a público en general, por ello, existen soluciones comerciales, reflejadas en la **Tabla 2**, ofrecen plataformas incipientes para el uso y desarrollo de la tecnología de gemelos digitales entre las que destacan:

Tabla 2. Soluciones comerciales basadas en Digital Twin

Plataforma	Descripción
AWS TwinMaker [33]	Framework Cloud para la creación de gemelos digitales multipropósito de Amazon
Azure Digital Twin [34]	Framework Cloud para la creación de gemelos digitales enfocado a IoT de Microsoft
Supply Chain Twin [35]	Solución Cloud para la creación de gemelos digitales enfocado a cadena de suministro de Google
BaseN [36]	Plataforma encargada de ofrecer Frameworks para gemelos digitales enfocado a IoT
Akila [37]	Plataforma cloud para gemelos digitales
Ansys [38]	Plataforma software para gemelos digitales basada en simulación
Aveva [39] Digital Twin Exchange [40]	Solución basada en gemelos digitales Plataforma para el intercambio de gemelos digitales ofrecida por IBM
IoT Digital Twin [41]	Framework cloud enfocado a gemelos digitales de IoT
GE Digital [23]	Paquete software basado en gemelos digitales como producto

2.1.3. Tecnologías habilitadoras

Tal y como se introdujo al inicio de este capítulo, la tecnología basada en gemelos digitales ha sido objeto de estudio y desarrollo en la literatura científica durante muchos años, pero su relevancia ha experimentado un marcado aumento en la última década. Este cambio significativo se debe en gran medida a la evolución tecnológica en campos diversos, que han actuado como catalizadores para el avance de los gemelos

digitales[42],[43], [44]. La conjunción de innovaciones en áreas como la conectividad, el procesamiento masivo de datos, la computación en la nube y las capacidades de la inteligencia artificial ha abierto un gran abanico de posibilidades para la implementación y el uso del paradigma basado en gemelos digitales en una amplia variedad de sectores. Estas tecnologías habilitadoras transforman la forma en la que concebimos, recolectamos y gestionamos la información, permitiendo llevar a cabo la implementación de réplicas digitales fieles de activos y procesos físicos. En este contexto, el IoT, la computación en la nube, el Big Data y la inteligencia artificial destacan como pilares fundamentales sobre los cuales es posible sustentar el paradigma de gemelos digitales. A continuación, se profundizará en cada una de estas tecnologías, poniendo en valor la importancia para el desarrollo e implementación de los gemelos digitales.

- **Internet of Things (IoT).** Concepto que referencia la interconexión de objetos heterogéneos a través de internet. Permitiendo tanto la comunicación entre dispositivos, como con sistemas más complejos, habilitando la recolección de información en tiempo real, la automatización de acciones, incluso la mejora en la eficiencia y optimización. Relacionado a Digital Twins, dispositivos IoT pueden actuar como sensores encargados de recolectar el estado de una entidad física y transmitirse al entorno digital y actuar como actuadores en el mundo físico a través de las decisiones o comandos emitidos desde el gemelo digital.
- **Cloud Computing.** Con la evolución hacia el uso de recursos compartidos en la nube, basados en tecnologías de virtualización, ofreciendo una alta demanda y disponibilidad, es un habilitador estratégico para la tecnología DT, permitiendo realizar tareas de procesamiento en tiempo real, adaptándose a las capacidades de cómputo de los gemelos frente a alta demanda.
- **Big Data.** El término hace referencia a la capacidad de recolección, almacenaje y análisis de grandes volúmenes de datos. En el contexto de Digital Twin, el Big Data es crucial, tanto para las fases de modelado como para alimentar al gemelo digital con información en tiempo real del sistema físico. Con técnicas de análisis basadas en Big Data es posible facilitar la toma de decisiones, la identificación de patrones, la detección de problemas o la optimización de operaciones.

- **Inteligencia Artificial.** Dentro de los distintos tipos de IA, la más relevante es el Machine Learning, el cual permite a los gemelos digitales aprender y adaptarse a partir de datos recolectados en tiempo real. A través de estos datos, es posible utilizar ML para identificar patrones, predecir eventos y optimizar el rendimiento de sistemas o procesos.

2.1.4. Trabajos relacionados

En la actualidad, encontramos que la aplicación del paradigma basado en gemelos digitales se extiende a áreas tan diversas como el Internet de las cosas (IoT), la industria 4.0, la atención médica y más tal y como se muestra en la [Figura 2.1.4](#). Esta amplia adopción pone de manifiesto la versatilidad y el potencial de este paradigma para llevar a cabo tareas como la optimización de procesos o la mejora en la toma de decisiones.

Un aspecto clave a tener en cuenta es que estas soluciones están intrínsecamente vinculadas a casos de uso específicos. Cada gemelo digital se diseña y adapta según las necesidades y características particulares de su aplicación. Este diseño específico permite que los gemelos digitales puedan abordar desafíos y problemáticas concretas de manera eficiente y efectiva.

En este contexto, es común ver en la literatura científica que el paradigma Digital Twins muchas veces se apoya en Frameworks específicos como los identificados en la [Figura 2.1.5](#). Estas infraestructuras ofrecen escalabilidad, flexibilidad y capacidad de procesamiento necesarias para gestionar grandes volúmenes de datos y soportar operaciones complejas en tiempo real. La nube proporciona un entorno propicio para el desarrollo, despliegue y operación de los gemelos digitales, facilitando su integración con otros sistemas y herramientas existentes.

Sin embargo, es importante destacar que la aplicación de este paradigma no siempre es directamente transferible o extrapolable entre diferentes dominios. Mientras que, en algunos sectores como la manufactura o la atención médica, las implementaciones pueden compartir ciertas similitudes y enfoques, en otros ámbitos como las redes de comunicación, surgen desafíos únicos que requieren consideraciones específicas.

Las redes de comunicación, con su complejidad y dinamismo inherentes, presentan un entorno único para la aplicación de Digital Twins. Aquí, la capacidad de modelar y simular sistemas en tiempo real, predecir comportamientos y optimizar el rendimiento se convierte en un desafío técnico y conceptual.

En resumen, si bien los Digital Twins ofrecen un potencial transformador en una variedad de sectores, su implementación exitosa requiere un enfoque específico y adaptado a las características y desafíos de cada dominio de aplicación. La convergencia de tecnologías como el IoT, la nube y los Digital Twins abre un amplio abanico de oportunidades para mejorar la eficiencia, la innovación y la toma de decisiones en un mundo cada vez más interconectado y digitalizado.

Tabla 3. Trabajos relacionados a la aplicabilidad del paradigma Digital Twin

Ámbito	Trabajo
Smart manufacturing	Digital Twin in Smart manufacturing [45] Digital Twin in Industry: State of the Art [46]
Healthcare	The Digital Twin Revolution in Healthcare [47] The concept of Digital Twin and Digital Shadow in Manufacturing [48] Digital Twin for Intelligent Context-Aware IoT Healthcare Systems [49] Digital Twin and Internet of Things – Current Standards Landscape [50]
Internet of things	IoT-based Mine Ventilation Control System Architecture with Digital Twin [51] Management of Digital Twin-Driven IoT using Federated Learning [52]
Industrial Internet of things	IoT-Based Digital Twin for Energy Cyber-Physical Systems: Design and Implementation [53] Industrial IoT lifecycle via Digital Twin [54]
Industry 4.0	Digital Twin as a Service (DTaaS) in Industry 4.0: An Architecture Reference Model [55] Digital Twin in industry 4.0 [56]
Unmanned aerial vehicles	Drone-Based AI and 3D Reconstruction for Digital Twin Augmentation [57] Simulation and Digital Twin Support for Managed Drone Applications [58] CitySim: A Drone-Based Vehicle Trajectory Dataset for Safety Research and Digital Twins [59] TwinPort: 5G drone-assisted data collection with digital twin for Smart seaports [60]
Multidisciplinary	Digital Twin and its applications: A Survey [61]
Others	City Digital Twin Potentials: A Review and Research Agenda [62] A Digital Twin for Smart Farming [63]

2.2. Gemelos Digitales de Red (NDT). Conceptos, principios y habilitadores.

Tal y como se ha introducido en la sección anterior, el paradigma de gemelos digitales puede ser aplicado a una infinidad de áreas, siendo una de ellas las redes de comunicación. Este tipo de redes, debido al aumento de negocios que hacen uso de ellas, la escala o la complejidad de estas, son un candidato ideal para utilizar la tecnología de gemelos digitales con el objetivo de mejorar las soluciones existentes. Desde principios de 2017 se ha estado investigando en la aplicación de este paradigma a las redes de comunicación, a continuación, se citan algunos trabajos relevantes en este campo.

En el ámbito académico, los autores en [64] aplican el paradigma de gemelos digitales en redes *Multi-access Edge Computing* (MEC) 5G, en este contexto, el gemelo digital se utiliza de manera offline con el objetivo de entrar algoritmos a través de entrenamiento por refuerzo utilizados para la optimización en la asignación de recursos y el ahorro energético de los componentes hardware asociados. En [65], los autores establecen una red *Edge* en forma de gemelo digital, en la cual despliegan un servidor, en forma de gemelo digital, que evalúa el estado del servidor físico, de esta forma la red *Edge* gemela digital proporciona datos para entrenar las estrategias de *Offloading*. Otro caso de uso es el presentado en 2021 por los autores en [66], discutiendo sobre como desplegar gemelos digitales aplicados a redes 5G complejas. Debido a las características especiales de este tipo de aplicación a redes de comunicación, es necesario desarrollar plataformas que difieren mucho de algunos framework o plataformas presentadas en la sección anterior, por ello, en [66] los autores presentan una plataforma orientada a la gestión automatizada e inteligente de las redes de centros de datos, proponiendo la simplificación de los flujos de trabajo asociados a la gestión de los servicios de red.

No obstante, aunque existen esfuerzos para la aplicación de la tecnología basada en gemelos digitales en el campo de las redes de comunicación, la investigación para la implantación de esta tecnología aún está en etapas tempranas. Actualmente la aplicabilidad de esta tecnología se centra en escenarios concretos, como puede ser la optimización de redes, donde el gemelo digital es utilizado como una herramienta de simulación encargada de resolver problemas ligados a la operación y el mantenimiento de la red. Pese a ello, esta tecnología debe ser vista como un todo, integrado con las redes físicas, permitiendo abarcar todo el ciclo de vida de dichas redes, permitiendo en un

futuro ser aplicado a tareas de planificación, despliegue, mantenimiento y optimización, mejorando el nivel de automatización e inteligencia de la red.

2.2.1. Conceptos

Con la aplicación del paradigma de gemelos digitales a redes de comunicaciones surgen dos conceptos asociados al caso de uso. El primero es el concepto de *Network Digital Twin* (NDT). Un NDT es esencialmente una representación digital altamente detallada de una red de comunicaciones, que abarca tanto la infraestructura física como los aspectos relacionados con el tráfico y el funcionamiento de los dispositivos y componentes que componen la red.

En su núcleo, el NDT se basa en la idea de utilizar tecnologías habilitadoras, como el Internet de las Cosas (IoT), la inteligencia artificial (IA) y el aprendizaje automático, para recopilar y procesar datos en tiempo real de todos los elementos de una red. Estos datos se utilizan para crear una representación digital en constante actualización, lo que permite una supervisión y análisis detallados de la red en tiempo real.

La principal ventaja del NDT es que ofrece a las organizaciones una visión completa y precisa la red, lo que permite tomar decisiones en base a los datos y realizar una predicción de problemas antes de que ocurran. Por ejemplo, se pueden identificar cuellos de botella en el tráfico de la red, predecir posibles fallas en los componentes de la infraestructura u optimizar la asignación de recursos. Adicionalmente, el NDT no se limita únicamente a la supervisión y el análisis; también es útil para la planificación y el diseño de redes. Las organizaciones pueden simular diferentes escenarios y estrategias de implementación antes de realizar inversiones costosas en infraestructura. Esto permite una planificación más eficiente y una implementación más ágil de nuevas tecnologías y servicios.

El segundo concepto de *Mobile Network Digital Twin* (MNDT), es una aplicabilidad del primero a un campo aún más específico, el cual representa una evolución en la gestión de las redes de comunicaciones móviles. A medida que la demanda de conectividad móvil aumenta de manera constante y con la transición hacia tecnologías avanzadas como el 5G, surge la necesidad de una mayor supervisión y optimización de las redes móviles.

Este enfoque tecnológico permite la creación de una representación digital altamente precisa y en tiempo real de una red de comunicaciones móviles completa. Esto abarca desde las estaciones base, las antenas y los nodos de acceso hasta la infraestructura de backhaul, pasando por el tráfico de datos y la experiencia del usuario final. La clave radica en la recopilación de datos en tiempo real procedentes de todos estos elementos de la red, lo que permite una supervisión y análisis continuos.

Uno de los principales beneficios del MNDT de redes móviles es su capacidad para proporcionar información detallada sobre la calidad del servicio, el rendimiento de la red y la experiencia del usuario. Los operadores de redes móviles pueden utilizar esta información para realizar una planificación más precisa y eficiente para el despliegue de nuevas tecnologías móviles, como el 5G, y garantiza la optimización en el uso de los recursos.

2.2.2. Principios

Actualmente, el IETF define el concepto de un Gemelo Digital a través del borrador actual "Red de Gemelos Digitales: Conceptos y Arquitectura de Referencia" [67], como una representación en tiempo real de una entidad o elemento físico en un entorno digital. Esta entidad digital tiene varias propiedades, incluyendo interacción en tiempo real, operación iterativa y optimización de procesos, abarca todo el ciclo de vida y construye una red integral basada en datos.

Aunque no hay una definición formal del concepto de Gemelo Digital de Red, el borrador del IETF lo define como una representación virtual de la red física que se utiliza para el análisis, diagnóstico, emulación y control de la red. Como se muestra en la **Figura 1** incluye cuatro elementos clave: Datos, Mapeo, Modelos e Interfaces.

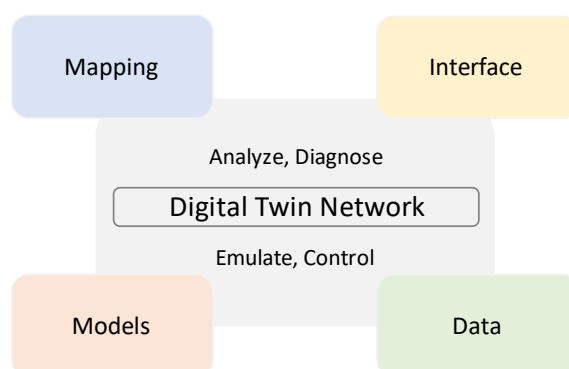


Figura 1. Pilares gemelo digital de red

- **Datos.** Un Gemelo Digital de Red debe mantener datos históricos y/o datos en tiempo real (datos de configuración, datos de estado operativo, datos de topología, datos de trazado, datos métricos, datos de procesos, etc.) sobre su gemelo real (es decir, la red física) que son necesarios para que los modelos representen y comprendan los estados y comportamientos reales del gemelo.
- **Modelos.** Técnicas que implican recopilar datos de una o varias fuentes en el gemelo del mundo real y desarrollar una representación integral de los datos (por ejemplo, sistema, entidad, proceso) utilizando modelos específicos. Estos modelos se utilizan como base para la emulación y el diagnóstico para representar la dinámica de la red física y los elementos de operación, y generar datos para la toma de decisiones.
- **Interfaces.** Las interfaces estandarizadas pueden garantizar la interoperabilidad del Gemelo Digital de Red. Hay dos tipos principales de interfaces:
 - La interfaz entre la plataforma del Gemelo Digital de Red y la infraestructura de la red física.
 - La interfaz entre la plataforma del Gemelo Digital de Red y las aplicaciones.
- **Mapeo.** Se utiliza para identificar el Gemelo Digital y las entidades subyacentes, y establecer una relación interactiva en tiempo real entre la red física y el gemelo de la red o entre dos gemelos de red. El mapeo puede ser:
 - Uno a uno (emparejamiento, vertical): Sincronización entre una red física y su gemelo virtual con flujos continuos.
 - Uno a muchos (acoplamiento, horizontal): Sincronización entre gemelos virtuales con intercambio de datos ocasional.

Ligada a la definición del concepto NDT propuesto por el IETF, proponen una arquitectura que tiene como objetivo ser la referencia para la implementación de la tecnología DT aplicada al caso de uso específico de redes de comunicación. Esta propuesta de arquitectura está conformada por tres capas principales como se ve en la **Figura 2**.

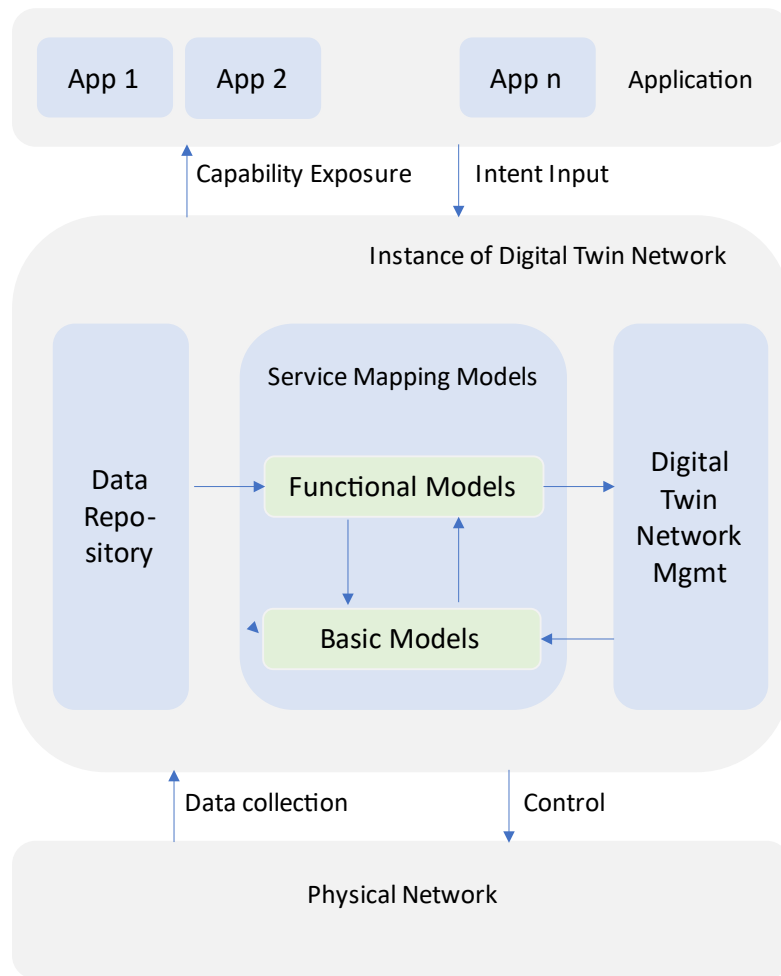


Figura 2. Arquitectura referencia NDT propuesta por el IETF

- **Red real.** Entendida como un conjunto total o parcial de elementos de una red, en la cual se realiza intercambios de datos de red y mensajes de control contra el gemelo digital de dicha red haciendo uso de interfaces de control entre ambos gemelos. La red puede abarcar uno o varios dominios administrados y puede incluir tanto entidades físicas como entidades virtualizadas (conmutadores virtuales, controladores SDN, VNFs, etc)
- **Capa de gemelo digital.** Conformada por tres subsistemas independientes, repositorio de datos, modelado de servicio y gestión de red de gemelos digitales. Siendo el primero el encargado de recopilar y almacenar información en tiempo real de la red real, el segundo encargado de la obtención de modelos básicos basados asociados a topología, comportamiento, información de entorno, estado operacional, etc y modelos funcionales utilizados para el análisis de red, predicción, tareas de diagnóstico, etc. Por último, el subsistema de gestión se encarga

de manejar el ciclo de vida de las entidades digitales, monitorizando el consumo de recursos, el comportamiento, la topología, o la seguridad. Estos subsistemas pueden ubicarse en un único dominio administrativo de red y proporcionar el servicio a una aplicación superior o pueden integrarse en cada dominio administrativo de la red y coordinarse entre ellos para ofrecer servicios a la capa de aplicación.

- **Capa de aplicación.** Alberga aplicaciones asociadas a la administración, mantenimiento u operación de la red, los cuales pueden hacer uso de la información generada en el gemelo digital a través de los datos recolectados en tiempo real de la red real.

Esta propuesta de uso de gemelos digitales ligada a las redes de comunicación plantea una serie de beneficios entre los que destacan:

- **Optimización en el coste de operación.** Tradicionalmente los operadores han utilizado conceptos como las maquetas de red, o entornos de laboratorios para realizar pruebas de optimización, las cuales posteriormente eran trasladadas a la red real, se pone de manifiesto la complejidad a la hora de operar redes a gran escala. Este tipo de evaluación tradicional aumenta el gasto '*Operational Expenditures*' (OPEX) del operador de red, el cual previsiblemente no se verá tan afectado con el uso de la tecnología basada en NDT.
- **Optimización en la toma de decisiones.** Dado que la operación y gestión tradicional en las redes se basa en implementar y gestionar los servicios en ejecución, tratando de minimizar el impacto en los usuarios finales, no facilita las tareas de mantenimiento preventivo. El NDT combinado con la adquisición de datos en tiempo real, el *Big Data* y la IA, permitirían realizar una evaluación de estado real, prediciendo tendencias futuras para así poder realizar un mantenimiento preventivo.
- **Cumplimiento normativo asociado a la privacidad.** Con las nuevas leyes en materia de protección de datos, aumenta la complejidad a la hora de gestionar las redes por parte de los proveedores, dado que para muchas de las tareas ligadas a la operación es necesario la inspección del tráfico cursado por la red. En el caso de NDT, a través de la capacidad de los modelos, el análisis inteligente basado en datos permite anonimizar completamente al usuario, permitiendo cumplir los

requisitos de privacidad y simplificar el uso de técnicas de preservación de la privacidad como la anonimización de los datos.

2.2.3. Habilitadores tecnológicos

Dado que el concepto de NDT surge de la aplicación del concepto DT a redes de comunicación, este se ve igualmente potenciado por habilitadores tecnológicos vistos anteriormente como el IoT, la computación en la nube, *Big Data* o la inteligencia artificial. No obstante, existen otras tecnologías que se plantean como habilitadores de suma importancia en ecosistemas NDT, entre las que podemos diferenciar a la tecnología de *Software Defined Network* (SDN), aplicada principalmente al plano de datos, la tecnología de *Network Function Virtualized* (NFV) aplicada al plano de control. Debido a la relevancia de estas dos últimas, se desarrollarán de manera más extensa en puntos posteriores. En la **Tabla 4** se muestran trabajos enfocados a la aplicación de estas tecnologías al caso de uso concreto de gemelos digitales de red.

Tabla 4. SDN - NFV aplicado a Digital Twin

Trabajo	Resumen
Building a Digital Twin Network of SDN using Knowledge Graphs [68]	Implementación de un DTN basada en SDN sobre la cual se aplica un Knowledge Graph (KG) para modelar los datos. La DTN persigue la monitorización y la verificación de cambios en la red SDN sin impactar con la red real.
Suitability of SDN and MEC to facilitate digital twin communication over LTE-A [69]	Estudio que plantea la integración de tecnologías SDN y MEC en redes móviles LTE-A con el fin de dar soporte a aplicaciones basadas en Gemelos Digitales, poniendo énfasis en la optimización en términos de calidad de servicio.
Digital Twin-driven SDN for Smart grid [70]	Se propone la integración de la tecnología basada en gemelos digitales, junto a IoT, SDN, Deep Learning y Blockchain, para mejorar la seguridad y eficiencia en redes Smart Grid.
When Digital Twin Meets Network Softwarization in the Industrial IoT [71]	En el contexto de IoT Industrial (IIoT) se presenta el concepto de gemelos digitales, unido a SDN para poder establecer conexiones entre los distintos elementos en bucle cerrado, permitiendo la gestión de todo el ciclo de vida de la red.
VNF Migration in Digital Twin Network	Se propone el uso de arquitecturas de red basadas en gemelos digitales con el fin de ser utilizado como testbed para reducir los riesgos de

for NFV Environment [72]	validación durante las fases de entrenamiento y predicción de algoritmos basados en aprendizaje profundo
Diseño de un ecosistema DTN basado en NFV para el análisis de ciberseguridad en redes 5G [73]	Diseño inicial de un framework completo capaz de desplegar un ecosistema DTN para escenarios de red basados en la tecnología NFV con el objetivo de utilizar la DTN como testbed para la realización de pruebas de seguridad sobre redes 5G

Debido a las características del modelado de redes de comunicación en forma de gemelo digital, existen habilitadores tecnológicos ligados a la recolección de datos (SNMP, NetConf, IPFix, ALTO, etc), asociados al modelado (NS-3, Mininet, EVE-NG, GNS-3, etc) [66] [74] [75] o relacionados con las interfaces de comunicación (XMPP, HTTP/3.0, etc)

2.2.4. Trabajos relacionados

Durante la última década, la comunidad investigadora se ha mostrado cada vez más interesada en tecnologías basadas en gemelos digitales. Como se indica en un extenso trabajo, los autores en [76], esta tecnología se basa en la representación digital de entidades físicas, así como su conexión e interacción entre ambos mundos. Actualmente, existen diferentes enfoques para el uso de esta tecnología, dependiendo en gran medida, del dominio de aplicación, prestando especial atención a las etapas de modelado de los gemelos.

Pese a que la literatura sobre el concepto general de gemelo digital es extensa, el enfoque de esta tecnología aplicada a redes está dando lugar al concepto de gemelos digitales de red.

Los autores en [77] remarcan las tecnologías clave asociadas a áreas como las comunicaciones, la computación física, el modelado o la computación en la nube y en el borde de la red. Al igual que los autores en [76], identifican varias áreas de aplicación, entre las que destacan la nueva generación de redes 5G y 6G.

En [28], los autores llevan a cabo un exhaustivo estudio el cual aborda las tareas de Diseño, modelado e implementación de gemelos digitales. Este trabajo centra la fase de Diseño del gemelo basándose en requisitos funcionales como la optimización, la seguridad, el monitoreo o el desempeño. Pese a que el modelado basado en datos se menciona dentro de una amplia clasificación de modelos que se pueden aplicar al caso

específico de redes de comunicación, el requisito funcional asociado a NDT no se aborda explícitamente. Por ello, se identifica como un desafío abierto, la caracterización, tanto topológica como de comportamiento de los elementos presentes en una red 5G. La cual no presenta los mismos requisitos que, por ejemplo, un proceso industrial, desde la perspectiva del modelado, la recopilación de datos o el propio despliegue de estos gemelos digitales de red, siendo necesario aplicar modelos basados en la interrelación de datos (físicos, lógicos e incluso la inferencia de nuevos datos a partir de tecnologías basadas en ontologías).

Tras el análisis del estado del arte asociado a gemelos digitales, para el caso específico de aplicación a redes de comunicaciones, se puede concluir que los Frameworks actuales como Eclipse Ditto [78], FIWARE [79], AWS [33], Azure [34], Google [35] o Albo no son extrapolables, lo que implica la necesidad de establecer nuevos framework que permitan la implementación de la tecnología de gemelos digitales aplicadas a redes de comunicación.

Volviendo al caso de NDT, los autores en [80] establecen la relación del concepto con la aplicación específica en redes de operador. En este trabajo, los autores destacan las ventajas de esta tecnología en ámbitos como la detección temprana de fallos, la planificación de redes, la detección de anomalías, la formación y capacitación. Adicionalmente, proponen una arquitectura de referencia para NDTs formada por tres elementos. El primero de ellos es el modelo de gemelo digital de red, en el cual se crea un gemelo digital basado en aprendizaje profundo, alimentado con datos de tráfico, topología, enrutamiento, políticas de programación, etc y es capaz de proporcionar resultados como porcentaje de utilización, retraso en las comunicaciones, pérdida de paquetes, etc. Estos resultados pueden replicarse en la red de producción para evaluar los *Service Level Agreement* (SLA). En el trabajo se destaca la importancia de utilizar tecnologías de telemetría o monitorización de datos, debido a que, en el contexto de gemelos digitales, la información se intercambia entre ambos mundos, digital y real. Finalmente establecen la importancia de la captura de datos, tanto en términos de almacenamiento como en la necesidad de un formato estándar para la captura de datos heterogéneos.

Siguiendo el enfoque de recopilación de datos introducido en la referencia [80], los autores en [81] presentan un algoritmo para la recopilación de datos a través de una red física. Los datos recopilados son procesados mediante un algoritmo de enriquecimiento basado en el lenguaje de representación semántica *Web Ontology Language* (OWL), propuesto por el *World Wide Web Consortium*. Adicionalmente a esto, presentan una arquitectura NDT similar a la propuesta por el IETF en el borrador [67], que identifica cuatro capas diferenciadas, capa de red física, capa de gemelo digital, capa de aplicación de red y capa de datos.

Como se mencionó al inicio de este apartado, una de las áreas de aplicación de la tecnología basada en gemelos digitales son las redes de comunicación. Los autores en [82] señalan la importancia de esta tecnología en combinación con 5G y la inteligencia artificial. Este trabajo se centra en el uso de esta tecnología para proporcionar un acceso fácil y rentable a la tecnología 5G, permitiendo desarrollos flexibles y replicables en el tiempo, permitiendo llevar a cabo el modelado proactivo del tráfico y ayudando a acelerar la investigación y el desarrollo de nuevos servicios basados en redes de próxima generación. Además, en el ámbito de redes 5G, los autores en [83] abordan la problemática asociada al concepto de “*network slicing*”, el cual permitiría optimizar el rendimiento de la red gracias a la tecnología DT. Sin embargo, debido a la interconexión inherente a la tecnología de DT, los autores proponen modelos DT basados en *Graph Neural Networks* (GNN) que permitirían descubrir las relaciones y dependencias entre los diferentes slices de red, la utilización de recursos y la infraestructura física asociada al gemelo físico.

Así como la tecnología DT está despertando interés en su aplicabilidad a 5G, también lo está haciendo en trabajos emergentes sobre redes 6G. Los autores en [84] proponen el uso de la tecnología de gemelos digitales en el borde de la red 6G, logrando así una combinación entre el *Multi-access Edge Computing* (MEC) y el gemelo digital, lo cual permitiría monitorizar en tiempo real toda la red. El MEC combinado con 6G permitiría alimentar con datos un módulo de toma de decisiones capaz de realizar acciones en la red física de forma automatizada con el objetivo de optimizar la gestión de red.

A pesar de que los trabajos presentados previamente se centran en propuestas de arquitectura NDT o en casos de uso más específicos, hay dos trabajos [85] y [86] en los cuales los autores abordan un concepto interesante, el cual no está completamente

desarrollado en estudios relacionados, referente a una metodología capaz de modelar e implementar estos gemelos digitales. En ellos, se pone de manifiesto que las fases de adquisición de datos, necesarios para el modelado del gemelo, presentan una enorme variabilidad a la hora de llevar a cabo una implementación técnica funcional, ya que existe una fuerte dependencia del elemento físico que se está modelando. Por ello, en estos trabajos, se puede concluir que los modelos presentados son perfectamente adecuados para procesos industriales, pero si la naturaleza de datos cambia, sería difícil utilizarlos para modelar gemelos digitales de redes.

Finalmente, tras revisar los trabajos anteriormente citados y hacer referencia a los puntos clave de este tipo de tecnologías, la comunicación bidireccional y la integración en el ecosistema DT de redes o elementos en producción, se presta especial atención al trabajo presentado por los autores en [87], donde se abordan requisitos como la seguridad en la sincronización, la latencia en las comunicaciones, el control de acceso, el aislamiento de red o la resiliencia. Además, proponen una arquitectura para la inclusión de seguridad, considerando tanto la seguridad de ambos gemelos como la seguridad necesaria en la comunicación entre ellos.

Adicionalmente, en [28], los autores remarcan los desafíos abiertos relacionados con el Diseño Seguro de gemelos digitales, centrándose en la confiabilidad de los datos intercambiados entre gemelos. En el caso específico del modelado de gemelos digitales para redes de comunicación 6G, este aspecto se puede cubrir aplicando mecanismos de bastionado actuales que se aplican a entornos de tecnología de la información y comunicaciones (TIC). Sin embargo, debido a la importancia crítica de la intercomunicación entre gemelos, existen trabajos como [88] y [89] que proponen el uso de *Post-Quantum Cryptography* (PQC) sobre la capa de transporte (TLS) con el objetivo de crear una capa de comunicación más robusta.

2.2.5. Retos

El paradigma general DT, según se establece en el artículo [90], conlleva una serie de retos independientemente del campo de aplicación de la tecnología, siendo los más relevantes:

- La adquisición y procesado de datos
- El modelado de alta fidelidad

- La comunicación bidireccional en tiempo real
- Unificación de plataformas y herramientas
- Tecnologías para la interconexión entre ambos mundos

No obstante, la aplicación de DT al caso concreto de las redes de comunicación plantea una serie de retos adicionales debido a las características únicas de las mismas, entre los que se destacan:

- Redes a gran escala.
- Interoperabilidad
- Dificultades a la hora de modelar las redes
- Requerimientos de tiempo real
- Metodologías que estandaricen el proceso de creación del gemelo digital, así como la definición de módulos que permitan llevar a cabo la implementación completa del paradigma.
- Mecanismos que garanticen el intercambio seguro de los datos en términos de confidencialidad, integridad y autenticidad de la información intercambiada entre los gemelos.

2.3. Tecnologías de soporte al NDT

2.3.1. Tecnología SDN

Las redes SDN (Software-Defined Networking) [91] se caracterizan por un nuevo enfoque de la programabilidad de las redes; con el uso de controladores basados en software, que permiten controlar y gestionar de manera centralizada y dinámica el comportamiento de las redes a través de interfaces de programación (APIs). Evitando al administrador de red gestionar dichos servicios a bajo nivel, todo esto se consigue mediante la separación del plano de control del plano de datos [92].

Estas redes presentan una arquitectura flexible basada en tres capas principales [91] [92] [93] como se muestra en la **Figura 3**:

- **Capa de aplicación.** Residen las aplicaciones y servicios que definen el comportamiento de la red. Estas aplicaciones permiten gestionar el rendimiento de los recursos de red en función de los requisitos específicos. Para lograr esto, se

conectan con el plano del controlador SDN a través de la interfaz norte (NorthBound Interface). Los programas en la capa de aplicación acceden a una vista global y abstracta de los recursos de red, lo cual les permite tomar decisiones internas basadas en esta información.

- **Capa de control.** Responsable de tomar decisiones sobre cómo deben ser reenviados los paquetes por los dispositivos de red y de comunicar estas decisiones a los dispositivos de la red. Los controladores SDN son los encargados de gestionar la capa de control y ajustar las tablas de flujo en los dispositivos de red, basándose en la topología y las características de la red y los requisitos de calidad de servicio definidos.
- **Capa de infraestructura.** Alberga los dispositivos físicos de red, como switches, routers y enlaces de comunicación. Estos dispositivos se comunican con los controladores SDN a través de la interfaz sur (*SouthBound Interface*), recibiendo instrucciones sobre cómo manejar el tráfico. Esto simplifica los dispositivos de infraestructura, ya que solo necesitan seguir las directivas proporcionadas por los controladores en lugar de tener que manejar múltiples estándares de protocolo.

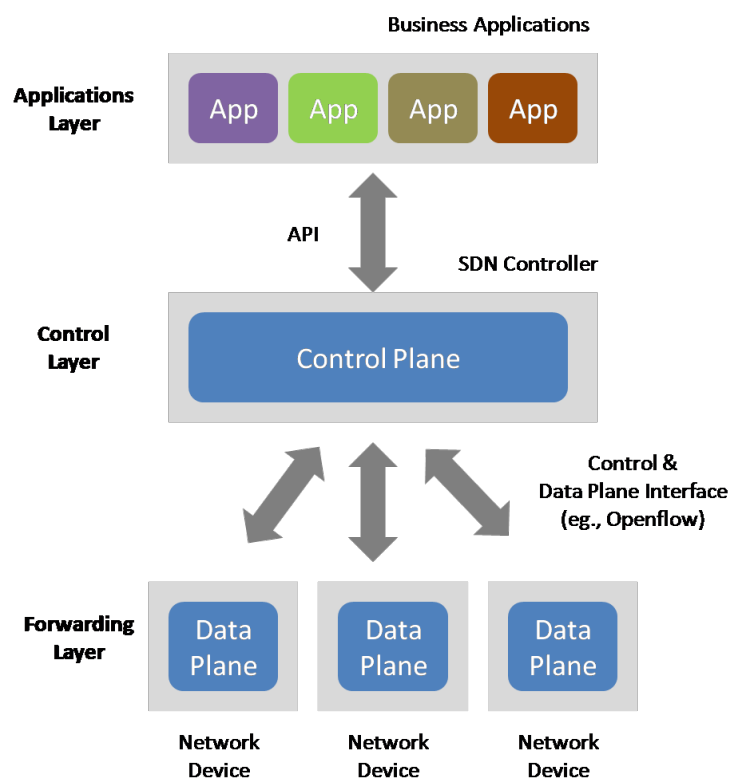


Figura 3. Capas de la arquitectura SDN

La arquitectura de redes las definidas por software ofrece ventajas en términos de control, flexibilidad y seguridad. La utilización de controladores y la separación de los planos de datos y de control permiten gestionar el tráfico de la red de forma programática, simplificando la configuración y permitiendo una mayor agilidad en la gestión. Además, es posible contar con una infraestructura de red personalizable, donde los administradores pueden asignar recursos virtuales y optimizar el flujo de datos en tiempo real. En cuanto a la seguridad, las SDN ofrecen una mayor visibilidad y control, permitiendo una respuesta rápida a las amenazas y la implementación de políticas granulares.

2.3.2. Tecnología NFV

La virtualización de funciones de red, NFV por sus siglas en inglés, es una tecnología de arquitectura de red cuyo objetivo principal es desacoplar los equipos físicos que forman la red de las funciones que se ejecutan en ellos. De esta forma, NFV rompe con el diseño de las redes tradicionales permitiendo descomponer un servicio de red (NS) en funciones de red (NF), tales como routers o firewalls, que se implementan en un entorno virtualizado. Es decir, lo que antes requería un hardware muy específico y costoso ahora se puede desplegar como una función de red completamente basada en software que se ejecuta sobre máquinas virtuales [94].

Ello ha supuesto un cambio en la forma de diseñar, desplegar y gestionar las redes de comunicaciones, que cada vez demandan mayor flexibilidad y control en la red. Los principales beneficios para los operadores de telecomunicaciones se resumen en [95]:

- Reducción de costes: reducción tanto el '*Capital Expenditures*' (CAPEX) como el OPEX gracias al ahorro de gastos en hardware específico, consumo energético y otros temas logísticos.
- Uso más eficiente de los recursos.
- Mayor flexibilidad y escalabilidad de la red.
- Mayor facilidad y agilidad para la introducción de nuevos servicios.
- Centralización de la gestión y reducción considerable del tiempo dedicado a diseño de la red.

2.3.2.1. ETSI-NFV

Desde octubre de 2012 el *European Telecommunications Standards Institute* (ETSI) y concretamente el *NFV Industry Specification Group*, lidera la estandarización de NFV. Su misión es alcanzar un consenso en el sector sobre los requisitos técnicos y de negocio de NFV, y acordar enfoques comunes para alcanzar estos requisitos. Los resultados se publican abiertamente y se comparten con los principales organismos de estandarización, foros industriales y consorcios para fomentar un esfuerzo de colaboración más amplio [96].

En la **Figura 4** se muestra arquitectura propuesta por ETSI. A alto nivel está formada por tres bloques:

- **Virtualized Network Function (VNF)**: implementación software de una función de red que se ejecuta sobre una o varias máquinas virtuales proporcionadas por la NFVI.
- **Network Function Virtualization Infrastructure (NFVI)**: es el soporte para la ejecución de las VNF. Incluye todos los recursos necesarios tanto hardware (computación, almacenamiento y red) como software (capa de virtualización y los recursos virtualizados).
- **Network Function Virtualization Management and Orchestration (NFV MANO)**: bloque responsable de la orquestación y gestión del ciclo de vida tanto de los recursos hardware y software proporcionados por la infraestructura de NFV como de las VNF. Es decir, controla todos los elementos que forman la arquitectura de NFV.

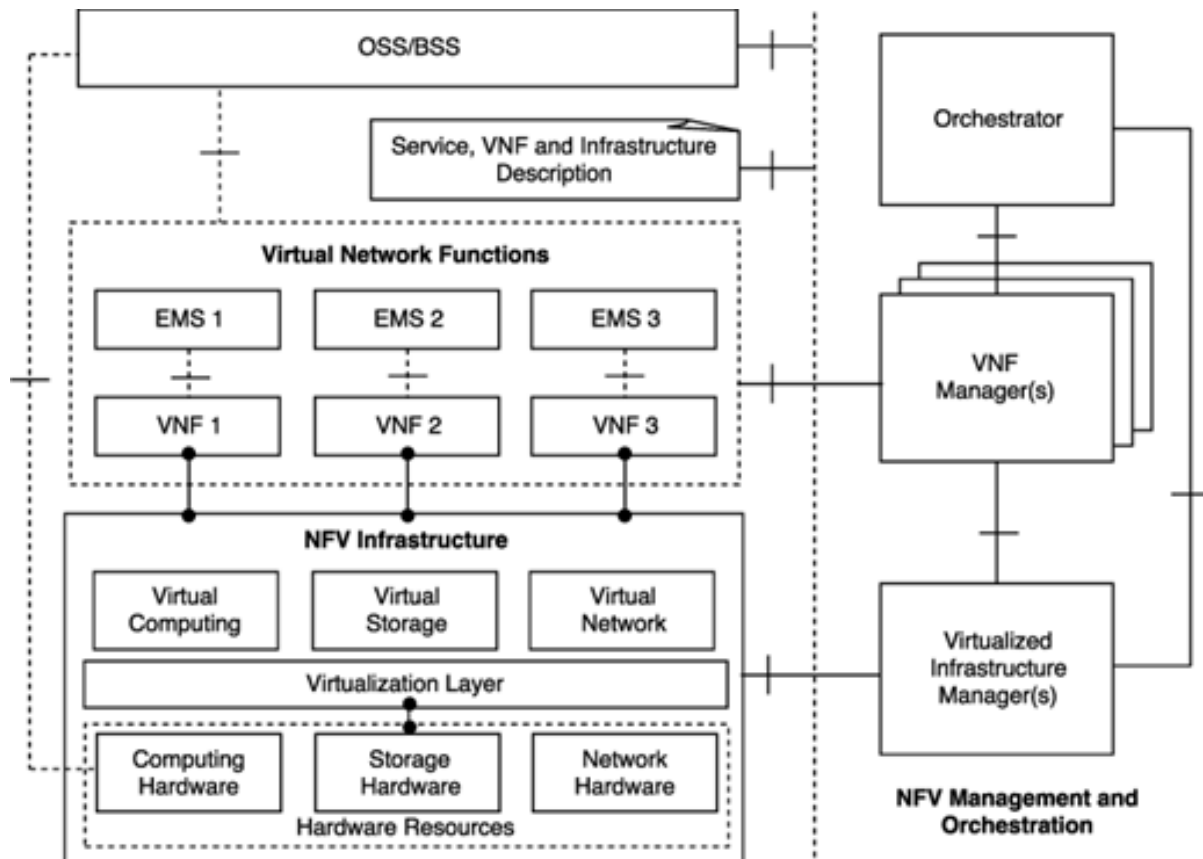


Figura 4. Arquitectura NFV propuesta por ETSI

2.3.2.2. NFV-MANO

El elemento NFV-MANO [97] es considerado dentro de la arquitectura NFV como un elemento fundamental del sistema, encargado de los aspectos de gestión y orquestación de todo el conjunto de VNFs incluyendo tanto la parte computacional, redes, gestión de recursos virtuales, así como la supervisión del ciclo de vida de la red.

En la **Figura 5** se puede observar la interacción de los diferentes componentes del sistema:

- VNF Manager, encargado del control de cada VNF, así como de administrar el catálogo de VNF.
- La conexión al Virtualized Infrastructure Manager (VIM) se encarga de todo lo relacionado con la infraestructura de virtualización de funciones de red, incluyendo las tareas de gestión y monitorización de recursos hardware asociados.

- El catálogo NS, la instancia NFV, el catálogo VNF y los recursos NFVI son repositorios que contiene metadatos sobre los diferentes componentes que pueden ser utilizados para la creación del servicio NFV.
- El orquestador NFV es el elemento encargado de la coordinación y gestión de los diferentes repositorios que proporcionan recursos para la creación de servicios de red.
- Conexión entre NFVI y VIM encargado de tareas como:
 - Gestión de fallos y rendimiento de hardware, software y recursos virtuales.
 - Gestión del ciclo de vida de los recursos virtuales en el dominio NFVI.
 - Monitorización del inventario de máquinas virtuales y recursos físicos asociados a las mismas.
 - Gestión de la emulación de recursos físicos en recursos virtuales.

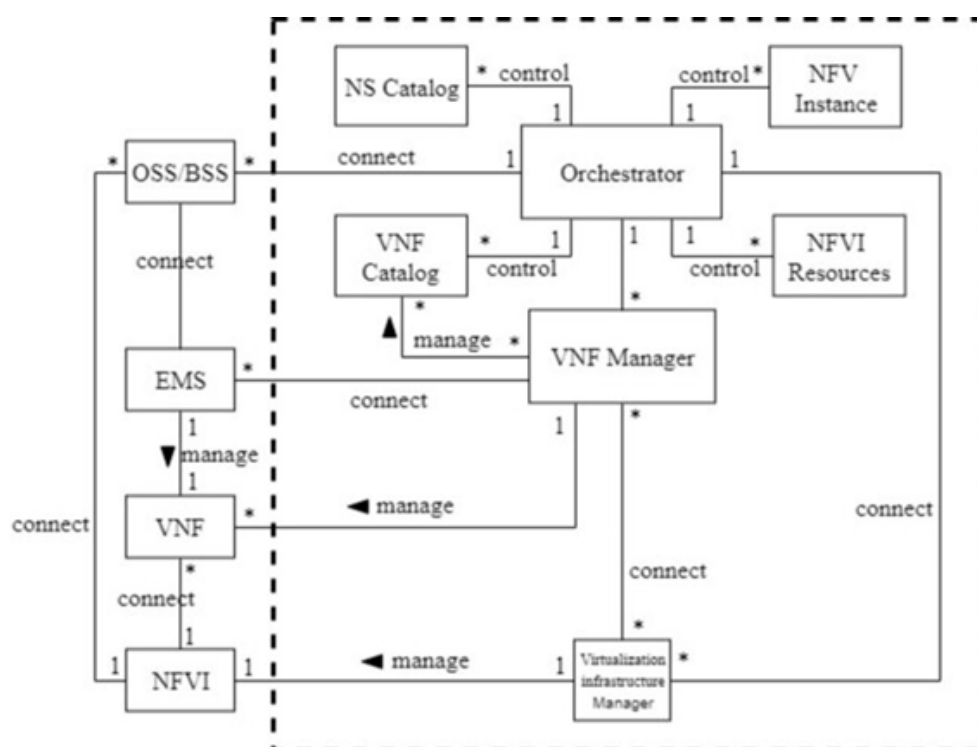


Figura 5. Diagrama de interconexión de elementos dentro de la unidad NFV-MANO

2.3.2.2.1 OpenSourceMANO

Dentro de las aproximaciones de MANO, destaca la solución basada en software Open Source, Open Source MANO (OSM) la cual es una iniciativa liderada por ETSI que para la

gestión y orquestación de NFV. Complementa el trabajo de ETSI NFV proporcionando realimentación práctica, y pretende incrementar la interoperabilidad de distintas soluciones. Asimismo, se aprovecha de las ventajas que supone fusionar la estandarización con enfoques de código abierto. Entre ellas destacan una mayor cantidad y diversidad de contribuidores y desarrolladores que tienen la posibilidad de participar, y maximizar la innovación, eficiencia y tiempo de lanzamiento [98].

Su arquitectura está completamente alineada con la arquitectura de referencia de ETSI, cubriendo los dos principales bloques de ETSI MANO, el *NFV Orchestrator* y *NFV Manager* descritos en la sección anterior. Para el tercer bloque, el VIM, existen varias posibilidades con los que OSM puede interactuar. El flujo de interacción es el mostrado en la **Figura 6**.

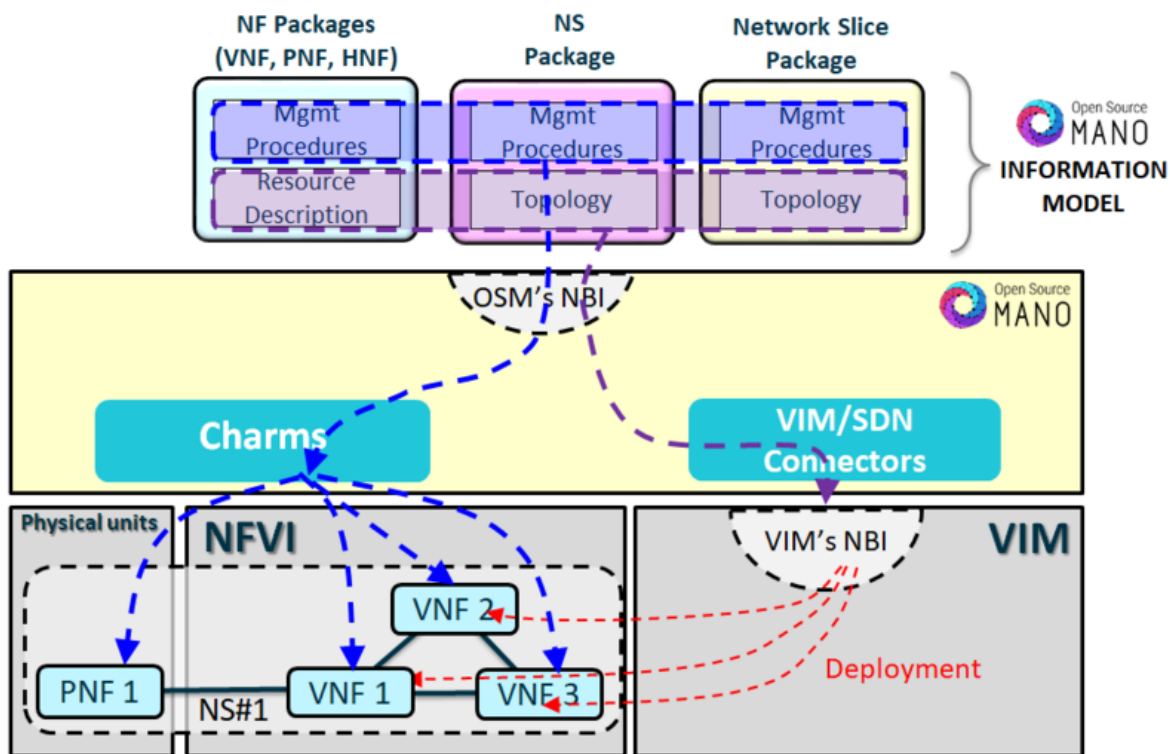


Figura 6. Interacción entre componentes OSM

Algunas de las características generales más relevantes de OSM son:

- Software de código abierto y disponible para todo el mundo. Además de fácil de usar gracias a su documentación y soporte por parte de la comunidad.
- Modelo de información bien conocido y alineado con ETSI NFV capaz de modelar y automatizar el ciclo de vida completo de funciones, servicios y segmentos de red

(Network Slices, NSI) desde su instanciación y configuración inicial hasta su operación y monitoreo diarios.

- Apto para todo tipo de funciones de red: virtuales (VNF), físicas (PNF) o híbridas (HNF).
- Independiente del VIM: OSM necesita los recursos de un VIM para desplegar las VNF y enlaces virtuales que las conectan. No obstante, es completamente independiente de esta infraestructura y permite que un mismo modelo se pueda usar para desplegar elementos en una gran variedad de VIM distintos (OpenStack, OpenVIM, AWS, Microsoft Azure, Kubernetes, etc.).
- *Northbound Interface* (NBI) unificada, basada en NFV SOL005, que permite el funcionamiento completo del sistema y ofrece como servicio todas las abstracciones necesarias para permitir el control, operación y supervisión del ciclo de vida de NS/NSI por sistemas del cliente, evitando la exposición de detalles innecesarios de sus elementos constitutivos.

Arquitectura interna de OSM

Para entender mejor el funcionamiento de OSM se va a detallar su arquitectura, una arquitectura que es, de forma innata, modular y basada en modelos. Además, es una arquitectura en continua evolución que busca adaptarse a la creciente funcionalidad [99]. En la **Figura 7** se muestran los módulos que forman la arquitectura original de OSM y más abajo se hace un breve resume de las funcionalidades de cada uno de ellos. Cada módulo se ejecuta sobre un contenedor Docker independiente.

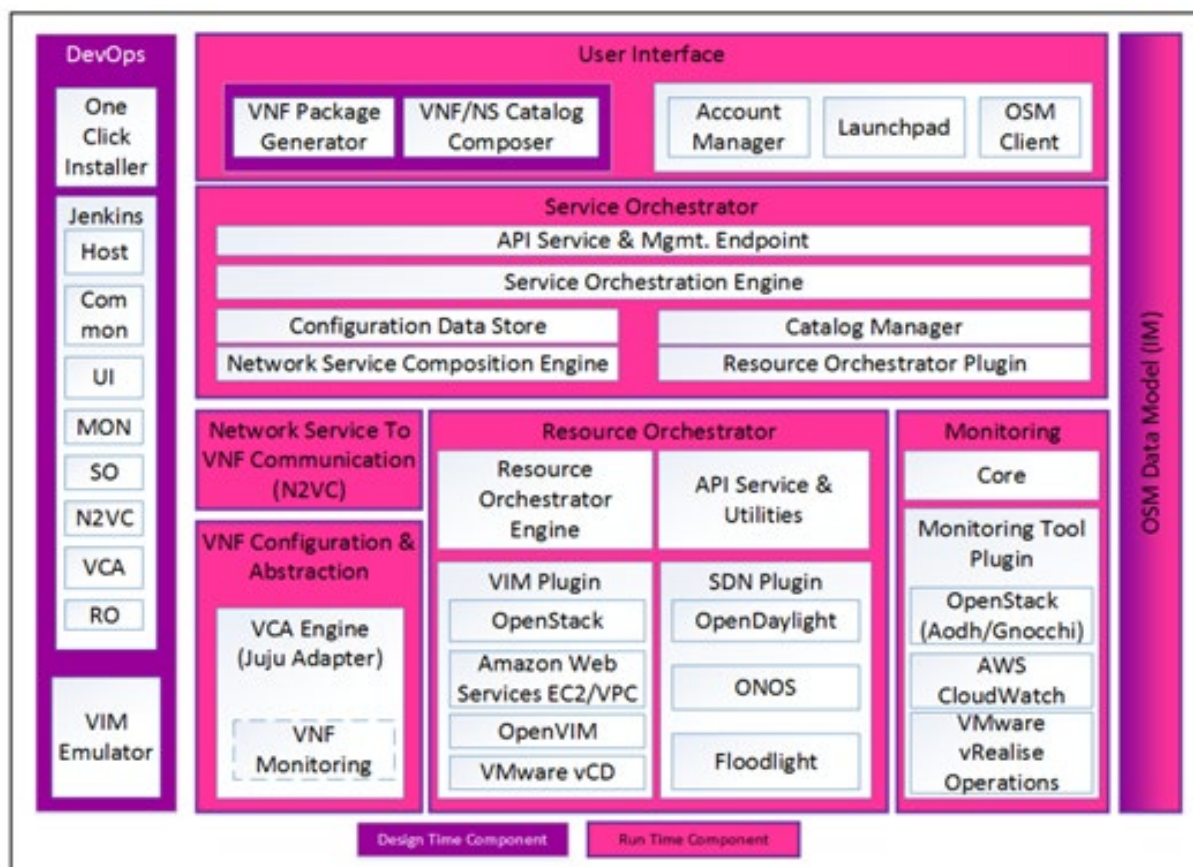


Figura 7. Arquitectura en detalle OSM

- **User Interface (UI):** por un lado, tiene el *VNF Package Generator* y el *VNF/NS Catalog Composer*, que permiten al usuario desarrollar descriptores de la entidad a implementar de manera rápida y sencilla y crear los correspondientes paquetes de forma correcta. Por otro lado, el *Account Manager* gestiona las credenciales de acceso al VIM; el *launchpad* es la interfaz gráfica interactiva del sistema que permite a los usuarios la gestión del ciclo de vida de los VNF/NS, proporciona estadísticas en tiempo real y muestra una vista detallada de la topología de red; el cliente CLI de OSM permite interactuar con su *Northbound API REST* de manera remota.
- **Service Orchestrator (SO):** su componente principal es el *Service Orchestration Engine*, responsable de todos los aspectos del servicio de orquestación, incluyendo gestión del ciclo de vida y ejecución de primitivas. Además, controla los usuarios y proyectos y el acceso basado en roles. Otros componentes incluyen el *Configuration Data Store* para almacenar el estado del SO, el *Network Service Composition Engine* para validar que los descriptores de VNF/NS se ajustan al

esquema YANG, el *Catalog Manager* para soporte de operaciones CRUD del ciclo de vida de los descriptores y paquetes, y el *Resource Orchestration Plugin* para proporcionar una interfaz que conecte con el orquestador de recursos (RO).

Actualmente, este componente se ha transformado en una versión más ligera, denominada el *LifeCycle Management* (LCM).

- **Network Service to VNF Communication (N2VC):** comunicación entre el SO y el VCA.
- **VNF Configuration and Abstraction (VCA):** con la ayuda de un controlador Juju y a través de “charms” crea VNFMs para gestionar el ciclo de vida y las configuraciones de Day-0, Day-1 y Day-2. Esta es una terminología usada por OSM que hace referencia al despliegue (Day-0), configuración inicial (Day-1) y aprovisionamiento (Day-2).
- **Resource Orchestrator (RO):** encargado de gestionar y coordinar la asignación de recursos en múltiples VIM y controladores SDN. Contiene los plugins necesarios para conectar el RO con las interfaces específicas de los mismos.
- **Monitoring (MON):** diseñado para integrarse con la herramienta de monitorización elegida: OpenStack Aodh, AWS CloudWatch, VMWare vRealize...

Esta arquitectura ha experimentado ciertas evoluciones a lo largo de los años entre las que destacan la introducción de un bus de comunicación común que proporciona un canal asíncrono entre componentes utilizando la tecnología Kafka. Esto permite la integración de nuevos módulos de manera sencilla y facilita la centralización de servicios (base de datos, almacenamiento, logs y autenticación). Y un nuevo módulo, el *Policy Module* (POL) para automatización de operaciones en bucle cerrado (*close loop automation*).

Modelo de datos: descriptores de NFV MANO

Los descriptores son las plantillas de configuración esenciales para que OSM pueda crear las instancias de funciones y servicios de red y gestionar su ciclo de vida, por ello deben incluir todos los campos necesarios para llevar a cabo esta tarea y describir correctamente sus características.

Desde la Release 9 el modelo de datos de OSM está alineado con ETSI NFV-SOL006 para garantizar la estandarización. No obstante, también amplía este modelo con campos que

todavía no se han abordado en los estándares de NFV y proporciona la correspondiente realimentación. Se trata de un modelo de datos basado en la especificación YANG [97] para estandarizar el formato de datos utilizado en los descriptores NFV.

Se pueden distinguir dos tipos de descriptores que se corresponden con las dos entidades fundamentales de NFV: el VNFD y el NSD [100].

Virtual Network Function Descriptor (VNFD)

Un descriptor de VNF describe una función de red virtualizada compuesta por una o varias *Virtual Deployment Unit* (VDU), tal y como ilustra la **Figura 8**, cada una de las cuales representa una máquina virtual o contenedor, y los enlaces virtuales (VL) que las unen. Los puntos de conexión entre VDU y VL se denominan *Connection Points* (CP). En la figura se puede ver un ejemplo de una VNF compuesta por dos VDU unidas con un enlace interno.

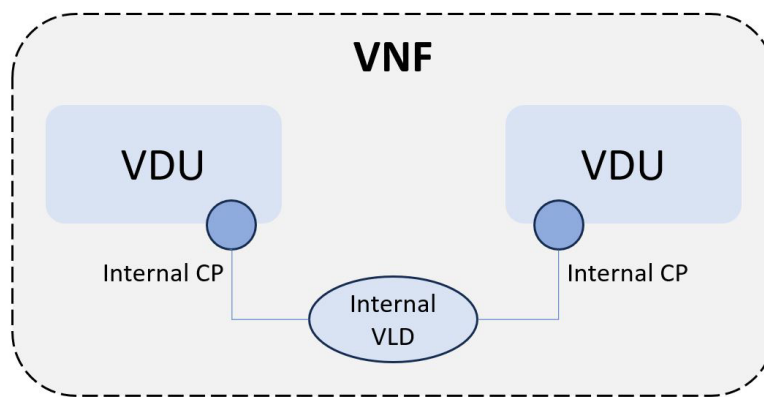


Figura 8. Diagrama VNF

Los elementos principales de un VNFD son:

- **VNF Connection Points:** puntos de conexión externos para conectar una VNF con otras o con una red externa.
- **Constituent VDUs:** lista de las máquinas/contenedores que componen una VNF. Para cada una se detallan las interfaces y recursos de computación requeridos.
- **VDU Dependencies:** dependencias entre VDU, usadas por el orquestador para determinar el orden de inicio.
- **Internal VLDs:** enlaces internos que conectan las VDU de una VNF.
- **VNF Configuration Primitives:** primitivas de configuración para realizar configuraciones remotas.

- **Monitoring Parameters:** lista de parámetros a monitorizar.

Network Service Descriptor (NSD)

De manera similar a los VNFD, los descriptores de NS describen un servicio de red como una agrupación de VNF, conectadas mediante VL, según muestra el diagrama de la **Figura 9**.

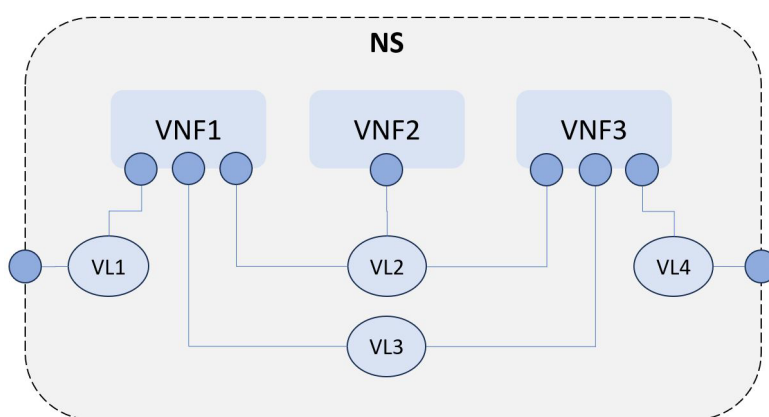


Figura 9. Diagrama NS

Los elementos principales de un NSD son:

- **NS Connection Points:** puntos de conexión externos para conectar un NS con otros o con una red externa.
- **Constituent VNFs:** lista de VNF que forman el servicio de red.
- **VNF Dependencies:** dependencias entre VNF, usadas por el orquestador para determinar el orden de inicio.
- **Virtual Links:** enlaces que conectan las VNF de un NS.
- **NS Configuration Primitives:** primitivas de configuración para realizar configuraciones remotas a nivel de servicio.
- **Input Parameters:** lista de parámetros dentro del NSD que pueden ser personalizados durante la instanciación.

VIM soportados

Se ha visto que OSM solo cubre dos de los bloques funcionales de la arquitectura NFV MANO y necesita de un tercero para gestionar los recursos de computación, almacenamiento y red, el VIM. Se van a analizar tres de las opciones disponibles: VIM de

tipo nube pública como Amazon Web Services (AWS) o Microsoft Azure, OpenStack y clúster de K8s.

- **Amazon Web Services (AWS).** Es la nube más adoptada y completa en el mundo, que ofrece más de 200 servicios integrales de centros de datos a nivel global. Se caracteriza por ofrecer gran variedad de servicios en diferentes categorías: computación, almacenamiento, bases de datos, redes y entrega de contenido, análisis, aprendizaje automático e inteligencia artificial, y seguridad, identidad y cumplimiento. Entre los servicios más destacados se encuentran *Amazon Elastic Compute Cloud* (Amazon EC2), un servicio web que proporciona capacidad de computación mediante servidores físicos en los centros de datos de Amazon. *Amazon Simple Storage Service* (Amazon S3), proporciona recursos de almacenamiento escalables, flexibles y rápidos. *Amazon Virtual Private Cloud* (Amazon VPC) permite crear una sección aislada de la *AWS Cloud* para gestionar una nube privada.
- **Microsoft Azure.** Al igual que AWS, es una nube pública de pago que ofrece Infraestructura y Plataforma como Servicio (IaaS y PaaS). Está gestionada por Microsoft y con ella se pueden crear, ejecutar y administrar aplicaciones en varias nubes, en el entorno local y en el perímetro, usando gran variedad de herramientas y marcos. Los servicios que ofrece son muy parecidos a los de AWS, destacando la creación de máquinas virtuales al contado, *Azure Kubernetes Service* (AKS) para crear y escalar aplicaciones en un clúster de Kubernetes o *Azure Stack* para soluciones híbridas.
- **OpenStack.** A diferencia de las anteriores, es un software de código abierto que, mediante su oferta IaaS, permite diseñar y gestionar nubes privadas sin el soporte de una nube pública. Se trata de un sistema que controla grandes grupos de recursos de computación, almacenamiento y red distribuidos en un centro de datos, todos ellos gestionados y aprovisionados a través de APIs con mecanismos de autenticación comunes. A parte de las funciones estándar de IaaS, ofrece componentes adicionales para orquestación, gestión de fallos y gestión de servicios, así como una interfaz web para facilitar su uso [101].

OpenStack busca proporcionar las siguientes capacidades [102]: compatibilidad con otras capas de abstracción (PaaS, motores de orquestación de contenedores), escalado continuo y eficiente, confiabilidad y durabilidad, integración personalizable y soporte

de operaciones especializadas. Estas características convierten a OpenStack en una de las plataformas de computación en la nube más importantes del mundo y hacen que se integre a la perfección como VIM en un entorno NFV.

Una de las maneras más sencillas de desplegar un entorno de OpenStack es mediante DevStack [103], un conjunto de scripts ejecutables que instalan los siete servicios básicos de OpenStack. En la **Tabla 5** se muestran los servicios más relevantes de la pila OpenStack.

Tabla 5. Servicios OpenStack

Servicio	Descripción
Nova	Gestiona la creación de instancias de máquinas virtuales y soporta distintos hipervisores
Placement	Proporciona una API HTTP para realizar un seguimiento de los inventarios y usos de los proveedores de recursos.
Cinder	Servicio de almacenamiento persistente en bloque para las instancias de computación. Es el responsable del ciclo de vida de los volúmenes, desde su creación y asociación a una instancia, hasta su liberación
Neutron	Incluye la gestión de direcciones IP, DNS, DHCP, balanceo de carga y gestión de los grupos de seguridad
Horizon	Interfaz gráfica de OpenStack. A través de ella se puede aprovisionar, gestionar y supervisar los recursos de la nube
Keystone	Servicio de identidad compartido que proporciona autenticación y autorización a toda la infraestructura. Admite varias formas de autenticación como un par usuario/contraseña o un token
Glance	Servicio de gestión de imágenes, entendiendo imágenes como copias virtuales de discos duros que se utilizan para la creación de instancias

- **Clúster de K8s**

Para dar soporte a la orquestación de contenedores y aplicaciones basadas en microservicios, OSM acepta como VIM un clúster de Kubernetes, conocido de forma abreviada como K8s. Kubernetes se define como una plataforma portable y extensible de código abierto para administrar cargas de trabajo y servicios [104]. A diferencia de otros orquestadores, K8s no opera a nivel de hardware, si no de contenedor. Los contenedores se basan en virtualización a nivel de sistema operativo independiente del sistema

anfitrión, lo que hace que sean pequeños y rápidos y más fáciles de administrar que una máquina virtual.

Un clúster de K8s está formado por una o varias máquinas (virtuales o físicas), llamadas nodos. Todo clúster debe tener al menos un nodo trabajador, donde se despliegan los Pods. Estos son conjuntos de contenedores con almacenamiento y red comunes que modelan los contenedores específicos de una aplicación. El otro elemento fundamental es el plano de control, que permite controlar los nodos trabajadores y gestionar la creación y ubicación de los Pods. Los componentes del plano de control pueden desplegarse en una cualquiera de las máquinas del clúster, sobre la que normalmente no se despliegan Pods y se conoce como nodo maestro.

Existen varias maneras de desplegar un clúster de K8s de diferente complejidad. La elección depende del uso que se le vaya a dar y las características que se necesiten. OSM propone dos:

- **Microk8s:** es la manera más rápida y sencilla de instalar Kubernetes. Se trata de una versión ligera de Kubernetes que permite desplegar un clúster mono o multi nodo, y que incluye todas las funcionalidades básicas sin tener que realizar configuraciones complejas. Se instala mediante un único paquete y es compatible con los principales sistemas operativos. Su creador es Canonical, compañía mantenedora de Ubuntu [105] .
- **Kubeadm:** es una herramienta que permite la creación manual de un clúster. Para ello se debe, en primer lugar, instalar los paquetes necesarios (Docker, kubelet, kubectl y kubeadm), y luego crear el clúster siguiendo la documentación oficial de Kubernetes [106].

Configuraciones Day-0, Day-1 y Day-2.

OSM utiliza la terminología de Day-0, Day-1, Day-2 para referirse a las distintas etapas de configuración de una VNF. De acuerdo con la documentación oficial [107] cada etapa hace referencia a las siguientes configuraciones:

- **Day-0:** la máquina se instancia y se configura el acceso para que pueda ser gestionada en etapas posteriores (importar claves ssh, crear usuario/contraseña, etc.). Para ello se utilizan ficheros *cloud-init*, un método de configuración de instancias en la nube que permite automatizar estas tareas.

- **Day-1:** la máquina se configura para empezar a proporcionar un servicio concreto. Incluye la instalación de paquetes, ejecución de comandos, modificación de ficheros de configuración, etc. Estos servicios se inician automáticamente después de la instanciación de la máquina.
- **Day2:** permite la posibilidad de reconfigurar la máquina una vez desplegada, para poder realizar operaciones sobre ella bajo demanda: hacer copias de seguridad o actualizar usuarios.

Mientras que las configuraciones de Day-0 son configuraciones muy básicas que se pueden implementar con cloud-init, las de Day-1 y Day-2 hacen uso de las tecnologías de Juju o Helm.

- **Juju Charms.** Es un software de código abierto desarrollado por Canonical que permite desplegar aplicaciones e infraestructura en la nube y gestionar sus operaciones de configuración en Kubernetes, contenedores, máquinas virtuales y nubes públicas y privadas [108]. Hace uso de “charms”, un conjunto de scripts para desplegar y operar software que se pueden adaptar al uso requerido y permiten gestionar todas las fases del ciclo de vida de una VNF.

En el contexto de OSM, el uso de charms es gestionado por el módulo VCA, que es el que contiene un controlador Juju. Los más utilizados son los “*Proxy charms*”: una vez se crea el “*charm*” y se incluye en el VNFD, el software que contiene se ejecuta en un contenedor LXD en la propia máquina de OSM y mediante ssh se conecta a la VNF necesaria.

- **Helm Charts.** Desde la Release 8 de OSM, se incluye también la posibilidad de ejecutar primitivas (acciones de configuración) haciendo uso de Helm Charts, como se muestra en la Figura 10. Helm es un proyecto mantenido por la Helm Community que facilita la gestión de aplicaciones en Kubernetes [109]. De forma parecida a los “juju charms”, un chart es una colección de ficheros que describen un conjunto de recursos de Kubernetes tales como pods o servicios. La comunidad proporciona plantillas genéricas que facilitan la creación y personalización de charts.

Cuando en OSM se quiere ejecutar una primitiva de Day-1 o de Day-2 utilizando un entorno de ejecución basado en Helm, se despliega de manera automática una pareja extra de pods en el clúster de K8s sobre el que está montado OSM. Estos pods ejecutan el

software contenido en el Helm chart y, al igual que los juju charms, siguen el ciclo de vida de la VNF, accediendo a ellas de manera remota.

En la siguiente imagen se muestra, a modo de resumen, el flujo que sigue la ejecución de primitivas de Day-1 y Day-2 usando ambas opciones. En este caso, la primitiva usando Helm corresponde a una VNF desplegada sobre un clúster de K8s, conocida como KNF, pero ambas tecnologías son válidas para los distintos tipos de funciones de red virtualizadas soportados por OSM.

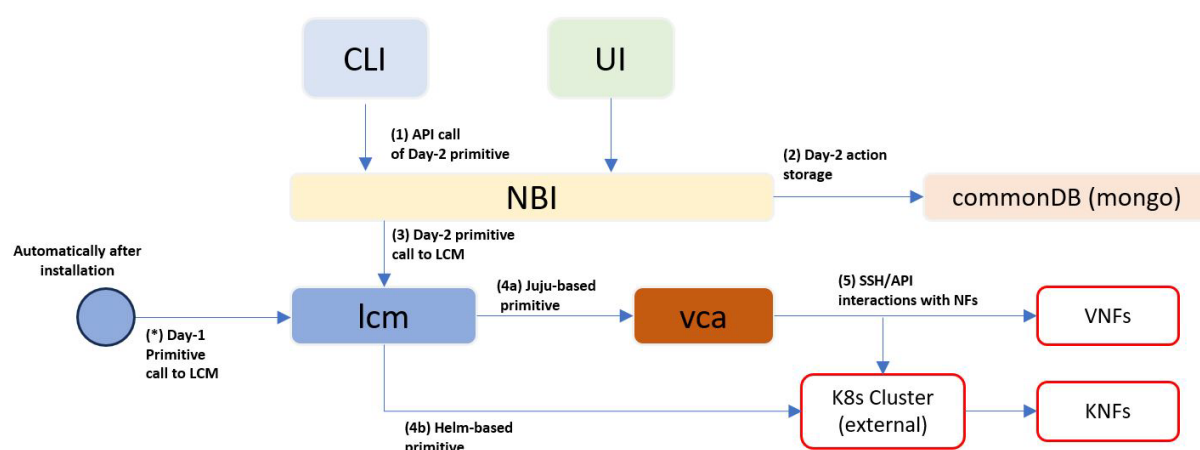


Figura 10. Acciones primitivas OSM

2.3.3. Tecnologías de comunicación

2.3.3.1. Message Queuing Telemetry Transport (MQTT)

MQTT (Message Queing Telemetry Transport) [110] es un protocolo Open Source M2M (Machine to Machine) de tipo colas de mensajes (message queue). Ha sido adoptado como estándar en la organización OASIS (Organization for the Advancement of Structured Information Standards), así como en ISO en 2016 (ISO/IEC PRF 20922) [111].

MQTT basa su funcionamiento en la pila TCP/IP, ofreciendo una comunicación asíncrona entre emisores y receptores de mensajes. Esta asincronía en la comunicación permite la separación, tanto en tiempo como en espacio, de los emisores y los receptores.

Para poder ofrecer un filtrado y una gestión de los mensajes generados por los clientes, dichos mensajes se encaminan a determinados topics organizados de forma jerárquica. El funcionamiento básico es:

- El cliente se conecta al servidor broker. Se permite la suscripción a cualquier topic disponible en el broker. Dicha conexión puede ser una conexión TCP/IP simple o una conexión TLS cifrada.
- El cliente publica los mensajes en el topic, enviando el mensaje y el topic al servidor broker.
- El broker se encarga de remitir el mensaje a todos los clientes que se encuentren suscritos a ese topic.

Los topics a su vez mantienen una estructura jerárquica, por lo que un cliente suscrito a un topic raíz, recibirá también todos los mensajes que cuelgan del topic al que se ha suscrito. La estructura basada en topics de MQTT se ve en la **Figura 11**.

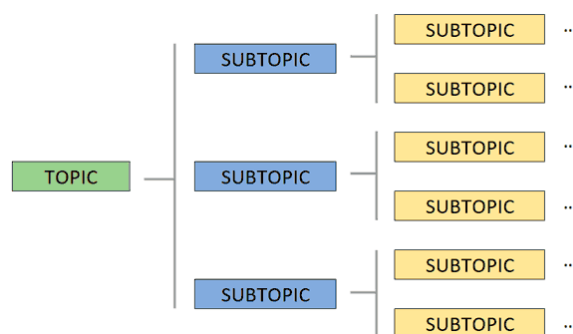


Figura 11. Jerarquía de topics MQTT

Los clientes son los encargados de iniciar una conexión TCP/IP con el broker, el cual mantiene un registro de los clientes conectados. Esta conexión se mantiene activa hasta que el cliente la finaliza.

Por defecto, MQTT emplea el puerto 1883 cuando se configura sin seguridad, y el puerto 8883 cuando se activa *Transport Layer Security* (TLS). Adicionalmente, la implementación de MQTT basada en C, permite el uso de la librería *liboqs* [112] la cual ofrece una colección de implementaciones, tanto para *Key Encapsulation Mechanism* (KEM) como esquemas de firma, para el uso de criptografía postcuántica en el marco de NIST [113], haciendo uso de TLS v1.3. Esta característica actualmente posiciona a MQTT como un protocolo de mensajería a tener en cuenta en el marco de las comunicaciones masivas seguras.

El broker MQTT es el elemento encargado de realizar la gestión de mensajes entre los clientes publicadores y suscriptores, garantizando la entrega de mensajes en función de

los topics definidos para la comunicación (**Figura 13** y ¡Error! No se encuentra el origen de la referencia.).

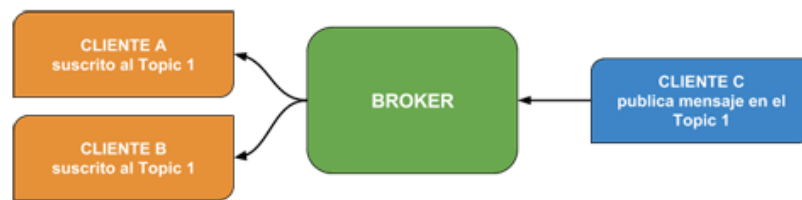


Figura 12. Funcionamiento protocolo MQTT a través de Mosquitto MQTT

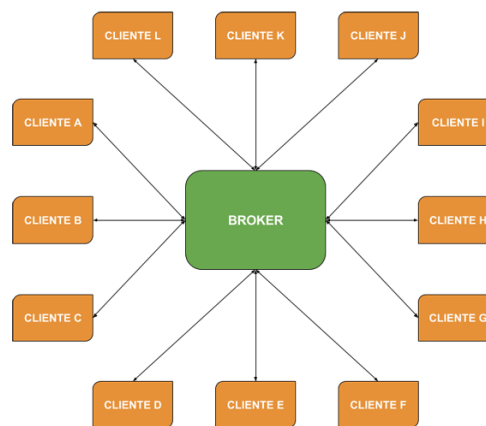


Figura 13. Arquitectura general Broker Mosquitto MQTT

Desde el sitio oficial del proyecto Mosquitto se proporcionan implementaciones de servidor y clientes MQTT que cumplen con el estándar del protocolo de mensajería MQTT utilizando el modelo de publicación – suscripción.

Mosquitto ofrece un software ligero, lo cual permite implementarse tanto en dispositivos de baja potencia, tales como microcontroladores o sensores, hasta servidores complejos.

El proyecto Mosquitto pertenece a la fundación Eclipse. Dicho proyecto se divide en tres partes:

- Servidor principal Mosquitto.
- Funciones para cliente (*mosquitto_pub* y *mosquitto_sub*) para permitir que los clientes se comuniquen con el servidor Mosquitto MQTT.
- Biblioteca Open Source de cliente MQTT en C.

Actualmente el proyecto Mosquitto ofrece tanto el código fuente a través de su repositorio oficial en GitHub como los paquetes binarios para Windows (en sus versiones de 64 y 32 bits), IOS, distribuciones Linux y S.O. específicos como Raspbian para Raspberry Pi.

2.3.3.2. Apache Kafka

Apache Kafka [114] es una plataforma de gestión de flujos de datos que ofrece las siguientes funcionalidades principales:

- Cola de mensajes con productores y consumidores para flujos de mensajes.
- Almacén tolerante a fallos de flujos de datos.
- Procesado de flujos de datos según se generan.

Los clústeres de Kafka permiten gestionar altos volúmenes de datos mediante el uso de una arquitectura distribuida de nodos llamados “*brokers*” apoyada en *ZooKeeper*. Este actúa como coordinador del clúster, así como orquestador de los clientes, almacenando entre otros los offset o último dato recibido para cada grupo de consumidores. La arquitectura general de esta tecnología se muestra en la **Figura 14**.

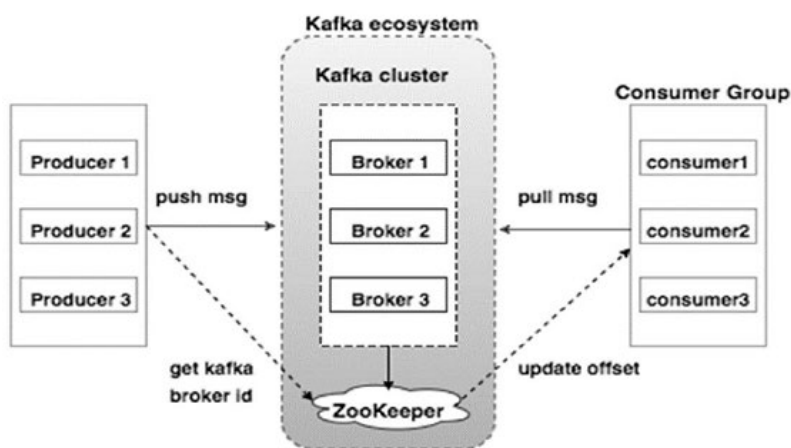


Figura 14. Arquitectura general Apache Kafka

El concepto principal de Kafka es el “*topic*”, los cuales permiten almacenar categorías de datos en diferentes estructuras de datos distribuidas, de manera muy similar a una cola individual en un servicio de colas o a una tabla de una base de datos. Internamente el flujo de datos de cada topic, conocido como “*log*”, se particiona en unidades de datos llamadas particiones, las cuales permiten distribuir la carga de trabajo entre distintos nodos y optimizar la paralelización de trabajos de los grupos de consumidores. No obstante, no

garantiza el orden de los datos dentro de distintas particiones de un mismo topic, aunque sí permite usar distintos criterios para la asignación de un dato a una partición en concreto dentro de un topic. La tolerancia a fallos queda condicionada al número de réplicas por topic, el cual es configurable.

Kafka se posiciona como cola de mensajes altamente resiliente o como buffer de recepción y procesamiento distribuido, siendo muy común su uso en arquitecturas de procesamiento de flujos de datos en tiempo real.

2.3.3.3. RabbitMQ

RabbitMQ [115] es un servidor de colas de mensaje de código abierto utilizado en el desarrollo de aplicaciones distribuidas. Permite a las aplicaciones intercambiar mensajes de manera asíncrona y desacoplada, lo que facilita la creación de sistemas altamente escalables y resilientes. A continuación, se puede observar el funcionamiento básico en la **Figura 15**.



Figura 15. Funcionamiento básico RabbitMQ

RabbitMQ se basa en el estándar *Advanced Message Queuing Protocol* (AMQP) [116] y ofrece una amplia gama de características para facilitar el intercambio de mensajes entre aplicaciones. Por ejemplo, permite a las aplicaciones publicar y suscribirse a mensajes de manera eficiente, gestiona la entrega de mensajes de manera confiable y soporta distintos tipos de colas, como colas persistentes y colas en memoria.

En resumen, RabbitMQ es un servidor de colas de mensajes de código abierto que facilita el intercambio de mensajes entre aplicaciones. Ofrece una amplia gama de características y es altamente escalable y resiliente.

2.3.4. Trabajos relacionados

Antes de abordar los diferentes trabajos relacionados a la gestión asociada al plano de datos y al plano de control en el contexto de SDN y NFV aplicados a redes 5G, es esencial

comprender el panorama general de estos enfoques. El uso de estas tecnologías ha revolucionado la forma en que se diseñan, implementan y gestionan las redes de telecomunicaciones modernas. En el caso de SDN, permite llevar a cabo una separación entre el plano de datos y el plano de control, permitiendo una gestión centralizada y dinámica de los recursos de red. Por otro lado, NFV permite la virtualización las funciones de red tradicionalmente implementadas en hardware dedicado, permitiendo una mayor flexibilidad y eficiencia en la prestación de servicios. Ambos paradigmas han sido ampliamente adoptados en el contexto de las redes 5G para mejorar la escalabilidad, la flexibilidad y la eficiencia operativa. En esta sección, exploraremos en detalle cómo se aplican estos principios en la gestión de los planos de datos y control en el contexto específico de las redes 5G.

- **Gestión asociada al plano de datos**

La tecnología Software Defined Network (SDN) se ha consolidado como un paradigma crucial en el contexto de las redes de comunicaciones móviles 5G, particularmente en lo referente al plano de datos. La aplicabilidad de SDN en este contexto se ha investigado en profundidad, destacando los beneficios que aporta a la próxima generación de redes inalámbricas.

Los autores en [117] destacan la capacidad de SDN para habilitar la gestión eficiente del tráfico en las redes 5G, permitiendo una asignación de recursos dinámica y personalizada, lo que conduce a una mejora significativa en la calidad de servicio. Adicionalmente esta tecnología favorece la orquestación de servicios en redes 5G, destacando la flexibilidad para implementar y adaptar servicios de manera rápida y eficaz.

Adicionalmente a la gestión de recursos y la calidad de servicio, existen numerosos trabajos en los cuales se pone de manifiesto la criticidad de las redes 5G desde una perspectiva de seguridad. Los autores en [118] exploran cómo la tecnología SDN puede fortalecer la seguridad de las redes 5G permitiendo la implementación de políticas de seguridad dinámicas y la detección de amenazas en tiempo real. Además, la investigación de [119] se centra en la aplicación de SDN en redes de acceso radio 5G y 6G para una gestión más eficiente y segura de recursos.

La automatización es otro aspecto relevante en las redes 5G. En el artículo [120] se analiza la automatización de políticas de calidad de servicio (*Quality of Service* - QoS) en redes 5G mediante SDN, destacando la simplificación de la administración y la mejora de la eficiencia operativa. Por último, el estudio [121] se centra en la virtualización de funciones de red en 5G a través de SDN, subrayando cómo esta combinación puede optimizar la infraestructura de red reduciendo los costos operativos asociados.

En resumen, la aplicabilidad de SDN en el plano de datos de redes de comunicaciones móviles 5G es una realidad que ha sido respaldada por la comunidad científica, estableciéndose como una herramienta capaz de abordar los desafíos y oportunidades que presenta la tecnología 5G, ofreciendo mejoras significativas en la gestión de tráfico, seguridad, automatización y eficiencia operativa.

- **Gestión asociada al plano de control**

La Virtualización de Funciones de Red (Network Function Virtualization - NFV) desempeña un papel fundamental en la evolución de las redes de comunicaciones móviles 5G, particularmente en el plano de control. A diferencia de la tecnología SDN que se enfoca en la capa de datos, NFV se centra en la capa de control. La aplicabilidad de NFV en el plano de control aborda desafíos específicos de las redes 5G.

Uno de los aspectos más destacados de NFV es la capacidad para facilitar la escalabilidad en las redes 5G. En el estudio [122], se pone de manifiesto cómo la tecnología NFV es capaz de ofrecer mecanismos para alcanzar una escalabilidad eficiente al posibilitar la asignación dinámica de recursos y la provisión de servicios personalizados a gran escala.

La eficiencia operativa es otro beneficio fundamental de NFV en el contexto de las redes 5G. La separación de funciones de red de la infraestructura física ofrece a los operadores la posibilidad de administrar recursos de manera más efectiva, lo que a su vez conduce a una reducción de los costos operativos. El trabajo [94] examina cómo NFV puede reducir los gastos operativos asociados a la gestión de redes de comunicaciones, contribuyendo al aumento de rentabilidad en las arquitecturas de operador.

La orquestación de servicios y la implementación de políticas de red también se benefician enormemente de esta tecnología aplicada a las redes de comunicación. En el estudio [122], se plantea cómo la tecnología NFV facilita la rápida implementación de servicios y políticas de red personalizados, mejorando la agilidad de la red y la capacidad de respuesta a las demandas de los usuarios. Por otro lado, el artículo de [123] se centra en la automatización de políticas de seguridad en redes virtualizadas mediante NFV, destacando cómo esta tecnología refuerza la seguridad de las redes 5G.

De manera complementaria a estos trabajos, distintos grupos de trabajo, como la ETSI [3] o 5GPPP [124] han publicado trabajos en los que se proporciona una visión general de los estándares y especificaciones relacionados con NFV en la industria de las telecomunicaciones alabando las ventajas del uso de esta tecnología en combinación con las redes 5G.

En resumen, NFV en el plano de control de redes de comunicaciones móviles 5G representa un avance esencial respaldado por investigaciones reales que abordan desafíos críticos y ofrecen beneficios en términos de escalabilidad, eficiencia operativa, orquestación de servicios y seguridad de la red.

2.4. Tecnologías de Redes Móviles Avanzadas

En la última década, las tecnologías de comunicaciones móviles han experimentado un rápido avance con la llegada de las redes 5G a través del estándar internacional para las telecomunicaciones móviles, IMT-2020 [125] definido por la Unión Internacional de Telecomunicaciones (UIT). Concretamente el IMT-2020 se apoya en las normas UIT-R M.2410 [126], a través de la cual se describen las especificaciones técnicas y funcionales de 5G, así como en la Release 15 y posteriores lideradas por la 3GPP.

Estas nuevas redes prometen revolucionar la forma en que nos comunicamos, interactuamos y consumimos datos, ofreciendo velocidades de transmisión ultrarrápidas, menor latencia y capacidad para conectar numerosos dispositivos en tiempo real. Se espera que esta tecnología desempeñe un papel fundamental en múltiples sectores, como el Internet de las cosas, la realidad virtual y aumentada, la conducción autónoma, la telemedicina y muchos otros.

Entre las ventajas que aportan las redes 5G respecto de otras generaciones de redes de comunicaciones móviles anteriores se encuentran una velocidad de transferencia de datos mucho mayor, una menor latencia, un ancho de banda más amplio, la posibilidad de interconectar muchos más dispositivos y un ahorro energético considerable. Estas ventajas vienen definidas por los estándares y los *Key Performance Indicators* (KPIs) detallados para las redes de quinta generación. El trabajo realizado en la presente tesis doctoral se centra en el soporte del paradigma de gemelos digitales en escenarios de red 5G.

2.4.1. Conceptos

Las redes 5G tienen definidos diferentes requerimientos basados en conseguir mejoras en los aspectos ya mencionados con anterioridad, pero, además, podemos observar que estos dependen del tipo de servicio que vaya a prestar la red, tal y como se muestra en la **Figura 16**.

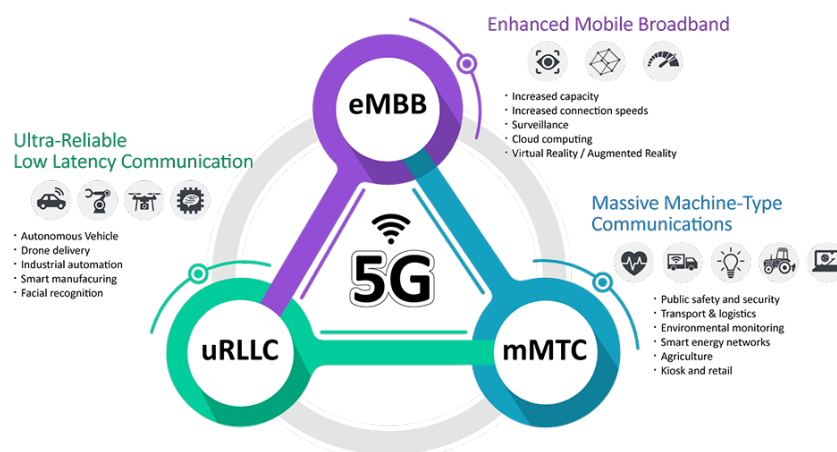


Figura 16. Escenarios de uso en 5G

- **Enhanced Mobile Broadband (eMBB):** se refiere a servicios y aplicaciones que requieren altas velocidades de transmisión, los usuarios pueden disfrutar de descargas y transmisiones de datos de alta calidad, como *streaming* de video en ultra alta definición (UHD), realidad virtual (VR) o juegos en línea. Su principal característica es la capacidad de proporcionar una conexión de banda ancha móvil más rápida y eficiente que las generaciones anteriores de redes móviles.
- **Massive Machine-Type Communications (mMTC):** se centra en el soporte de un gran número de dispositivos conectados que generan pequeñas cantidades de datos

en intervalos irregulares. Este tipo de comunicación es esencial para el Internet de las cosas (IoT), donde se espera tener un gran número de dispositivos interconectados. Se caracteriza por su capacidad para manejar eficientemente el tráfico de datos de baja potencia y tasa de bits de un gran número de dispositivos, optimizando así la vida útil de la batería y los recursos de la red.

- **Ultra-Reliable and Low Latency Communications (uRLLC):** se enfoca en aplicaciones y servicios que requieren una latencia extremadamente baja y una confiabilidad muy alta. Estos servicios deben ser críticos en términos de seguridad y tiempo de respuesta, para aplicaciones como las comunicaciones vehiculares de baja latencia y las operaciones críticas en tiempo real.

La arquitectura de red 5G se puede desplegar en dos modos diferentes: *Stand Alone* (SA) y *Non-Stand Alone* (NSA). Estos modos se refieren a la forma en que se implementa la red y las opciones de despliegue que se presentan. A continuación, se explican detalladamente las diferencias entre ambas arquitecturas:

- **Stand Alone (SA).** La arquitectura *Stand Alone* (SA) de 5G es una implementación completamente independiente, en la que tanto el *Control Plane* (CP) como el *User Plane* (UP) se despliegan utilizando recursos y funciones de red dedicados exclusivamente a la red 5G. En este modo, se utiliza el *5G Core Network* (5GC) y las funciones de red 5G se despliegan nativamente. Esto permite una mayor flexibilidad y capacidad para ofrecer todos los servicios 5G avanzados y cumpliendo con los niveles de capacidad y latencia definidos [127]. No se incluye una explicación de las arquitecturas de las opciones 1, al tratarse de una arquitectura únicamente con componentes LTE y no 5G, y las opciones 5 y 6 por tratarse estas de opciones poco relevantes ya que solamente se daría su caso si toda la red de acceso fuese 4G, pero se conectase a un 5G Core o al revés, diferenciándose estas de las opciones NSA que se explicarán a continuación.
- **Non-Stand Alone (NSA).** La arquitectura *Non-Stand Alone* (NSA) de 5G es una implementación que aprovecha la infraestructura existente de las redes 4G LTE para facilitar, agilizar y reducir los costes para el despliegue de las nuevas redes 5G. En esta arquitectura, el 5G utiliza sus estaciones base llamadas gNB (gNodeB), que se conectan a la red de eNBs (*Evolved NodeB*) existente de la red LTE.

2.4.2. Arquitectura basada en servicios

El núcleo de la red 5G, también conocido como Core 5G, presenta una arquitectura basada en servicios (SBA). En esta arquitectura, una serie de funciones de red trabajan en conjunto para habilitar las capacidades avanzadas de las redes 5G. La estandarización de estas funciones de red (NFs) y sus interfaces es fundamental, ya que permitirá a las operadoras desplegar un núcleo de red *multi-vendor*. A continuación, se describen algunas de las principales funciones del núcleo 5G, reflejadas en la **Figura 17**.

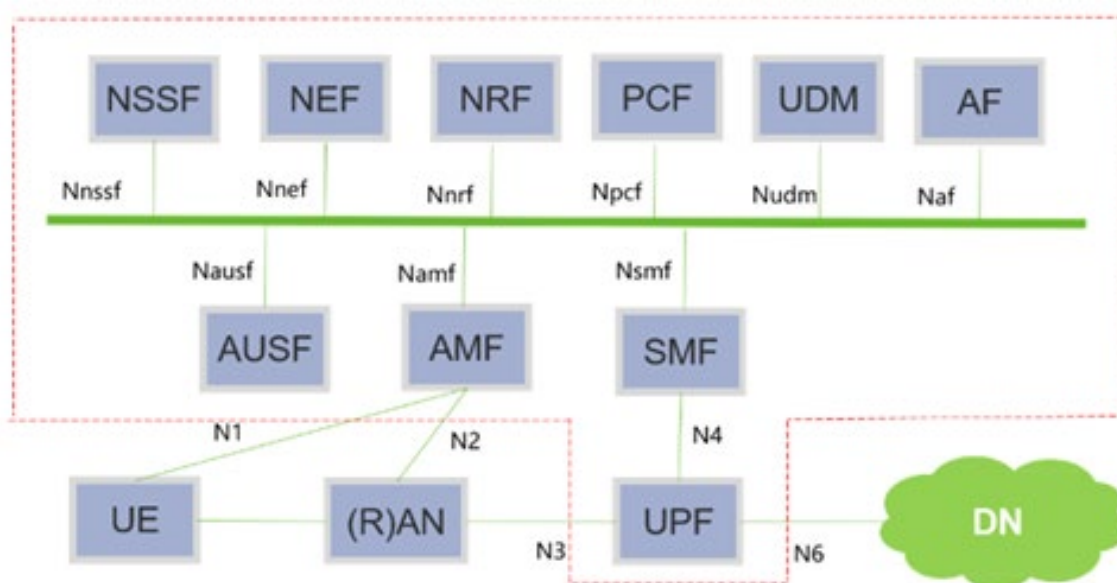


Figura 17. Arquitectura core 5G basada en funciones de red

- **Access and Mobility Management Function (AMF):** se encarga de gestionar el acceso y la movilidad de los dispositivos conectados a la red 5G. Proporciona funciones de autenticación, autorización y gestión de la movilidad de forma eficiente para un gran número de dispositivos, lo que permite una transición suave entre diferentes redes y asegura una conexión continua mientras el usuario se desplaza.
- **User Plane Function (UPF):** maneja el tráfico de datos de usuario y asegura una entrega eficiente de los mismos. Permite el enrutamiento óptimo de los paquetes de datos y se encarga de la gestión de la calidad de servicio, asegurando una experiencia de usuario óptima en términos de velocidad y latencia.
- **Session Management Function (SMF):** gestiona y controla las sesiones de comunicación en la red 5G. Proporciona funciones de establecimiento,

modificación y terminación de sesiones, lo que permite una comunicación continua y eficiente entre los dispositivos y las aplicaciones. Además, el SMF se encarga de aplicar políticas de red y de habilitar servicios específicos para las aplicaciones y los usuarios.

- **Unified Data Management (UDM):** se encarga de la gestión unificada de datos de usuario y de la autenticación centralizada. Almacena y administra la información de los usuarios, lo que facilita una autenticación y autorización rápidas y seguras en diferentes servicios y aplicaciones.
- **Authentication Server Function (AUSF):** es responsable de la autenticación y la gestión de las claves de seguridad de los usuarios. Proporciona un nivel de seguridad robusto al autenticar a los usuarios y garantizar la integridad y confidencialidad de las comunicaciones.
- **Policy Control Function (PCF):** se encarga de la gestión de políticas de red y de la asignación de recursos en función de los requisitos de las aplicaciones y los usuarios. Permite la implementación de políticas de control de tráfico, garantizando una distribución equitativa y eficiente de los recursos de red, cumpliendo con los acuerdos de nivel de servicio (SLA).
- **Application Function (AF):** se ocupa de la interacción entre las aplicaciones y la red 5G. Proporciona servicios específicos para las aplicaciones, permitiendo una integración fluida y optimizando el rendimiento de estas. El AF permite la implementación de servicios personalizados y adaptados a las necesidades de las aplicaciones y los usuarios.
- **Network Slice Selection Function (NSSF):** es responsable de seleccionar las instancias de segmento de red (*network slice*) adecuada para un servicio en particular, garantizando que los recursos de red se utilicen de manera eficiente.
- **Network Exposure Function (NEF):** proporciona una interfaz de programación (API) para que las aplicaciones externas puedan acceder a información y servicios de la red 5G. Permite la exposición y el acceso de forma segura a datos y capacidades de la red, lo que facilita el desarrollo de aplicaciones y servicios innovadores que se integren con la red 5G.
- **NF Repository Function (NRF):** es responsable de mantener un registro de las funciones de red disponibles en la red 5G. Proporciona información sobre las

capacidades y características de las funciones de red registradas, lo que permite a otras funciones de red descubrir y seleccionar las funciones adecuadas para satisfacer sus requisitos.

2.4.3. Implementaciones 5G

En el ámbito de la tecnología 5G, las implementaciones software desempeñan un papel crucial en el despliegue y funcionamiento de los diferentes componentes que conforman esta nueva generación de redes móviles. A continuación, se presenta la **Tabla 6** que recopila las implementaciones software más relevantes asociadas a los componentes fundamentales de 5G, como son el core de red, el *Radio Access Network* (RAN), el *User Equipment* (UE) o el *gNodeB*.

Tabla 6. Implementaciones de componentes 5G

Proyecto	Descripción
5GCore [128]	Implementación de sistema 5G en python
free5GC [129]	Implementación OpenSource del core 5G basado en 3GPP R15
Open5gs [130]	Implementación OpenSource del core 5G basado en 3GPP escrito en C
OAI-CN [131]	Implementación del core 5G
Internship-5GCN [132]	Implementación de API REST ful para comunicación del plano de control
Towards5gs [133]	Implementación del core 5G basada en Kubernetes
Free5GRAN [134]	Implementación del Radio Access Network 5G
gnbsim [135]	Simulador de los elementos gNB/UE
OAI-RAN [136]	Implementación del RAN incluyendo NodeB y UE
srsRAN [137]	Software Defined Radio aplicado a 4G/5G
UERANSIM [138]	Implementación del RAN incluyendo gNodeB y UE
STGUTG [139], [140]	Conector para la generación de tráfico de señalización y tráfico de usuario sobre core 5G

2.4.4. Especificación Gemelo Digital en 6G

Como se presentó en la sección anterior, la estandarización de la quinta generación de redes móviles, 5G, a través de la IMT-2020 representó un avance significativo en relación con la tecnología de comunicaciones móviles.

De igual forma, a finales de 2023, la ITU cerró el proceso de especificaciones para la IMT-2030 [141] a través de la recomendación ITU-R M.2160 [142]. Este estándar surge como un avance tecnológico, incluyendo en las comunicaciones móviles, áreas como la inteligencia artificial, el IoT a gran escala o la computación distribuida en la nube.

La IMT-2030 pretende establecer la transición, como muestra la **Figura 18** hacia la próxima generación de redes móviles, denominada 6G. Pese a que aun esté en sus primeras etapas de definición y desarrollo, la sexta generación se perfila como una evolución más avanzada, permitiendo el incremento en la velocidad, o la reducción de la latencia respecto al 5G.

Adicionalmente a estos avances, la IMT-2030, a través de las recomendaciones ITU-T Y.3090 [143] e ITU-R M.2160 introducen los conceptos de Digital Twin Network (DTN) y Quantum Key Distribution (QKD) como partes esenciales de la nueva arquitectura de red para el 6G.

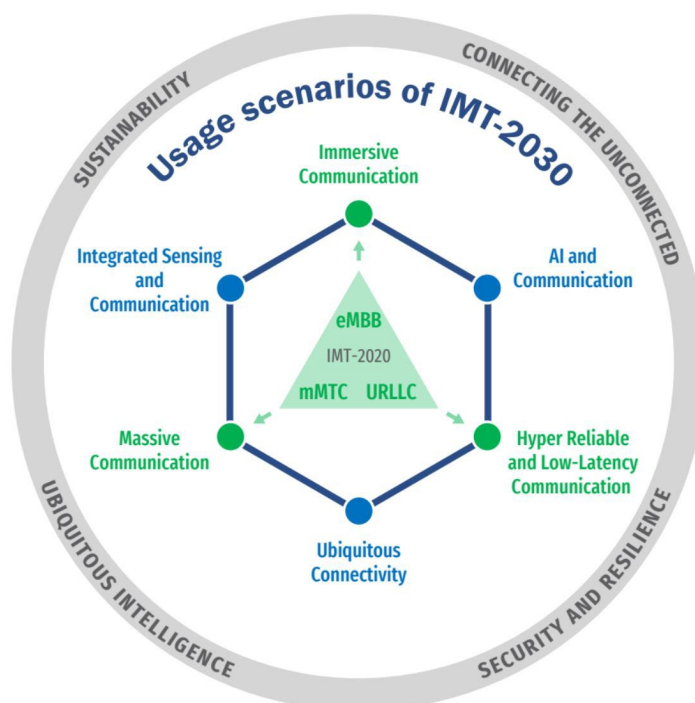


Figura 18. Escenarios de utilización y aspectos generales IMT-2030

En relación con la tecnología DTN, la recomendación ITU-T Y.3090 recoge una serie de requerimientos para estas redes, entre los cuales destacan:

- Requisitos funcionales:
 - Recolección eficiente de los datos, los cuales reflejen completamente el estado de la red física (topología, estados, enlaces, estadísticas de paquetes, etc).
 - Recolección basada en diferentes herramientas como SNMP, NetConf, in-band network Telemetry (INT), etc.
 - Necesidad de tener un repositorio eficiente para la unificación de los datos obtenidos del gemelo físico.
 - Gestión completa del ciclo de vida de los gemelos digitales de red, permitiendo la gestión de los datos, topología, o seguridad, así como las capacidades de automatización y registro de todas las transacciones realizadas en el contexto de la DTN.
- Requisitos de servicio:
 - Compatibilidad. Enfocada a permitir la compatibilidad de equipamiento de red de diferentes fabricantes, diferentes tipos de redes y diferentes aplicaciones de red.
 - Escalabilidad. Permitiendo el crecimiento o reducción de la escala de las redes virtuales desplegadas en la DTN bajo demanda ajustando automáticamente los recursos.
 - Fiabilidad. Necesidad de obtener una red altamente estable y confiable que permita la correcta interacción entre los entornos virtual y real.
 - Seguridad y privacidad. Debido a la sensibilidad de los datos en la DTN, esta debe garantizar la confidencialidad, integridad y autenticidad de los datos durante todo el ciclo de vida.

Pese a que la especificación 6G aún se encuentra en fases muy tempranas, existen trabajos incipientes como el presentado en [144], el cual pone de manifiesto el potencial de la tecnología basada en gemelos digitales de red, junto a los avances en el campo de IA y ML para poder llevar a cabo la optimización y control de las nuevas redes en tiempo real.

2.4.5. Trabajos relacionados

Actualmente, se pone de manifiesto el interés despertado por la tecnología basada en gemelos digitales aplicada a redes de comunicación 5G. En la **Tabla 7** se muestran una serie de trabajos relacionados en los últimos años, los cuales hacen uso del paradigma DT.

Tabla 7. Trabajos relacionados a NDT

Trabajo	Descripción
Network Traffic Prediction Model in a Data-Driven Digital Twin Network Architecture [145]	Diseño de un DTN basada en OMNeT++ con el objetivo de obtener datos para acelerar los procesos basados en ML
Digital Twin for 5G and Beyond [82]	Trabajo en el cual se describen los topics asociados a redes 5G y como pueden verse potenciados con el uso de DT
Deep Learning for Hybrid 5G Services in Mobile Edge Computing System: Learn From a Digital Twin [64]	Utilización de un Digital Twin ligado a escenario real con el objetivo de ser utilizado para el entrenamiento offline de modelos de aprendizaje profundo.
Digital Twin-Assisted Real-Time Traffic Data Prediction Method for 5G-Enabled Internet of Vehicles [146]	Aproximación para tratar el tráfico de datos generado por vehículos inteligentes como si fueran gemelos digitales y realizar su representación a través de comunicaciones 5G.
Secure and Latency-Aware Digital Twin Assisted Resource Scheduling for 5G Edge Computing-Empowered Distribution Grids [147]	Propuesta de un framework de aprendizaje federado basado en gemelos digitales capaz de facilitar la planificación de recursos para el Edge 5G.
A Digital Twin Network for Security Training in 5G Industrial Environments [148]	Propuesta de framework capaz de virtualizar redes de última generación en forma de gemelo digital para analizar y validar la seguridad de estos entornos
Digital Twin-enabled AI Enhancement in Smart Critical Infrastructures for 5G [149]	Propuesta de gemelo digital aplicado a escenarios energéticos con el objetivo de optimizar la precisión de tareas realizadas por algoritmos de IA
Placement of Social Digital Twins at the Edge for Beyond 5G IoT Networks [150]	Plantea el término de gemelos digitales con capacidades sociales desplegados en la red Edge, permitiendo optimizar la movilidad en dispositivos IoT
A Digital Twin for the 5G Era: the SPIDER Cyber Range [151]	Se presenta el Cyber-Range SPIDER, el cual habilita la creación dinámica de escenarios de red en forma de gemelos digitales con capacidad de recolección de datasets para ser utilizados por modelos basados en ML/IA

Capítulo 3

Metodología para el modelado y despliegue de gemelos digitales de red

Pese a que el paradigma de gemelos digitales está despertando un gran interés en la comunidad académica y la industria, tal y como se ha puesto de manifiesto en los capítulos anteriores, aún está lejos de ser una tecnología estandarizada. Si bien es cierto que los conceptos y bases generales están definidas a través de distintos organismos y consorcios como puede ser el IETF o el Digital Twin Consortium, la principal complejidad de este paradigma radica en su aplicabilidad final en distintos campos, los cuales trasladan sus características, retos y problemáticas al paradigma basado en gemelo digitales. A través del análisis del estado del arte realizado con relación a DT aplicados a redes de comunicaciones, se identifica como barrera principal la inexistencia de una arquitectura genérica, modular y escalable, la cual permita integrar distintas soluciones que permitan la creación de entornos de red avanzada basados en gemelos digitales. Pese a que existen trabajos relacionados, generalmente se centran en contribuciones específicas a las diferentes fases del ciclo de vida de los gemelos digitales, sin llegar a cubrir el ciclo completo de forma estándar.

Por ello, este capítulo describe en detalle la contribución principal que permite abordar el problema sobre la falta de mecanismos estandarizados para la planificación y despliegue de gemelos digitales de red, cubriendo desde los aspectos relacionados con la recolección de datos, el modelado, la adaptación de información, así como el despliegue e interconexión de ambos gemelos, físico y digital. La metodología parte de la arquitectura de referencia propuesta por el IETF, concretando las tecnologías habilitantes que

permitirían obtener un framework funcional capaz de cubrir todas las fases identificadas en el ciclo de vida del ecosistema basado en DT.

Esta contribución trata de dar respuesta a una serie de retos identificados en el análisis del estado del arte, entre los cuales destacan:

- La necesidad de desarrollar mecanismos de recolección de información con características muy concretas debido a la aplicabilidad del paradigma DT a redes de comunicaciones. Dentro de esta información, destaca la necesidad de recolección de datos que permitan obtener la topología de la red objetivo, completa o simplificada, las configuraciones mínimas necesarias presentes en la red para conseguir un NDT realista, telemetría asociada a la red, flujos de comunicación, estados, etc.
- La necesidad de preprocesamiento de datos recolectados, con el fin de realizar un modelado basado en datos capaz de reflejar tanto las características físicas de la red como el comportamiento de sus elementos en respuesta a flujos de datos específicos.
- La necesidad de un marco para la implementación y gestión del ciclo de vida de los gemelos de red basado en arquitecturas NFV y alineando los modelos que representan las entidades físicas con el modelo de datos específico respaldado por las arquitecturas MANO.
- El desarrollo e implementación de interfaces para la conexión segura entre elementos físicos y digitales.
- La necesidad de estandarización siguiendo el trabajo principal realizado por ETSI en el campo de gemelos digitales de red.

3.1. Diseño de la metodología

A través de los diferentes trabajos analizados en relación con el paradigma de Gemelos Digitales, se comienza a ahondar en la aplicabilidad de este paradigma al área de las redes de comunicación, poniendo especial interés en las redes de comunicación móvil de última generación. Tras la revisión de los esfuerzos realizados por la ETSI en diversos draft alineados con la temática NDT y trabajos como los realizados en, [152] [153] [154] [155] [156] [157] surge la motivación para el diseño de una metodología, la cual permita el desarrollo de Gemelos Digitales de red cubriendo el ciclo completo, comprendiendo desde la fase inicial de captación de datos, hasta la fase interconexión de ambos gemelos.

La **Figura 19** representa la metodología propuesta, conformada por siete fases diferenciadas, las cuales pueden tratarse como módulos independientes. Cada una de estas fases cubren tareas específicas, necesarias para llevar a cabo la contribución en el diseño, despliegue e interconexión de gemelos digitales de red aplicado a redes de comunicación.

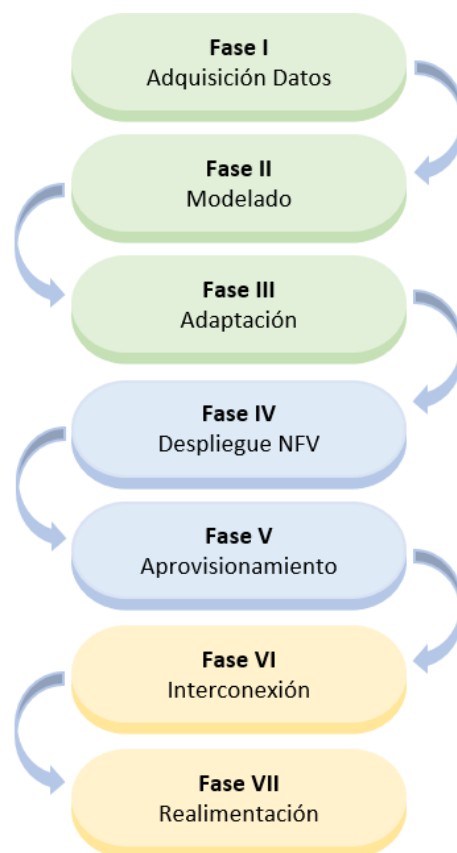


Figura 19. Metodología propuesta

- **Fase I. Adquisición de datos.** En esta fase inicial, se lleva a cabo una recolección exhaustiva de datos provenientes de la red objetivo, la cual quiere modelarse como NDT. En el caso específico de redes de comunicación, se abordan diferentes estrategias de adquisición de datos en función de la naturaleza de los equipos sondeados, diferenciando principalmente entre equipos de propósito específico, como routers o switches y equipos de propósito general, pudiendo ser estos últimos equipos virtuales, contenedores, etc. Dentro de la adquisición de datos, diferenciamos dos grandes grupos de datos. Los referentes a características físicas de la red, permitiendo la obtención de una representación semántica de la topología de red, y los referentes al comportamiento de los elementos de la red. Para llevar a cabo este trabajo, en el contexto de la fase I se diseñan, desarrollan e implementan una serie de agentes, los cuales, desplegados en la arquitectura física, serán capaces de orquestar correctamente la adquisición de datos.
- **Fase II. Modelado.** Una vez completada la fase de adquisición de datos, es necesario llevar a cabo el proceso de transformación y estructuración de los datos. Los agentes desarrollados en la fase I se encargan de preprocesar la información recibida a través de los diferentes sondeos a la red, generando un modelado de datos basado en JSON, el cual refleja los cuatro tipos de información que contempla la metodología, siendo datos topológicos y de comportamiento asociados a equipos de propósito general y específico. En esta fase no solo se organiza la información, sino que, tras el procesado de esta, se busca identificar patrones y relaciones que permitan enriquecer los datos para obtener una optimiza representación del sistema en forma de DT.
- **Fase III. Adaptación.** Una vez obtenido los ficheros JSON encargados de representar de forma semántica el comportamiento y la topología de la red física, es necesario realizar una adaptación a la especificación NFV-SOL006 [158] con el fin de desplegar el NDT en forma de funciones de red virtualizadas. Es importante remarcar la necesidad de crear un modelo de datos global, el cual permita recoger datos capaces de caracterizar o modelar el elemento identificado como gemelo físico, reflejando información que a priori no se contempla en la especificación NFV-SOL006 y que es necesaria para las fases posteriores, como el aprovisionamiento. Esta fase de adaptación pasa por implementar una serie de transformaciones capaces de alinear el formato de datos basado en JSON obtenido en la fase II con los descriptores de servicio

de red (NSD) y descriptores de función de red virtual (VNFD) necesarios para desplegar elementos de red en formato VNF. Esta adaptación toma principalmente los datos topológicos presentes en el formato de datos JSON.

- **Fase IV. Despliegue NFV.** Tras la adaptación de la información a la especificación NFV-SOL006, y la posterior generación de los ficheros NSD y VNFD asociados a la red objetivo, se agregan los ficheros al repositorio presente en la infraestructura MANO encargada de llevar a cabo la gestión del ciclo de vida del gemelo digital.
- **Fase V. Aprovisionamiento.** atrás la finalización de la fase de despliegue, se lleva a cabo la fase de aprovisionamiento sobre las instancias desplegadas, la cual en primer lugar se define a través de la terminología Day-0, Day-1 y Day-2 asociadas a las VNF. En este contexto Day-0 se corresponde con la instanciación y configuración básica de acceso del recurso virtualizado, haciendo uso de ficheros cloud-init, Day-1 lleva a cabo configuraciones específicas requeridas para el correcto funcionamiento de la VNF a desplegar, como puede ser la ejecución de comandos, modificación de configuraciones, etc. Finalmente, Day-2 permite la posibilidad de realizar tareas bajo demanda sobre la VNF, como puede ser la reconfiguración, la captura de tráfico, etc.
- **Fase VI. Interconexión.** Con el objetivo de dotar de un enlace de comunicación bidireccional, característica imprescindible y necesaria del paradigma DT. Se establece en la fase de interconexión el diseño de unas pasarelas de comunicación que permiten el establecimiento de distintos flujos de comunicación con el objetivo de intercambiar información entre la NDT y la red física, pasando por un elemento gestor encargado de orquestar la comunicación a alto nivel. Este proceso es esencial para garantizar la coordinación y sincronización entre las diferentes funciones virtuales y su contraparte física, permitiendo una gestión unificada, así como facilitando la monitorización y control del ecosistema completo.
- **Fase VII. Realimentación.** Esta fase se encarga de cerrar el proceso de desarrollo e implementación del gemelo digital de red, permitiendo la retroalimentación. Esta fase recae sobre el módulo gestor encargado de gestionar las comunicaciones a través de los gemelos, permitiendo la recolección y análisis de información, habilitando la opción de realizar ajustes o actualizaciones sobre la NDT. Esta realimentación continua asegura la adaptabilidad y la coevolución constante del ecosistema completo

garantizando su alineación hacia los cambios y requerimientos que puedan surgir en el entorno de red físico.

En relación con los agentes recolectores mencionados en la fase I, es importante remarcar que la metodología propuesta distingue dos estados independientes, asociados al funcionamiento de los agentes.

Ambos estados conllevan recolección de información, únicamente se tiene en cuenta si la NDT está creada o no. En caso de no existir aún, los agentes inician la recolección en estado 0, centrándose en datos topológicos necesarios para la creación del NDT. Una vez desplegada la NDT los agentes pasan al estado 1, recolectando información de comportamiento.

Como se puede apreciar en la **Figura 20**, la diferenciación de estados habilita o deshabilita las fases IV y V, debido a que estas fases únicamente se encargan del despliegue y configuración del NDT.

Adicionalmente a esto, este funcionamiento permite al ecosistema reconfigurarse en caso de detectarse un cambio en la topología, ya que al pasar del estado 1 al estado 0, se habilitan nuevamente las fase IV y V, permitiendo así una reconfiguración del NDT para adaptarse a los cambios detectados en la red física.

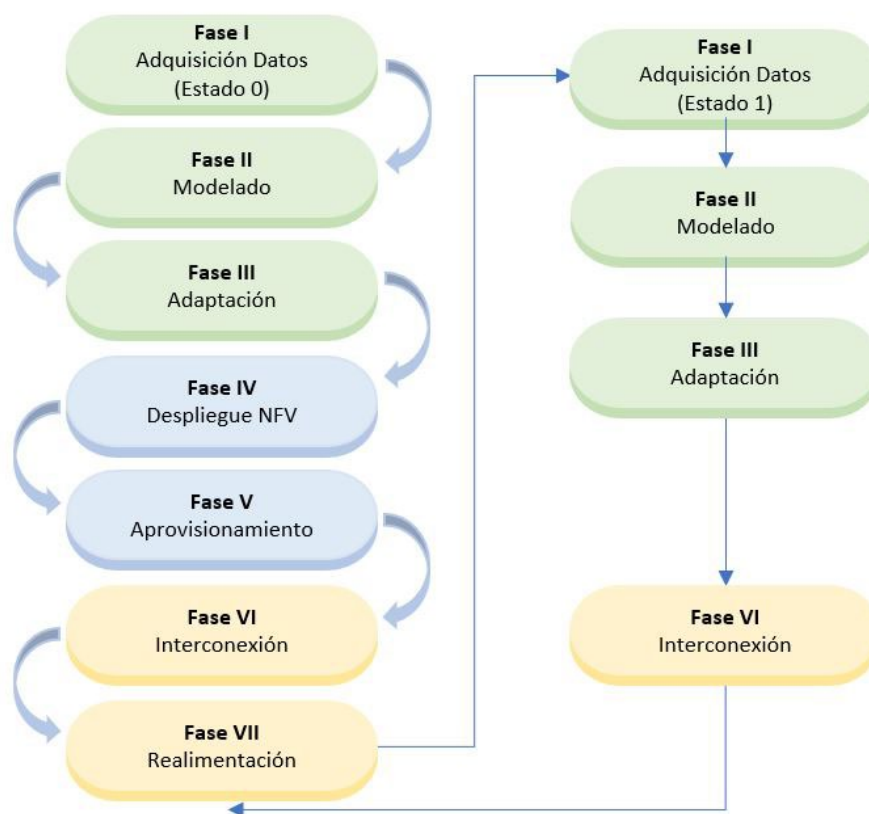


Figura 20. Relación entre estados y metodología

A continuación, se pasará a revisar en detalle el funcionamiento de cada una de las fases de la metodología, así como los elementos software diseñados para alcanzar los niveles de funcionalidad requeridos por cada una de ellas. Pese a que cada una de las fases pueden funcionar de manera más o menos independiente, es necesario entenderlas como piezas dentro de un ecosistema completo que permite el correcto despliegue y gestión del NDT.

En el siguiente capítulo se presentará la arquitectura diseñada capaz de dar soporte a la metodología, así como la aplicabilidad de esta a casos de uso concretos.

3.1.1. Fase I. Adquisición de datos

Tal y como se ha ido comentando a lo largo del documento, el paradigma de gemelos digitales se basa en la recolección de datos en el mundo físico con el objetivo de poder modelar y caracterizar el gemelo digital en el mundo virtual. Esta adquisición de datos está fuertemente condicionada al caso de uso sobre el cual se pretende aplicar el paradigma de gemelos digitales, abriendo un gran abanico de opciones acerca de qué tipo de datos recolectar y su posterior modelado, tal y como indican los autores en [28], los

cuales hacen una diferenciación basada en comportamiento o estructura, en función de lo que se persiga representar con el gemelo digital.

En el caso concreto de esta tesis doctoral, se persigue la aplicabilidad del paradigma de gemelos digitales a redes de comunicación, las cuales pueden ser caracterizadas a través de modelos basados en datos, reflejando información referente a la topología física y lógica de la red, las características software y hardware de los elementos que la conforman, así como información referente a configuración, flujos de comunicación, tráfico, etc.

Para poder llevar a cabo la recolección de datos sobre una red real, la cual permita obtener la información mínima necesaria para el modelado y despliegue de su contraparte digital, se plantea el diseño del sistema DANA (Digital Agent for Network data Acquisition). DANA es un agente software capaz de que ser desplegado de manera centralizada o distribuida sobre la red objetivo, en función de la tipología de equipos que la conformen.

En la **Figura 21** se muestra el diagrama de bloques de este sistema. No obstante, para comprender correctamente su funcionamiento, es necesario exponer las características específicas propias de los elementos que conforman una red de comunicación actual, las cuales han motivado las decisiones de diseño de este software.

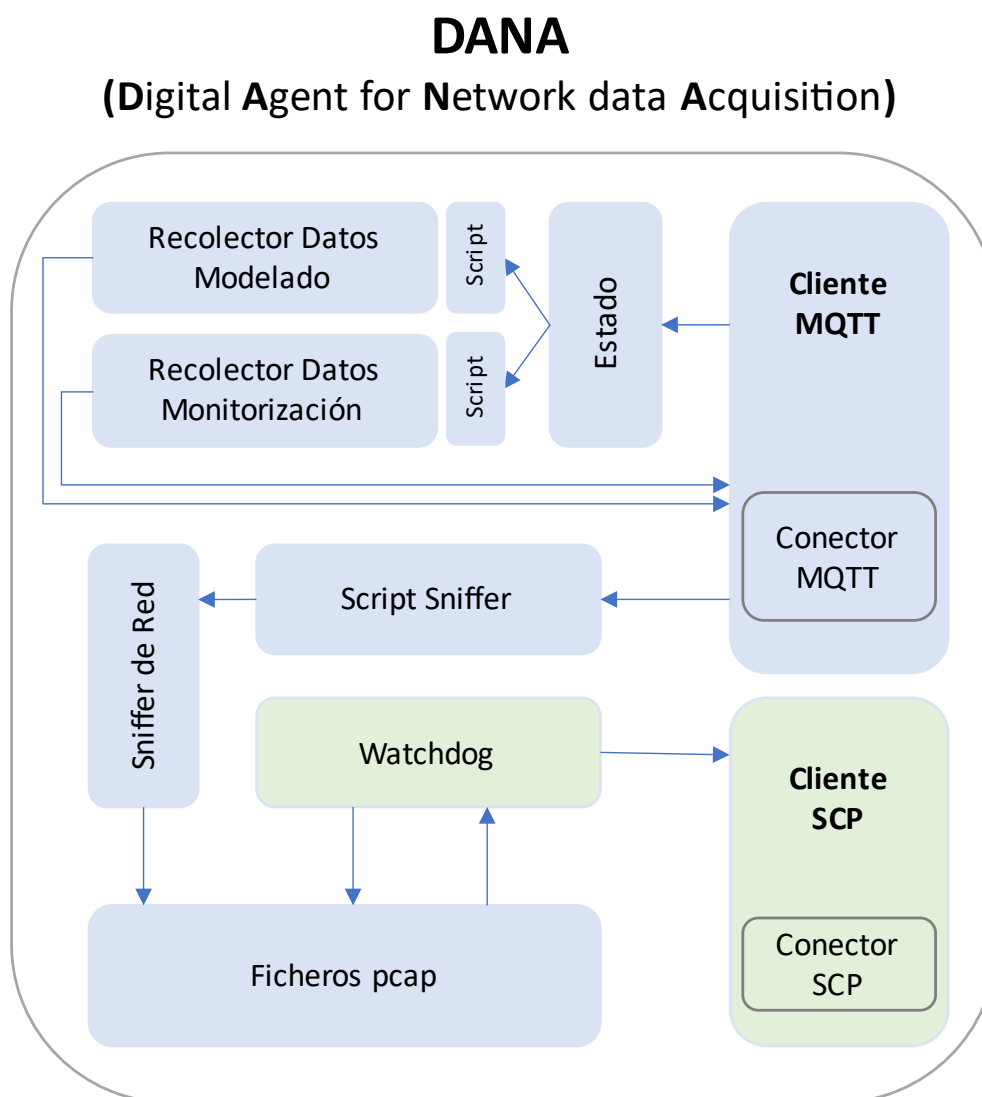


Figura 21. Diagrama de bloques funcionamiento del agente DANA

3.1.1.1. Estados

Según se puede ver en el diagrama de flujo de la **Figura 22**, DANA funciona en base a dos estados diferenciados controlados por un flag, el cual es enviado al agente vía protocolo MQTT desde el sistema de gestión del ecosistema NDT. Este elemento se verá en detalle en secciones posteriores. Estos estados controlan el funcionamiento específico del sistema:

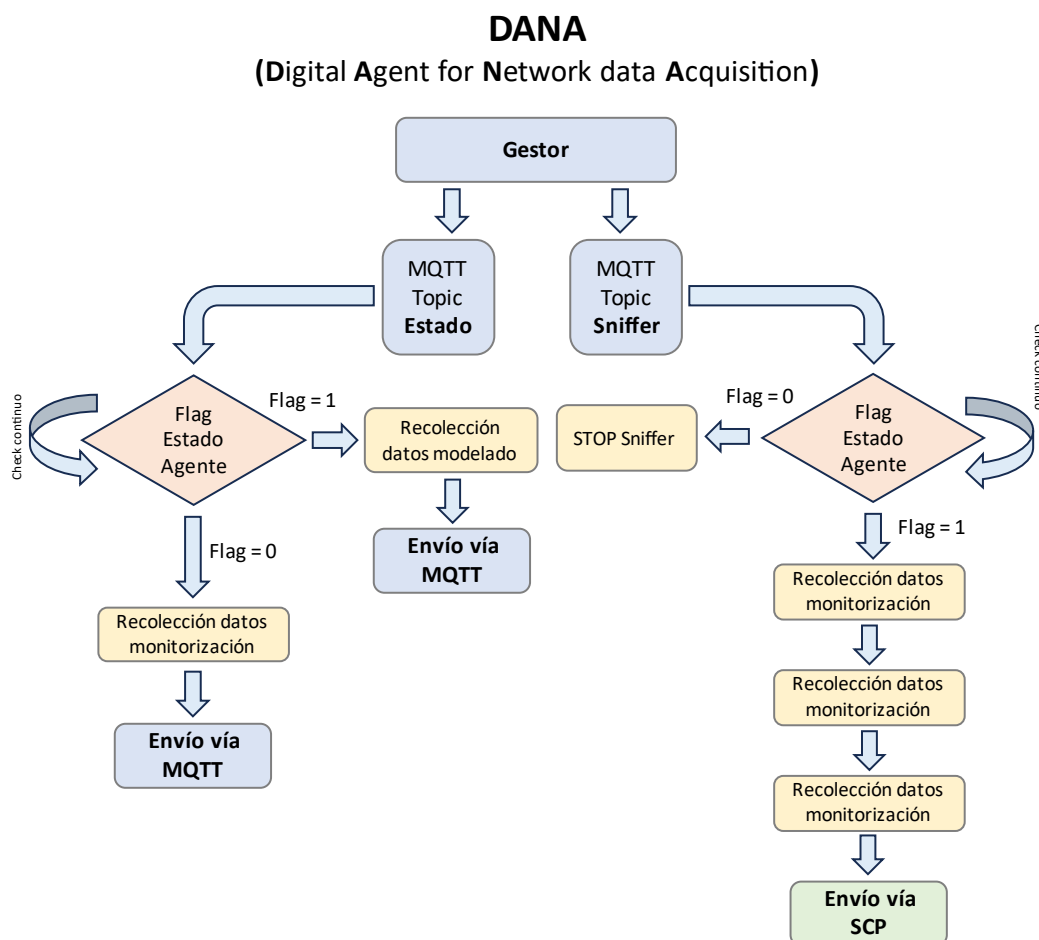


Figura 22. Diagrama de flujo funcionamiento del sistema DANA

- **Estado 0.** Se asume como estado inicial del sistema, en el cual aún no se ha generado ni desplegado el gemelo digital de la red. Este estado activa el submódulo encargado de la recolección de datos para el modelado de la red, centrándose en las características topológicas, hardware, software y de configuración.
- **Estado 1.** Una vez alcanzada la convergencia en la recolección de datos para el modelado, tras desplegar la NDT, el gestor notifica el cambio de estado al sistema DANA. En este estado se activa el submódulo encargado de recoger datos de tipo monitorización para reflejar el comportamiento de los elementos presentes en la red de comunicaciones.

Como se ha mencionado anteriormente, la diferenciación en estados dota al sistema DANA de la capacidad de reaccionar ante cambios sustanciales en la red física, permitiendo capturar dichos cambios y readaptar la NDT en consecuencia.

3.1.1.2. Modos de funcionamiento

Adicionalmente al funcionamiento principal basado en estados, es necesario realizar una diferenciación que afecta al modo de funcionamiento y por consecuencia a la recolección de datos siendo indiferente el estado en el que se encuentre el sistema DANA. Esta diferenciación está ligada a las características hardware y software de los equipos que conforman la red de comunicaciones objetivo, diferenciando entre dos tipos de equipos:

- **Equipos de propósito específico.** Tradicionalmente este tipo de equipos han sido y son los más habituales en cualquier red de comunicación. Podemos incluir en este grupo equipamiento de red como routers, switches, firewalls, gateways, etc.

Generalmente estos equipos carecen de un sistema operativo abierto, teniendo por lo general un sistema propietario, o restringido, el cual imposibilita la instalación de software adicional que permita, por ejemplo, la recolección de manera local información acerca del estado, funcionamiento, etc para su posterior envío a algún elemento encargado de centralizar esa información. No obstante, no quiere decir que sea imposible obtener información de estos dispositivos, ya que habitualmente implementan el protocolo *Simple network Management Protocol* (SNMP), el cual desde 1990, con su primera versión, ofrece un mecanismo estándar y ordenado para la recolección de información acerca del estado de equipos de red.

- **Equipos de propósito general.** Junto a los equipos de propósito específico, en los últimos años está cada vez más presente la coexistencia con equipos de hardware genérico el cual se encarga de alojar sistemas virtualizados, en contenedores, o arquitecturas lógicas basadas en tecnologías como SDN o NFV. A diferencia de los equipos de propósito específico, estos corren en hardware genérico con sistemas operativos los cuales ofrecen la posibilidad de instalar software adicional que permita recolectar información más detallada de la que se puede obtener a través de protocolos como SNMP o NETCONF, algo fundamental para tratar de obtener información de equipos que pueden presentar arquitecturas lógicas más complejas en

su interior al hacer uso de recursos virtualizados, ya sean máquinas virtuales, contenedores o funciones de red virtualizadas.

Como se puede observar en la **Figura 23**, dentro de los subsistemas recolectores de datos se diferencian dos estrategias de recolección: centralizada y descentralizada.

En la estrategia centralizada, el agente actúa como nodo central desplegado en la red, encargado de sondear a todos y cada uno de los elementos de red de propósito específico. Por el contrario, en la estrategia descentralizada, el sistema DANA pasa a jugar el papel de concentrador, similar a un *Network Management System* (NMS) encargado de recibir reportes periódicos de pequeños agentes satélite instalados en los equipos de propósito general. A continuación, pasaré a describir en detalle cada uno de estos funcionamientos.

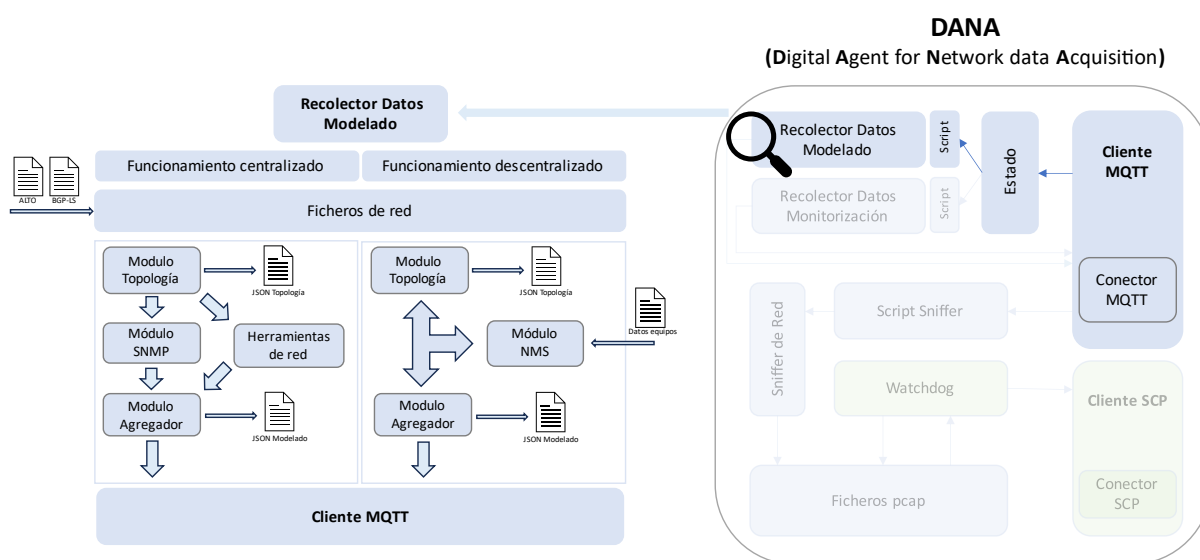


Figura 23. Diagrama de bloques funcionamiento agente (centralizado y descentralizado)

- Funcionamiento centralizado.** En este modo de funcionamiento, DANA se integra en la red física, actuando como nodo central encargado de realizar las peticiones SNMP hacia los diferentes elementos de propósito específico presentes en la red. En un primer momento se diseñó este submódulo de manera que realizara un autodescubrimiento de los elementos de red a través del primer equipo que al que estuviera conectado de manera directa. No obstante, esta aproximación planteaba una serie de problemas relacionados con bucles en caso de existir redes con redundancia en el conexionado de sus elementos. Por ello, se opta por tomar dos elementos de información generados por el protocolo *Application Layer Traffic Optimization* (ALTO) [159], el cual integra varios elementos encargados de analizar el tráfico asociado a

protocolos de enrutamiento e intercambiado entre los distintos elementos de red. Estos elementos se muestran en la **Figura 24** y **Figura 25**, siendo el mapa de red y mapa de costes respectivamente. Estos ficheros son utilizados por DANA como input del sistema, los cuales dan una primera visión de la topología de la red objetivo, permitiendo así realizar consultas SNMP directamente evitando tener que auto descubrir los elementos, y por consecuencia evitando inundar la red tráfico adicional asociado a los sondeos SNMP para el autodescubrimiento a través de las tablas de rutas de los equipos.

Uniendo el mapa de red proporcionado por ALTO a los sondeos SNMP específicos, es posible obtener una representación semántica de la red objetivo, obteniendo mayor detalle acerca de su topología, direccionamiento, configuración y funcionamiento, el cual se verá reflejado en la siguiente sección en el modelo de datos propuesto en la presente tesis doctoral.

```
GET /networkmap HTTP/1.1
Host: alto.example.com
Accept: application/alto-networkmap+json,application/alto-error+json

HTTP/1.1 200 OK
Content-Length: 449
Content-Type: application/alto-networkmap+json

{
  "meta" : {
    "vtag": {
      "resource-id": "my-default-network-map",
      "tag": "da65eca2eb7a10ce8b059740b0b2e3f8eb1d4785"
    }
  },
  "network-map" : {
    "PID1" : {
      "ipv4" : [
        "192.0.2.0/24",
        "198.51.100.0/25"
      ]
    },
    "PID2" : {
      "ipv4" : [
        "198.51.100.128/25"
      ]
    },
    "PID3" : {
      "ipv4" : [
        "0.0.0.0/0"
      ],
      "ipv6" : [
        "::/0"
      ]
    }
  }
}
```

Figura 24. Ejemplo del mapa generado por el protocolo ALTO

```
GET /costmap/num/routingcost HTTP/1.1
Host: alto.example.com
Accept: application/alto-costmap+json,application/alto-error+json

HTTP/1.1 200 OK
Content-Length: 435
Content-Type: application/alto-costmap+json

{
  "meta" : {
    "dependent-vtags" : [
      { "resource-id": "my-default-network-map",
        "tag": "3ee2cb7e8d63d9fab71b9b34cbf764436315542e"
      }
    ],
    "cost-type" : { "cost-mode" : "numerical",
                   "cost-metric": "routingcost"
                 }
  },
  "cost-map" : {
    "PID1": { "PID1": 1, "PID2": 5, "PID3": 10 },
    "PID2": { "PID1": 5, "PID2": 1, "PID3": 15 },
    "PID3": { "PID1": 20, "PID2": 15 }
  }
}
```

Figura 25. Ejemplo del mapa de coste generado por el protocolo ALTO

- **Funcionamiento descentralizado.** En este modo de funcionamiento, al igual que en el funcionamiento centralizado, DANA se integra en la red física, actuando esta vez como nodo central encargado de concentrar los reportes recibidos por agentes satélite instalados en los equipos de propósito general presentes en la red objetivo. Estos agentes permiten la obtención de información más específica que la ofrecida por el protocolo SNMP, a través de determinados módulos. Como se muestra en el ejemplo de la
- **Figura 26,** es posible obtener información del sistema e incluso la ejecución de scripts encargados de realizar tareas más complejas como por ejemplo el listado de imágenes de contenedores activas en un equipo. Estos agentes satélite se encargan de recuperar la información en local, generando un fichero XML, el cual es enviado de manera

periódica al agente inteligente que está actuando como concentrador en este caso.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<agent_data agent_name="20258da5d153c3ce1acef36" agent_alias="SRVPROYECTOAI" description=""
<module>
  <name><![CDATA[CPU Load]]></name>
  <type><![CDATA[generic_data]]></type>
  <description><![CDATA[User CPU Usage (%)]></description>
  <module_interval><![CDATA[1]]></module_interval>
  <min><![CDATA[0]]></min>
  <max><![CDATA[100]]></max>
  <min_critical><![CDATA[91]]></min_critical>
  <max_critical><![CDATA[100]]></max_critical>
  <min_warning><![CDATA[70]]></min_warning>
  <max_warning><![CDATA[90]]></max_warning>
  <unit><![CDATA[%]]></unit>
  <module_group>System</module_group>
  <datalist>
    <data>
      <value><![CDATA[1]]></value>
      <timestamp><![CDATA[2021-10-21 11:52:23]]></timestamp>
    </data>
    <data>
      <value><![CDATA[0]]></value>
      <timestamp><![CDATA[2021-10-21 11:52:23]]></timestamp>
    </data>
  </datalist>
</module>
```

Figura 26. Ejemplo de módulo de reporte agente satélite

De igual manera que en el funcionamiento centralizado, el mapa topológico proporcionado por ALTO es el punto de partida, permitiendo cubrir la recolección de información en escenarios mixtos formados tanto por equipamiento de red específico como por equipamiento genérico con capacidades de virtualización.

Estos agentes satélite son utilizados para la recolección de información, en el modo de funcionamiento distribuido, independientemente del estado (cero o uno) en el que se encuentre el sistema DANA, ya que es este último el encargado de tomar la información referente a topología o a comportamiento para completar el modelo de dato basado en JSON utilizado por las siguientes fases de la metodología propuesta.

3.1.1.3. Modelo de datos

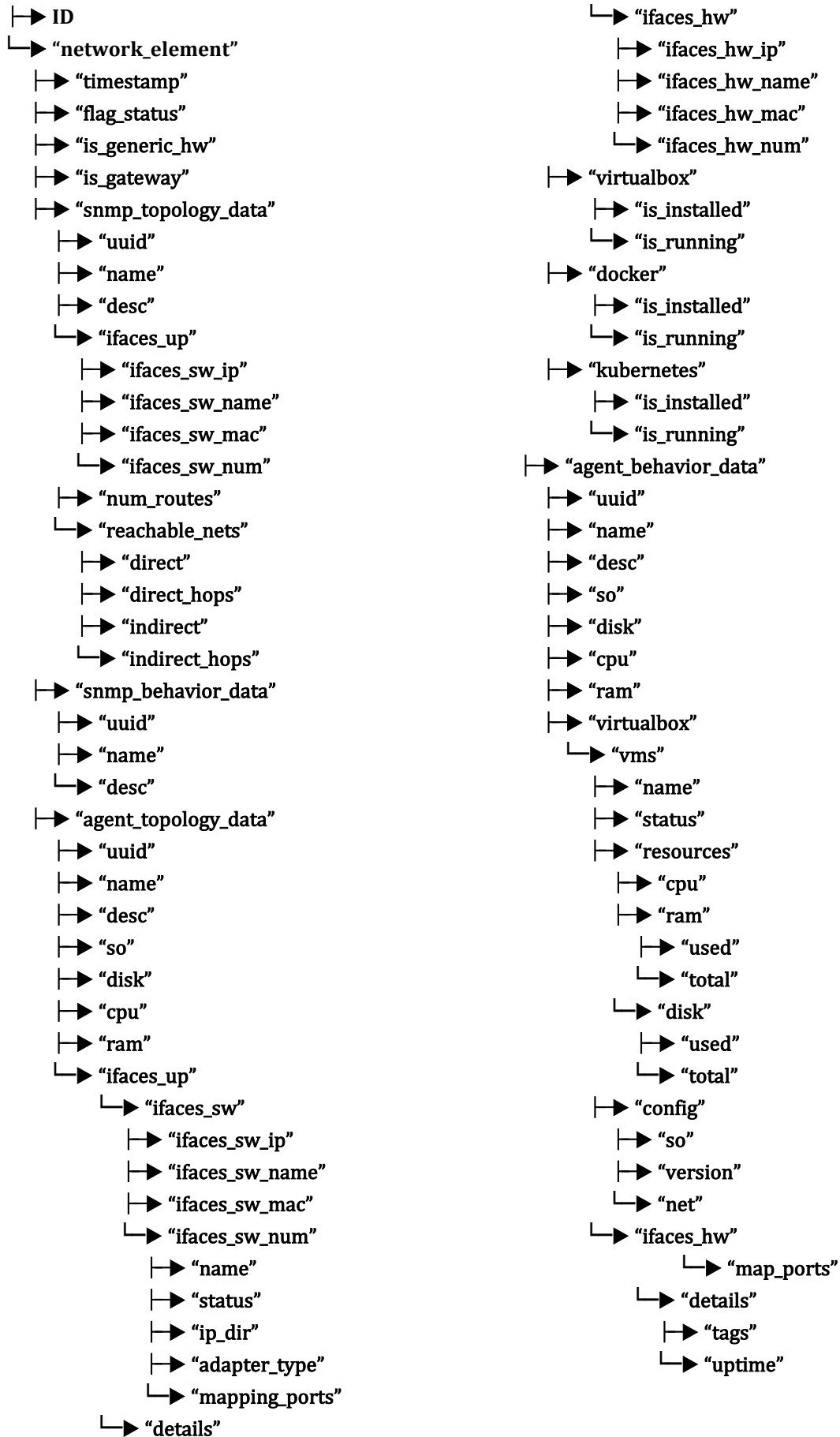
El resultado de los distintos estados y métodos de funcionamiento para la recolección de información tiene como objetivo final la creación de un modelo de datos basado en JSON, el cual refleje características específicas de cada uno de los elementos de la red, para ser utilizado posteriormente por las fases de modelado, adaptación y despliegue.

Este modelo de datos consta de dos bloques diferenciados. El primero de ellos formado por campos comunes para cada elemento de red, como la marca temporal, necesaria para poder ordenar los datos y mantener un histórico de la evolución de la red o el flag que identifica el estado en el que se encuentra el agente inteligente. El segundo bloque aborda campos específicos de cada elemento de red, destacando cuatro subgrupos en la **Tabla 8**.

Tabla 8. Subgrupos de datos dentro del modelo

Conjunto de datos	Funcionamiento del sistema DANA
snmp_topology_data	Asociado a estado 0 y modo centralizado
snmp_behavior_data	Asociado a estado 1 y modo centralizado
agent_topology_data	Asociado a estado 0 y modo descentralizado
agent_behavior_data	Asociado a estado 1 y modo descentralizado

Con el fin de manejar un único modelo de datos, se opta por unificar la información dentro de ese bloque, delegando en el agente la tarea de completar la información en función del estado en el que se encuentra y del tipo de elemento de red se trate. En la **Figura 27** se puede observar la arquitectura completa del modelo de datos, la cual permite recoger información topológica tanto de equipos sondeables únicamente por SNMP, equipos virtualizados, software basado en contenedores Docker y software basado en Kubernetes. Adicionalmente a la **Figura 27**, se incluye el detalle en la **Tabla 9** acerca del formato de dato y la descripción de cada campo presente en el modelo de datos.



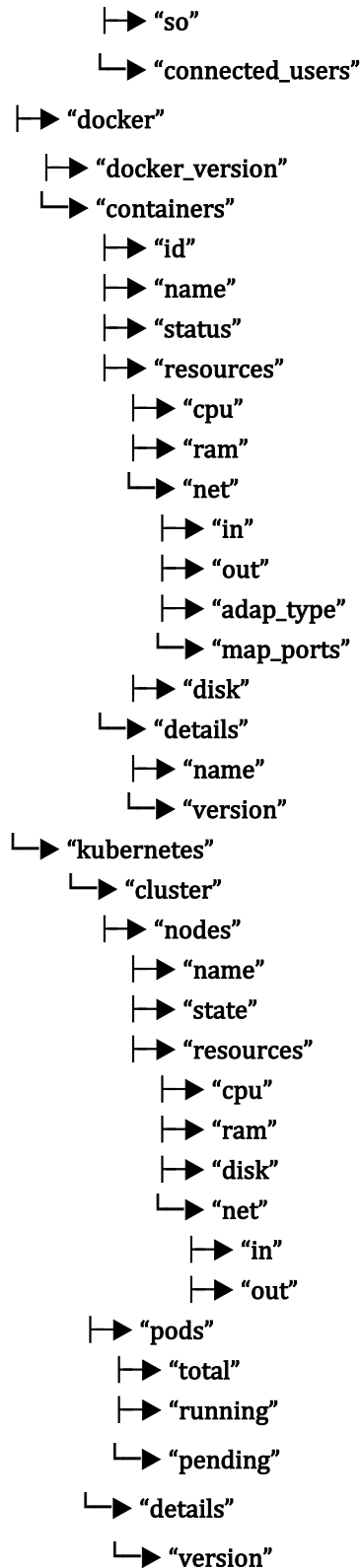


Figura 27. Arquitectura del modelo de datos

Tabla 9. Descripción de los campos del modelo de datos

Campo	Tipo de Dato	Descripción
id	JSON list	Identificador asociado a la red de comunicaciones objetivo de la recolección de datos.
network_element	JSON list	Listado de objetos JSON asociados a los diferentes elementos físicos o virtuales presentes en la red de comunicación.
timestamp	Epoch	Marca temporal asociada al instante de la captura de datos.
flag_status	Boolean	Flag referente al estado de DANA(true asociado a la obtención de datos para generar topología, false asociado a la obtención de datos para caracterizar el comportamiento.
is_generic_hw	Boolean	Flag referente al tipo de equipo que se está sondeando. En base al tipo de hardware (genérico o específico) se habilitará la recolección vía SNMP o mediante agentes.
is_gateway	Boolean	Flag referente al funcionamiento del equipo. Principalmente para la identificación de equipos router.
snmp_topology_data	JSON	
uuid	String	Identificador único asociado al equipo.
name	String	Nombre establecido en el equipo.
desc	String	Descripción establecida en el equipo.
ifaces_up	JSON	Objetos JSON referente a las interfaces activas en el equipo.
ifaces_ip	String list	Listado de direcciones IPs. establecidas en las interfaces del equipo.
ifaces_name	String list	Listado del tipo de interfaz presente en el equipo.
ifaces_mac	String list	Listado de direcciones MAC asociadas a cada interfaz.
ifaces_num	Integer list	Listado de identificadores asociados a cada interfaz.

num_routes	Integer	Número total de rutas presentes en la tabla de rutas del equipo.
reachable_nets	JSON	Listado de objetos JSON referentes a las redes alcanzables por el equipo.
direct	String list	Listado de segmentos de red conectadas directamente al equipo
direct_hops	Integer	Coste en formato saltos de alcanzar las redes direct
indirect	String list	Listado de segmentos de red alcanzables desde el equipo con número de saltos superior a 1
indirect_hops	Integer list	Listado de número de saltos asociado a cada segmento de red del campo anterior
snmp_behavior_data	JSON	
uuid	String	Identificador único asociado al equipo.
name	String	Nombre establecido en el equipo.
desc	String	Descripción establecida en el equipo.
agent_topology_data	JSON	
uuid	String	Identificador único asociado al equipo.
name	String	Nombre establecido en el equipo.
desc	String	Descripción establecida en el equipo.
disk	Integer	Tamaño de almacenamiento total (GB)
cpu	Integer	Número total de cores
ram	Integer	Tamaño total de memoria RAM (GB)
virtualbox	JSON	Objeto JSON asociado a VirtualBox
installed	Boolean	Flag asociado a la presencia en el equipo del sistema de virtualización VirtualBox
running	Boolean	Flag asociado al funcionamiento del sistema de virtualización VirtualBox
docker	JSON	Objeto JSON asociado a Docker
installed	Boolean	Flag asociado a la presencia en el equipo del sistema de virtualización Docker
running	Boolean	Flag asociado al funcionamiento del sistema de virtualización Docker

kubernetes	JSON	Objeto JSON asociado a Kubernetes
installed	Boolean	Flag asociado a la presencia en el equipo del sistema de virtualización Kubernetes
running	Boolean	Flag asociado al funcionamiento del sistema de virtualización Kubernetes
Agent_behavior_data	JSON list	
uuid	String	Identificador único asociado al equipo.
name	String	Nombre establecido en el equipo.
desc	String	Descripción establecida en el equipo.
so	String	Sistema Operativo
disk	Integer	Tamaño del almacenamiento utilizado (GB)
cpu	Integer	Porcentaje de uso de la CPU
ram	Integer	Tamaño de memoria RAM utilizada (GB)
virtualbox	JSON list	
vms	JSON list	Listado de objetos JSON en referencia a las máquinas virtuales presentes en el equipo
name	String	Nombre establecido para la máquina virtual
status	Boolean	Estado de la máquina virtual
resources	JSON list	
cpu	Integer	Número de procesadores asociados a la máquina virtual
ram	JSON list	
used	Integer	Tamaño de memoria RAM utilizada (GB)
total	Integer	Tamaño total de la memoria RAM establecida
disk	JSON list	
used	Integer	Tamaño de almacenamiento utilizado (GB)
total	Integer	Tamaño total de almacenamiento establecido (GB)
config	JSON list	
so	String	Sistema Operativo presente en la máquina virtual

version	String	Versión del S.O. instalado en la máquina virtual
net	JSON list	
name	String	Nombre de la red
status	Boolean	Estado del adaptador de red asociado
ip_dir	String	Dirección IP asociada al adaptador
adapter_type	String	Tipo de adaptador
mapping_ports	String	Listado de puertos mapeados
details	JSON list	
so	String	Sistema Operativo (redundante)
connected_users	Integer	Usuarios conectados a la máquina
docker	JSON list	
docker_version	String	Versión docker presente en la máquina
containers	JSON list	
id	String	Identificador asociado al contenedor docker
name	String	Nombre establecido al contendor
status	String	Estado del contendor
resources	JSON list	
cpu	String	Número de CPU asociados al contenedor
ram	String	Cantidad de memoria RAM asociada al contenedor (GB)
net	JSON list	
in	Integer	Número de bytes enviados durante el periodo de monitorización
out	Integer	Número de bytes recibidos durante el periodo de monitorización
adapter_type	String	Tipo de adaptador asociado al contenedor
mapping_ports	String List	Listado de puertos mapeados
disk	String	Tamaño del volumen docker utilizado
details	JSON list	
name	String	Nombre de la imagen Docker utilizada
version	String	Versión de la imagen Docker utilizada

kubernetes	JSON list	
cluster	JSON list	Lista de elementos presentes en el cluster de Kubernetes
nodes	JSON list	Listado de nodos presentes en el cluster
name	String	Nombre del nodo
state	String	Estado del nodo
resources	JSON list	
cpu	Integer	Cantidad de CPU asociada al nodo
ram	Integer	Cantidad de memoria RAM asociada al nodo (GB)
disk	Integer	Cantidad de disco asociado al nodo
net	JSON list	
in	Integer	Número de bytes enviados durante el periodo de monitorización
out	Integer	Número de bytes recibidos durante el periodo de monitorización
adapter_type	String	Tipo de adaptador
mapping_ports	String list	Listado de puertos mapeados
details	JSON list	
tags	String list	Etiquetas asignadas al nodo
uptime	String	Tiempo de vida del nodo
pods	JSON list	
total	Integer	Número total de pods presentes en el cluster
running	Integer	Número de pods activos en el cluster
pending	Integer	Número de pods pendientes de despliegue en el cluster
details	JSON list	
version	String	Versión de kubernetes

3.1.1.4. Comunicación

Una vez recolectada la información reflejada en el modelo de datos de la Tabla 9, y generado el fichero JSON, DANA hace uso del submódulo de comunicación basado en MQTT que habilita el envío de información hacia el sistema de gestión del ecosistema NDT, el cual se encargará, en función del estado del agente, de desplegar la NDT (estado 0) o de propagar información hacia el NDT (estado 1).

Como se puede apreciar en la **Figura 28**, el agente desplegado en la red real hace uso de dos elementos de comunicación, publicadores MQTT y SCP respectivamente. Los cuales son utilizados principalmente en función del tipo de dato a enviar, siendo el primero el utilizado únicamente para los datos asociados al modelo de dato presentado anteriormente, y el segundo utilizado para el envío de tráfico capturado en la red real.

Esta comunicación basada en flujos se explica con mayor detalle en la fase VI de la metodología presentada en la **sección 3.1.6**.

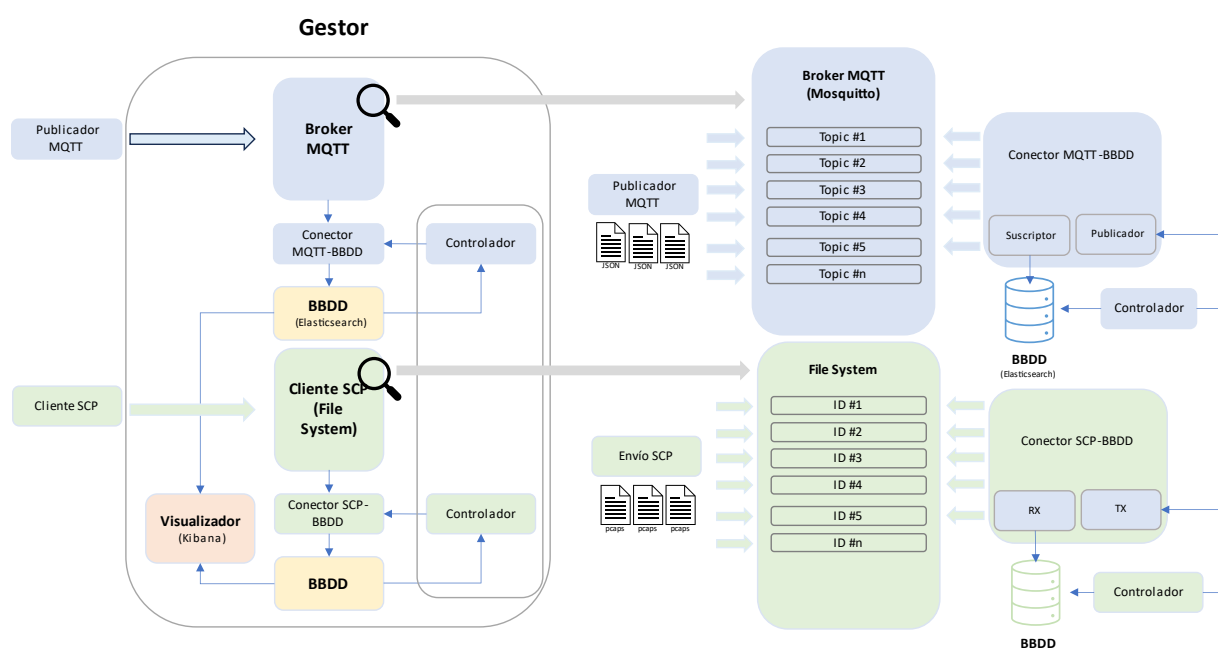


Figura 28. Diagrama de bloques flujos de comunicación hacia el gestor

3.1.2. Fase II. Modelado DTs

Una vez finalizada la fase I, la información recogida por el sistema DANA mediante diversas tecnologías y herramientas (ALTO, SNMP, ICMP, Iperf3, agentes, etc.) se almacena en el propio sistema. En la fase de modelado se realiza un análisis en profundidad de los datos obtenidos de la topología, como se muestra de manera

esquemática en la **Figura 29**. En este proceso se aplica correlación e inferencia de datos para obtener una representación semántica de la red en base a los datos recopilados en la fase I.

Es importante remarcar que, en esta fase, dado que las redes objetivo de esta metodología son redes que hacen uso de la tecnología 5G en modo Stand Alone, se incluye de manera *ad hoc* la descripción a alto nivel de los elementos virtualizados que conforman el núcleo de red 5G. Estos elementos son tratados como elementos estáticos en la fase de recolección de información, por lo que no son susceptibles de obtenerse mediante la fase de recolección de datos. El objetivo de esta fase es la obtención un modelo simplificado de la red física evitando una replicación completa de la red física en forma de NDT, lo que no sería efectivo desde el punto de vista de optimización de costes y recursos.

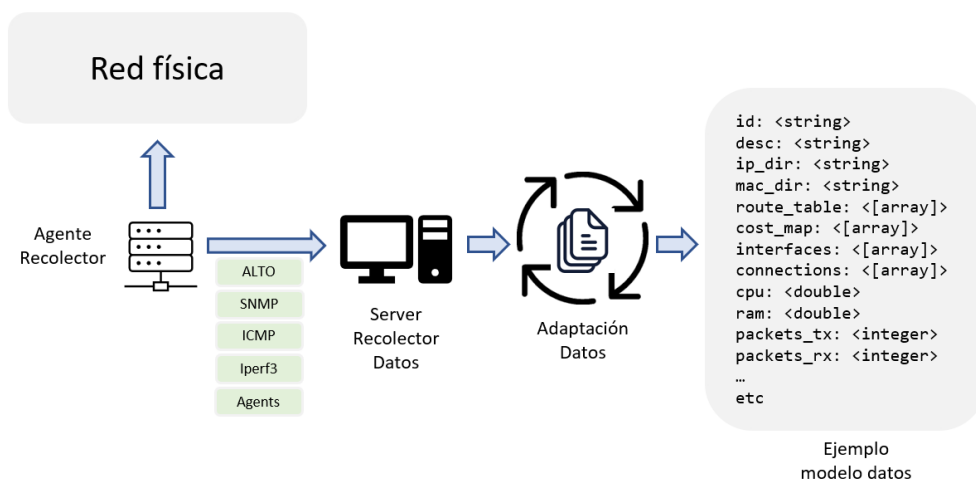


Figura 29. Modelado de la información recolectada

En el capítulo cuatro se expondrán diferentes casos de aplicabilidad de la metodología en los cuales se verá reflejado el resultado de modelado con el fin de tratar de simplificar redundancias de red y la agrupación de segmentos de la red caracterizándolos como un único segmento que cumpla por ejemplo con las características de retardo agregado.

3.1.3. Fase III. Adaptación

Tras la recolección y procesamiento de datos desde la red física, comienza la fase de adaptación de la información para poder desplegar el gemelo digital de red en el entorno digital. Antes de entrar en detalle en el procesamiento de adaptación, es importante destacar

que el despliegue del gemelo digital de red se basa en la tecnología NFV, concretamente en framework de gestión y orquestación OpenSourceMANO (OSM). Dicho framework trabaja principalmente con dos elementos fundamentales, los Network Services (NS) y las Virtual Network Functions(VNF).

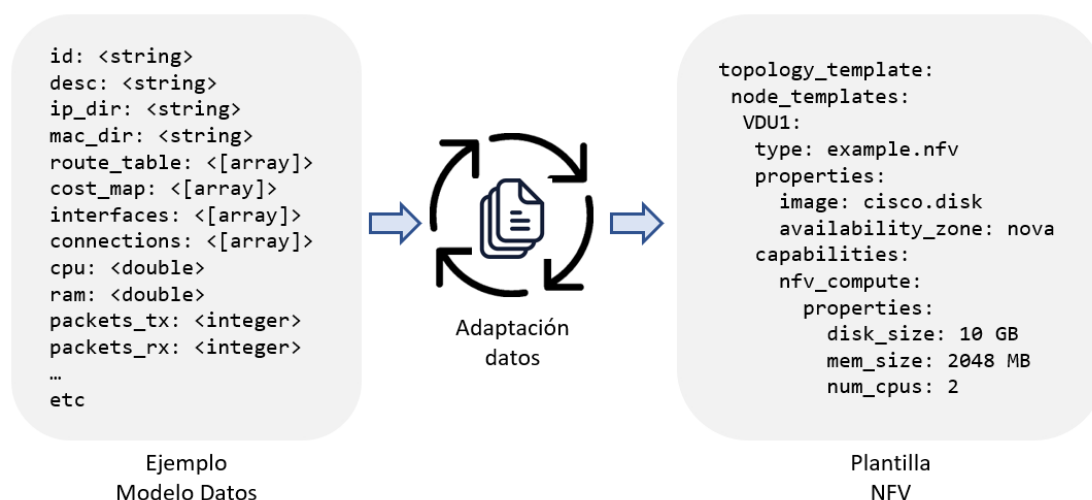


Figura 30. Adaptación del formato de datos a plantillas NFV

Las NS representan un conjunto de funciones de red interconectadas que permiten desplegar servicios completos a través de la red. Estas pueden ser desde servicios simples, como un firewall o un enrutador, hasta servicios más complejos, como redes privadas virtuales (VPN) o gestión de ancho de banda. La clave de las NS radica en su capacidad para orquestar múltiples VNF y otros elementos de la red para proporcionar servicios completos y específicos.

Por otro lado, las VNF son componentes individuales que desempeñan funciones específicas dentro de una red o virtualizan elementos concretos de la misma. Estas funciones, que tradicionalmente se ejecutaban en hardware dedicado, ahora se implementan mediante software y se ejecutan en entornos virtualizados. Las VNF abarcan una amplia gama de funcionalidades, como enrutamiento, balanceo de carga, inspección de paquetes, optimización de red y mucho más. Su flexibilidad y escalabilidad son aspectos clave, permitiendo su despliegue dinámico y ajuste según las necesidades de la red.

La flexibilidad, la escalabilidad y la eficiencia operativa son características centrales tanto de las NS como de las VNF. Su adopción ha revolucionado la forma en que se diseñan,

implementan y gestionan las redes, brindando a los operadores la capacidad de ofrecer servicios ágiles y adaptativos en un entorno de red dinámico y en constante evolución, siendo además una tecnología habilitadora para el despliegue de gemelos digitales de red.

3.1.3.1. Modelo de información

Para llevar a cabo la implementación del NDT a través de la tecnología NFV, tras el estudio realizado en el estado del arte, se adopta el uso del framework para la gestión y orquestación OSM. Dicho framework, desde su versión nueve define un modelo de información el cual está completamente alineado con la especificación ETSI NFV SOL006. Esta especificación define tanto los requisitos como las especificaciones para garantizar la interoperabilidad de los sistemas de gestión y orquestación (MANO) con las interfaces de programación de aplicaciones (API) de las infraestructuras NFV, por ello, su objetivo principal es asegurar la compatibilidad entre diferentes soluciones NFV.

Dentro del Information Model (IM) de OSM, se puede encontrar la información necesaria para llevar a cabo la implementación de los NS y sus VNFs asociadas a través de lo que se conoce como ficheros descriptores, NSD [160] y VNFD [161] respectivamente. Pese a que la información reflejada en el IM es amplia, permitiendo realizar configuraciones específicas en función del objetivo final, *la Tabla 10 y Tabla 11* muestran los elementos mínimos que han de tener los ficheros NSD y VNFD respectivamente.

Tabla 10. Elementos básicos del descriptor NSD

NSD	
Elemento	Descripción
nsd	Objeto principal del descriptor, con características como id, nombre, versión, etc.
constituent-vnfd	Permite especificar que VNFs están definidas en el NSD a través del nombre del VNFD
vld	Virtual Link Descriptor, permite la creación de diferentes redes para conectar los distintos connection-points definidos en los VNFDs

Tabla 11. Elementos básicos del descriptor VNFD

VNFD	
Elemento	Descripción
vnfd	Objeto principal del descriptor, el resto de los elementos dependen de este. Incluye campos como id, nombre, versión, etc
mgmt-interface	Cada VNFD necesita una interfaz de gestión, aunque la VNF no esté conectada a una red con acceso a internet.
vdu	Virtual Device Unit. Elemento encargado de definir las características virtuales de la VNF como los vcpus, RAM, disco, imagen, interfaces, etc
connection-point	Establece un punto de referencia utilizable para interconectar diferentes VNFs.

Los ficheros NSD en OSM tienen un rol fundamental, el cual se encarga de describir la topología y los requerimientos de un servicio de red virtualizado. Este descriptor actúa como un plano de diseño para el despliegue y la operación de servicios de red en entornos virtualizados. Detalla la estructura, configuración y relaciones entre los diferentes elementos que componen el servicio de red.

El NSD en OSM proporciona una representación detallada de los aspectos funcionales y de conectividad del servicio de red. Incluye la información referente a las VNFs que componen el servicio, sus relaciones y conexiones, los requisitos de red y los parámetros de configuración necesarios para desplegar correctamente el servicio.

A través del NSD, el ecosistema planteado en la tesis doctoral permite establecer definir las características del servicio de manera precisa en base a los datos recuperados en las fases previas, estableciendo las reglas de conexión, los flujos de datos y los recursos necesarios para garantizar su funcionamiento. Esto permite una implementación ligera y eficiente del gemelo digital de red ya que el NSD actúa como un plano guía para la orquestación y el despliegue en infraestructuras virtualizadas.

En el caso de los ficheros VNFD en OSM, se encargan de describir las características técnicas y funcionales de una Función de Red Virtual (VNF). Este descriptor actúa como un documento detallado que especifica los requisitos de implementación y

funcionamiento de cada elemento de red presente en el gemelo digital a implementar, incluyendo aspectos como la configuración, requisitos de recursos, interfaces de red, comportamiento funcional y cualquier otro parámetro relevante para su despliegue y operación en un entorno virtualizado. Describe los componentes y capacidades específicas de la VNF, así como los detalles sobre sus interfaces de comunicación y las dependencias de red. Al generar los ficheros VNFD, el ecosistema NDT tiene una visión clara de cómo debe instanciar y configurar cada DT ya que el VNFD actúa como una guía para la orquestación y el despliegue de estas funciones dentro de una infraestructura virtualizada.

En la **Figura 31** se puede observar un ejemplo simplificado de ambos descriptores.

```

nsd:
  nsd:
    - description: red5G
      designer: Mario Sanz Rodrigo
      df:
        - id: default-df
          vnf-profile:
            - id: ue
              virtual-link-connectivity:
                - constituent-cpd-id:
                  - constituent-base-element-id: ue
                    constituent-cpd-id: mgmt-ext
                  virtual-link-profile-id: public
                - constituent-cpd-id:
                  - constituent-base-element-id: ue
                    constituent-cpd-id: ue_stgutg_net
                    ip-address: 10.60.0.1
                  virtual-link-profile-id: ue_stgutg_datanet
              vnf-id: ue
            - id: free5gc
              virtual-link-connectivity:
                - constituent-cpd-id:
                  - constituent-base-element-id: free5gc
                    constituent-cpd-id: mgmt-ext
                  virtual-link-profile-id: public
                - constituent-cpd-id:
                  - constituent-base-element-id: free5gc
                    constituent-cpd-id: stgutg_free5gc_net
                    ip-address: 192.168.60.11
                  virtual-link-profile-id: stgutg_free5gc_datanet
                - constituent-cpd-id:
                  - constituent-base-element-id: free5gc
                    constituent-cpd-id: free5gc_dn_net
                    ip-address: 192.168.99.1
                  virtual-link-profile-id: free5gc_dn_datanet
              vnf-id: free5gc

          virtual-link-profile:
            - id: ue_stgutg_datanet
              virtual-link-desc-id: ue_stgutg_datanet
              virtual-link-protocol-data:
                l3-protocol-data:
                  cidr: 10.60.10.0/24
                  ip-version: ipv4
                  #gateway-ip: 13.10.10.1
                  #dhcp-enabled: 'true'

          id: red5G
          name: red5G
          version: 1.0
          virtual-link-desc:
            - id: public
              mgmt-network: 'true'
            - id: free5gc_dn_datanet
              mgmt-network: 'false'

vnfd:
  description: UE - User Equipment
  designer: Mario Sanz Rodrigo
  df:
    - id: default-df
      instantiation-level:
        - id: default-instantiation-level
          vdu-level:
            - number-of-instances: 1
              vdu-id: ue-vdu
          vdu-profile:
            - id: ue-vdu
              min-number-of-instances: 1
      ext-cpd:
        - id: mgmt-ext
          int-cpd:
            cpd: eth0-int
            vdu-id: ue-vdu
        - id: ue_stgutg_net
          int-cpd:
            cpd: eth1-int
            vdu-id: ue-vdu
      id: ue
      mgmt-cp: mgmt-ext
      product-name: ue
      provider: UPM_tesis
      sw-image-desc:
        - id: ubuntu18.04
          image: ubuntu18.04
          name: ubuntu18.04
      vdu:
        - id: ue-vdu
          name: ue-vdu
          description: ue-vdu
          cloud-init-file: cloud-config.txt
          sw-image-desc: ubuntu18.04
          virtual-compute-desc: ue-vdu-compute
          virtual-storage-desc:
            - ue-vdu-storage
          int-cpd:
            - id: eth0-int
              virtual-network-interface-requirement:
                - name: eth0
                  virtual-interface:
                    type: PARAVIRT
            - id: eth1-int
              virtual-network-interface-requirement:
                - name: eth1
                  ip-address: 10.60.0.1
                  virtual-interface:
                    type: PARAVIRT
          version: 1.0
          virtual-compute-desc:
            - id: ue-vdu-compute
              virtual-cpu:

```

Figura 31. Ejemplo de fichero NSD y VNFD

3.1.3.2. Adaptación del modelo de datos JSON a la especificación SOL006

Para llevar a cabo la fase de adaptación del modelo de datos a la especificación NFV SOL006, se hace uso del subsistema de adaptación, una pieza software basada en Jinja2, el cual se encarga de generar los descriptores necesarios y preparar su subida a la plataforma OSM que se encargará del despliegue.

El argumento de entrada para el subsistema de adaptación es el modelo de datos JSON elaborado en la fase de recolección de datos. En un primer instante, en el cual no existe la NDT, el modelo refleja información acerca de las interfaces, conexiones y redes alcanzables. Esta información será de vital importancia para poder crear los ficheros NSD que describirán la topología del NDT.

El objetivo final de esta fase es, en primer lugar, la obtención de un fichero NSD el cual se encargará de desplegar el NDT, seguidamente de la creación de los diferentes ficheros VNFD, a partir de una plantilla genérica, para cada elemento presente en la red física, pudiendo llegar a simplificar enlaces tras la fase de modelado, con el fin de obtener un NDT computacionalmente más ligero, evitando así desplegar un NDT que replique completamente toda la red física. En la plantilla genérica de VNFD se incluyen aquellos campos obligatorios según el IM, así como los campos adicionales que permiten mapear las características de la red, utilizando variables que se sustituyen por los datos de entrada al ejecutar el subsistema de adaptación. A continuación, se muestra el diagrama de flujo que realiza el subsistema de adaptación, seguido de un ejemplo con el fin de explicar el IM manejado por OSM se va a usar el descriptor correspondiente a un elemento de red de ejemplo, identificado como router R1.

```

Inicio
├─▶ Cargar datos desde archivo JSON
├─▶ Crear carpeta "vnf_descriptors" si no existe
|   ├─▶ Crear lista de VNFs vacía
|   ├─▶ Cargar plantilla VNFD
|   └─▶ Para cada elemento de red
|       ├─▶ Añadir nombre del elemento a la lista de VNFs
|       ├─▶ Crear carpeta "vnf_descriptors/<elemento>" si no existe
|       ├─▶ Crear archivo "<elemento>_nfd.yaml" en la carpeta correspondiente
|       |   └─▶ Renderizar plantilla VNFD con los datos del elemento
|       └─▶ Crear carpeta "vnf_descriptors/<elemento>/cloud_init"
|           └─▶ Copiar archivo "cloud-config.txt" a la carpeta correspondiente
├─▶ Crear carpeta "ns" si no existe
|   ├─▶ Crear archivo "<elemento>_nsd.yaml" en la carpeta "ns"
|   └─▶ Renderizar plantilla NSD con los datos completos
├─▶ Subir paquetes a OSM
|   └─▶ Para cada VNF en la lista de VNFs
|       └─▶ Crear paquete VNF en OSM con la ruta "vnf_descriptors/<VNF>"
|           └─▶ Crear paquete NS en OSM con la ruta "ns"
Fin

```

En primer lugar, se especifica un deployment flavour básico con los requisitos de despliegue:

```

df:
- id: default-df
  instantiation-level:
  - id: default-instantiation-level
    vdu-level:
    - number-of-instances: 1
      vdu-id: R1-vdu
  vdu-profile:
  - id: R1-vdu
    min-number-of-instances: 1

```

Después se obtienen todos los puntos de conexión externos, uno por cada red a la que se conecta el router, asociando a cada uno de ellos un identificador y la referencia al punto de conexión interno. Adicionalmente se incluye una conexión con la red de gestión.

```

ext-cpd:
-   id: mgmt-ext
    int-cpd:
      cpd: eth0-int
      vdu-id: R1-vdu
-   id: 11.10.10.0/24-ext
    int-cpd:
      cpd: eth1-int
      vdu-id: R1-vdu

```

Con el fin de realizar una gestión ordenada, cada VNF debe tener un identificador único, así como una referencia al punto de conexión de la red de gestión y los recursos asignados a las VDU que componen la VNF.

```

id: R1
mgmt-cp: mgmt-ext
product-name: R1
provider: OSM
sw-image-desc:
-   id: ubuntu18.04
    image: ubuntu18.04
    name: ubuntu18.04
[...]
virtual-compute-desc:
-   id: R1-vdu-compute
    virtual-cpu:
      num-virtual-cpu: 1
    virtual-memory:
      size: 1.0
virtual-storage-desc:
-   id: R1-vdu-storage
    size-of-storage: 10
vdu:
-   id: R1-vdu
    name: R1-vdu
    description: R1-vdu
    cloud-init-file: cloud-config.txt
    sw-image-desc: ubuntu18.04
    virtual-compute-desc: R1-vdu-compute
    virtual-storage-desc:
      - R1-vdu-storage

```

Adicionalmente, es posible incluir en el VNFD la referencia a un fichero cloud-init para realizar configuraciones iniciales en la máquina (Day-0, pudiendo por ejemplo definir una contraseña con la que acceder a las instancias).

```
#cloud-config
password: 1234
chpasswd: { expire: False }
ssh_pwauth: True
```

Dicho fichero debe alojarse en una carpeta llamada “*cloud_init*” en el directorio del descriptor, de cara a ser incorporado en el paquete VNFD que se sube a OSM.

El último elemento que general el subsistema de adaptación dentro de cada VDU es una lista de los puntos de conexión internos que corresponden a las interfaces de cada elemento de red. Aparte del identificador, nombre y tipo de la interfaz (PARAVIRT para usar la interfaz de paravirtualización por defecto).

```
int-cpd:
[...
-   id: eth1-int
    virtual-network-interface-requirement:
      -   name: eth1
          ip-address: 11.10.10.10
          virtual-interface:
            type: PARAVIRT
```

Una vez generado los diferentes ficheros VNFD, el subsistema pasa a generar el fichero NSD que orquestará todo el NDT. Para identificar al NSD, este debe incluir un identificador único y puede incluir otros campos generales como nombre, descripción, versión y proveedor:

```
nsd:
  nsd:
    - description: red5G
      designer: OSM
[...
  id: red5G
  name: red5G
  version: 1.0
```

A esto le siguen una lista de todas las VNF que componen el NDT (identificadas por el id indicado en el VNFD) y otra lista con todos los enlaces virtuales que conectas están VNF. Hay que especificar cuál de ellos corresponde a la red de gestión en el campo “*mgmt-network*”:

```
nsd:
  nsd:
  [...]
    virtual-link-desc:
      - id: mgmt
        mgmt-network: 'true'
      - id: datanet11
        mgmt-network: 'false'
      [...]
    vnfd-id:
      - R1
      - R2
```

El bloque de *deployment flavour* (df) se divide ahora en dos grandes, uno para las características de las VNF (*vnf-profile*) y otro para las de los enlaces virtuales (*virtual-link-profile*). El bloque de *vnf-profile* es una lista de las VNF del servicio en la que para cada una se describe como está conectada a los distintos enlaces. Para definir la conexión se listan todos los puntos de conexión de la VNF, incluyendo la VNF a la que pertenecen, una referencia al punto de conexión externo del VNFD, una referencia al VL al que se conecta y, en su caso, la dirección IP asignada a esa interfaz.

```
df:
- id: default-df
  vnf-profile:
  - id: R1
    virtual-link-connectivity:
      - constituent-cpd-id:
          - constituent-base-element-id: R1
            constituent-cpd-id: mgmt-ext
            virtual-link-profile-id: mgmt
      - constituent-cpd-id:
          - constituent-base-element-id: R1
            constituent-cpd-id: 11.10.10.0/24-ext
            ip-address: 11.10.10.10
            virtual-link-profile-id: datanet11
```

```

    [...]
    vnfd-id: R1
  - id: R2
    virtual-link-connectivity:
      - constituent-cpd-id:
          - constituent-base-element-id: R2
            constituent-cpd-id: mgmt-ext
        virtual-link-profile-id: mgmt
    [...]
    vnfd-id: R2

```

En este caso el campo *id* tiene un valor arbitrario, mientras que el campo *vnfd-id* debe corresponder al indicado en el VNFD.

El bloque de *virtual-link-profile* está formado por los VL para los que se quiere configurar requisitos adicionales. En nuestro caso, no se quiere copiar solo la topología de la red física si no también incluir sus características de direccionamiento. En consecuencia, para cada una de las redes de datos se debe configurar el CIDR, la dirección del *gateway* y habilitar el protocolo DHCP. Para cada una se crea un perfil similar a este:

```

df:
- id: default-df
  [...]
  virtual-link-profile:
  - id: datanet11
    virtual-link-desc-id: datanet11
    virtual-link-protocol-data:
      l3-protocol-data:
        cidr: 11.10.10.0/24
        ip-version: ipv4
        gateway-ip: 11.10.10.1
        dhcp-enabled: 'true'

```

3.1.3.3. Validación de los descriptores

El subsistema de adaptación se encarga de generar la estructura de carpetas que se muestra en la **Figura 32**, donde se van creando y depositando los ficheros a medida que se procesa el fichero JSON que contiene el modelo de datos que refleja la red física. Una vez el subsistema de adaptación finaliza de analizar el fichero JSON en primer lugar devuelve como salida los VNFD y NSD, los cuales contienen el descriptor en formato YAML

y un fichero de comprobación “checksum.txt” asociado, para seguidamente realizar una validación de la sintaxis de los ficheros con el fin de generar correctamente los paquetes en formato tar.gz requeridos por el repositorio de NSD y VNFD de OSM. Adicionalmente, el paquete puede contener elementos como scripts para la fase de aprovisionamiento, charms, iconos o ficheros de configuración.

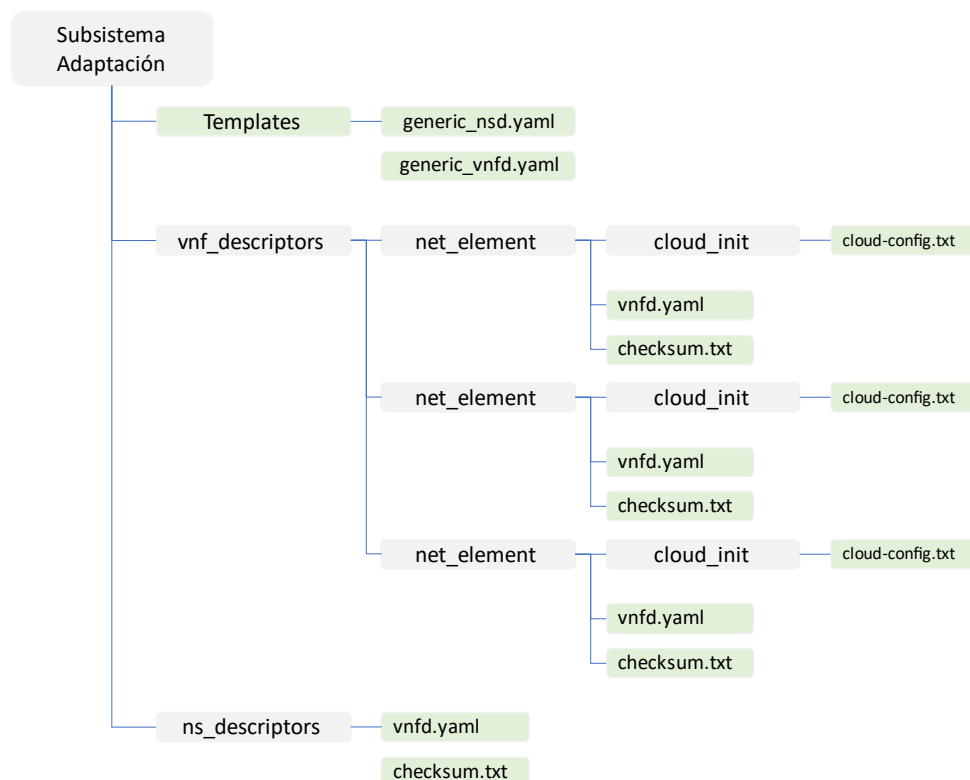


Figura 32. Estructura de carpetas generada por el subsistema de adaptación

3.1.4. Fase IV. Despliegue NFV

Tras la finalización de la fase de adaptación, se activa la fase de despliegue del NDT haciendo uso de la tecnología NFV. En este contexto, las plataformas MANO desempeñan un papel crucial en lo que a control y gestión de recursos virtualizados se refiere. En la fase de despliegue de la metodología propuesta se hace uso de implementación OSM, encargada de controlar tanto el despliegue como el ciclo de vida de la NDT.

En lo referente al despliegue, esta fase se encarga de llevar a cabo las siguientes tareas:

- **Recepción y almacenamiento de los descriptores.** Una vez generados los paquetes *vnfd.tar.gz* y *nsd.tar.gz* que contienen los descriptores de servicio que conformarán la

NDT a desplegar, se almacenan en el repositorio de OSM a la espera de realizar el despliegue.

- **Orquestación de recursos.** A través de los ficheros NSD, se realiza la solicitud y asignación de recursos específicos asociados al NDT en función del VIM registrado en OSM.
- **Automatización del despliegue.** A través de los ficheros VNFD, se realiza de manera automática la configuración y despliegue del NDT en términos de configuración de redes, instalación de software, etc necesario para el funcionamiento esperado.

En lo referente al ciclo de vida, OSM se encarga de llevar a cabo las siguientes tareas:

- **Instanciación y configuración inicial.** En esta fase se instancian las NS a partir de los paquetes que se han diseñado y cargado en las fases previas. OSM ofrece una API que permite operaciones CRUD sobre las instancias NS. Estas operaciones, a diferencia de la fase previa, no consisten en cargar y validar los paquetes NS, sino que son operaciones sobre las instancias que se crean a partir de los paquetes. Para el caso de una creación o instanciación de una NS, OSM interactúa con varias plataformas de servicio de niveles inferiores para crear el objeto de servicio de la instancia NS. En esta fase se distinguen dos tipos de operaciones en la terminología NFV. Por un lado, las operaciones Day-0, que abarcan la creación y puesta en marcha de las instancias. Por otro lado, las operaciones Day-1, que consisten en la ejecución de primitivas de forma automática inmediatamente después de la creación de las instancias.
- **Gestión continua.** Supervisan constantemente el rendimiento, analizan las métricas y responden a los cambios en la carga o en el estado de la red. Permitiendo realizar acciones de mantenimiento preventivo o correctivo según sea necesario.
- **Operación y optimización.** En esta fase, la instancia NS se convierte en el único objeto relevante para las operaciones restantes. Estas operaciones se pueden dividir en tres categorías, las cuales son ejecutadas de manera asíncrona. En primer lugar, se distinguen las operaciones comunes aplicables a cualquier NS. En segundo lugar, operaciones de tipo Day-2, las cuales se ejecutan sobre alguna funcionalidad específica que ofrece una determinada instancia NS. Usualmente suelen ser solicitadas por el cliente y pueden consistir en primitivas de configuración, de monitorización, etc. Por último, las operaciones internas son operaciones que no tiene por qué solicitarlas el cliente, pero que se ejecutan siguiendo unas políticas internas. Adicionalmente,

dentro de la monitorización, OSM permite varias métricas y alarmas para elementos relacionados con el nivel VDU-PDU o específicas del nivel de aplicación. También se permite realizar alarmas bajo demanda.

- **Terminación.** Esta fase es la última del ciclo de vida de las instancias NS. Cuando se inicia esta fase comienza el proceso de finalización de la instancia NS. Finalizar una instancia NS libera los recursos que se le había asignado.

3.1.5. Fase V. Aprovisionamiento del NDT

Tal y como se ha mencionado anteriormente en la fase de despliegue, es necesario llevar a cabo una serie de acciones de aprovisionamiento para garantizar el correcto despliegue y configuración del NDT. Pese a que ambas fases se tratan como fases diferenciadas, hay ciertas acciones relacionadas con el aprovisionamiento que se definen en las fases previas, ya que son requerimientos asociados al despliegue. A continuación, se revisarán las tres maneras de aprovisionamiento contempladas en la metodología propuesta, siendo las dos primeras mecanismos con una fuerte dependencia tanto de la fase III como de la fase IV, las cuales se podrían considerar como el aprovisionamiento mínimo necesario para cumplir funcionalidades básicas, dejando el tercer mecanismo como un aprovisionamiento más detallado, el cual se realiza una vez está el NDT desplegado.

3.1.5.1. Basado en cloud-init

La tecnología *cloud-init* permite la opción de personalizar las máquinas durante su instanciación y puesta en marcha. Tal y como se explicó en la fase de despliegue, los paquetes *tar.gz* generados con los descriptores VNFD permiten la inclusión de ficheros *cloud-config*, los cuales pueden establecer configuraciones mínimas como el *hostname*, usuario o contraseña asociada a cada instancia. En el contexto de OSM y NFV, esta configuración basada en *cloud-init* correspondería a las operaciones de tipo Day-1, las cuales engloban instalación de paquetes, dependencias, edición de archivos de configuración, ejecución de comandos, etc.

Tal y como establece OSM, es necesario que el fichero de configuración *cloud-init* se referencie desde el descriptor VNFD, concretamente en el subapartado VDU.

Este tipo de aproximación de aprovisionamiento es muy útil sobre todo cuando las instancias a desplegar son imágenes base como las ofrecidas por Ubuntu Cloud, las cuales necesitan ser configuradas con el software necesario. No obstante, también es una aproximación válida para el uso de instancias basadas en imágenes customizadas, las cuales únicamente requieren de modificación o adaptación de parámetros relevantes a la propia instancia, como por ejemplo el hostname, activación o renombrado de interfaces, etc.

3.1.5.2. Basado en primitivas Juju Charms

Juju es una herramienta de modelado Open Source pensada para operar software en la nube. Permite la implementación, configuración, administración y mantenimiento de aplicaciones de manera rápida y eficiente tanto en nubes públicas como nubes privadas, tanto basadas en máquinas o contenedores. En el contexto de OSM se establece que un charm es un conjunto de scripts encargados de la implementación y operación de software. Estos scripts engloban la lógica operativa de un software y son capaces de ejecutar acciones para la gestión de la configuración y el ciclo de vida de una VNF, salvo la fase de instanciación y eliminación.

Tal y como se ha visto anteriormente, OSM establece tres fases diferenciadas para la configuración y gestión del ciclo de vida de las VNF, fases de tipo Day-0, Day-1 y Day-2.

En el caso de los Charms de Juju, se cubren las configuraciones asociadas a Day-1 y Day-2, ejecutando las primeras de manera automática tras realizar la instanciación y dejando las segundas disponibles bajo demanda. Al igual que en el aprovisionamiento basado en cloud-init, las acciones Day-1 realizadas por los Charms juju cubren las tareas de configuración, instalación de paquetes, ejecución de software, etc. Dejando para las acciones Day-2 tareas asíncronas como la activación de software encargado de capturar tráfico, modificaciones en la configuración, etc.

3.1.5.3. Basado ANSIBLE

Por último, la metodología plantea la alternativa de realizar un aprovisionamiento basado en ANSIBLE, debido a su flexibilidad para la configuración de entornos de red. Debido a su enfoque basado en *Infrastructure as Code* (IaC) permite describir el estado deseado del NDT en vez de establecer pasos específicos para alcanzar dicho estado.

El proceso comienza con la creación de un inventario de hosts, siendo en este caso el inventario de las diferentes instancias que conformarán el NDT tras la fase de captura de datos topológicos del entorno real. Seguidamente, para realizar las configuraciones avanzadas se hace uso de “*playbooks*” específicos, los cuales, en conjunto con roles, establecen las tareas necesarias para configurar dispositivos, desde la configuración de interfaces de red hasta la gestión de servicios. Este aprovisionamiento se realiza de manera remota desde un nodo controlador presente en el sistema de gestión el cual tiene conectividad con el entorno digital donde se ha desplegado el NDT. La flexibilidad, reutilización de roles y la posibilidad de automatizar tareas clave hacen de Ansible una opción atractiva para mantener gemelos digitales de red. La modularidad inherente a Ansible simplifica el mantenimiento y escalabilidad a medida que el entorno digital evoluciona, ofreciendo así una automatización integral que reduce errores manuales y asegura la coherencia y fiabilidad en la configuración de la red.

3.1.6. Fase VI. Interconexión de gemelos

Tal y como establecen los autores en [30], [162], [163], así como la ETSI, uno de los elementos característicos más importantes del paradigma de gemelos digitales es la comunicación entre ambos mundos, el digital y el virtual, permitiendo la coevolución de ambos gemelos en tiempo real.

En esta fase de la metodología, una vez desplegado el NDT, se aborda la tarea de realizar las diferentes interconexiones y la diferenciación en flujos de información para habilitar el intercambio de datos entre los gemelos. Tal y como se muestra en la **Figura 33**, existen diferentes flujos de información debidamente identificados, así como una serie de elementos diseñados en el contexto de la metodología que facilitan la implementación de este requerimiento. Estos elementos, los cuales realizan acciones similares, se encuentran ubicados en diferentes partes de la arquitectura asociada al ecosistema NDT, distinguiendo tres zonas diferenciadas, el mundo físico, donde se encuentra la red real objetivo y los agentes recolectores, el sistema de gestión encargado de la adaptación de datos, el despliegue, aprovisionamiento y el almacenado de la información intercambiada por los distintos gemelos y el mundo digital o virtual, donde se encuentra la NDT desplegada. A continuación, se realiza una explicación detallada de cada uno de estos elementos:

- **Conector DANA.** Se trata de un submódulo presente en el sistema DANA desplegado en la arquitectura física, los cuales fueron presentados en la fase I de la metodología. Este submódulo actúa como una puerta de enlace o *gateway* permitiendo comunicar el mundo real con el sistema de gestión del NDT y los gemelos digitales presentes en el.
- **Conector BBDD.** Se trata de un subsistema presente en el sistema global de gestión, el cual cumple tres funciones principales. La primera, albergar el broker MQTT y el sistema de ficheros encargado de recibir los archivos formato pcap recogidos por el sniffer presente en la red física, encargados de recibir la información por los clientes publicadores/suscriptores presentes en el ecosistema NDT, la segunda consiste en suscribirse a toda la información intercambiada, tanto en sentido físico a digital como digital a físico, para almacenarla en base de datos, de manera que sea posible consultar el histórico de comportamiento del sistema global. Por último, este subsistema cumple con una función de control sobre la comunicación, evitando que la información fluya libremente del mundo físico al digital sin un control previo. Esta decisión de diseño viene principalmente motivada por temas de seguridad, dado que se pretende evitar que los fallos, ataques o malos funcionamientos intencionados en uno de los mundos pueda afectar a los gemelos presentes en el otro.
- **Conector DT.** Se trata de un submódulo similar al conector DANA presente en el mundo físico. Al igual que el primero, actúa como puerta de enlace permitiendo la comunicación con el sistema de gestión y el gemelo físico, así como habilitar la recepción de datos provenientes del gemelo físico.

Dentro de la información intercambiada o recolectada en los gemelos, diferenciamos dos grandes tipos de flujos, cada uno asociado a un protocolo para permitir su envío dentro del ecosistema NDT, MQTT y SCP.

- **Flujos basados en MQTT.** Estos flujos están ligados al modelo de datos presentado en las fases iniciales de la metodología, el cual alberga datos de dos tipologías diferentes, información referente a la topología e información referente al comportamiento de los elementos físicos. Estos flujos no tienen una necesidad de funcionamiento en tiempo real, por lo que el protocolo MQTT se toma como un habilitador capaz de llevar a cabo correctamente la interconexión, haciendo uso de

clientes tanto en modalidad publicador (encargados de enviar información) como en modalidad suscriptor (encargados de recibir información) tal y como se desglosa en la **Figura 33**. Se aprovecha la ventaja que ofrece MQTT a la hora de poder utilizar herencia en la definición de topics, permitiendo así poder suscribirse o publicar en topics padre, comunes a todos o a un gran número de elementos. Aparte del uso de MQTT para el flujo intercambiado entre el gemelo físico y el gemelo digital, también se hace uso de la capacidad de suscripción de los agentes para poder establecer flujos de gestión capaces de modificar de manera centralizada el funcionamiento de los agentes, por ejemplo, activando o desactivando los *sniffer* de red, o notificando sobre el cambio de estado asociado a los agentes para permitir la readaptación del NDT en caso de detectar cambios en la topología de la red real.

- **Flujos basados en SCP.** Estos flujos, a diferencia de los flujos ligados al protocolo MQTT, están ligados a flujos de tráfico capturado tanto en el mundo digital o en el mundo físico. Estos flujos de tráfico tienen como objetivo servir como datasets para poder ser utilizados por sistemas de procesamiento avanzado como pueden ser algoritmos de Machine Learning o sistemas basados en Inteligencia Artificial, los cuales facilitarían la tarea de analizar el comportamiento de la red con el objetivo de detectar, por ejemplo, problemas de seguridad, problemas en el rendimiento, etc.

Adicionalmente ambos flujos permiten la configuración segura basada en SSL/TLS, algo de vital importancia dado la sensibilidad de los datos intercambiados dentro de un ecosistema NDT, ya que típicamente estamos modelando redes en producción para poder obtener su gemelo digital conectado.

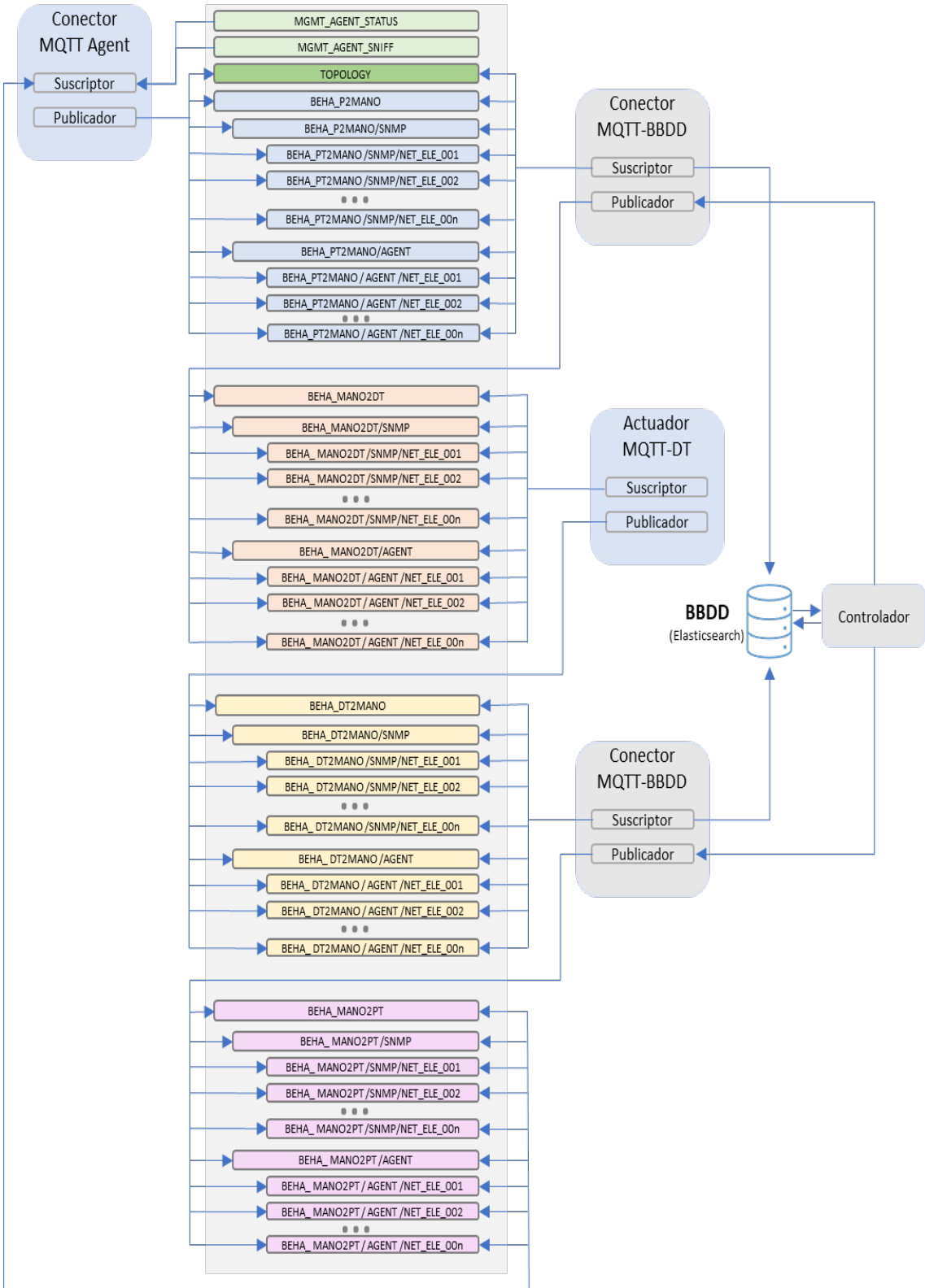


Figura 33. Interconexión de gemelos basada en flujos de comunicación

3.1.7. Fase VII. Realimentación del sistema NDT

La última fase de la metodología propuesta consiste en activar un circuito de retroalimentación entre los dos gemelos, utilizando la conexión bidireccional introducida en la fase anterior. Esta fase completa el estado 0, asociado con la creación automática de los gemelos digitales de la red, y pasa automáticamente al estado 1, en el que los agentes desplegados en los entornos gemelos comienzan a monitorear tareas para intercambiar información con el objetivo realizar tareas de nivel superior, como optimización de topología, adquisición de conjuntos de datos, prevención. acciones en situaciones estresantes en la red, etc.

Tal y como se introdujo en la fase anterior, en la **Figura 33** se definen una serie de elementos presentes en el sistema de gestión encargado de controlar las comunicaciones del ecosistema NDT. Esta información es almacenada en base de datos haciendo uso de los conectores MQTT-BBDD y conectando un elemento controlador capaz de recuperar esa información con dos propósitos diferenciados. El primero de ellos, simplemente se encarga en propagar la información al gemelo correspondiente, identificando por ello los siguientes cuatro tramos:

- Comunicación gemelo físico a plataforma MANO
- Comunicación plataforma MANO hacia el gemelo digital
- Comunicación gemelo digital a plataforma MANO
- Comunicación plataforma MANO a gemelo físico.

El segundo propósito tiene como objetivo proveer la información a capas superiores de aplicación integrables con el ecosistema NDT como pueden ser sistemas para el soporte a la toma de decisiones, sistemas de seguridad como IDs, SIEMs, elementos OSS/BSS o sistemas basados en Inteligencia Artificial o Machine Learning capaces de ejecutar algoritmos encargados de optimizar los procesos, configuraciones o topologías presentes en los gemelos.

Capítulo 4

Plataforma para el soporte del NDT

Para llevar a cabo la implementación de la metodología propuesta en el capítulo anterior, con el objetivo de validar el correcto funcionamiento de las distintas fases, así como identificar posibles limitaciones o mejoras, es necesario el establecimiento de un framework completo encargado de modelar y desplegar el gemelo digital de red. Pese a que distintos organismos como el IETF o el DTC plantean arquitecturas de referencia, muchas de ellas únicamente plantean las bases mínimas para poder llevar a cabo una implementación técnica del NDT, sin llegar a existir una plataforma estandarizada que permita llevar a cabo esta tarea de manera automática.

Tal y como se ha ido revisando a lo largo de los capítulos previos, el paradigma de gemelos digitales tiene una fuerte dependencia con el caso de uso final al que se aplica. Si bien es cierto que existen soluciones basadas en Digital Twins, como las planteadas en la **Tabla 2**, estas están mayoritariamente enfocadas a la tecnología IoT, la cual tiene una serie de requisitos para el funcionamiento y despliegue sustancialmente diferentes a los que plantea una red de comunicación de última generación como pueden ser las redes 5G en modo Stand Alone.

En resumen, el presente capítulo describe en detalle la plataforma diseñada y desarrollada la cual permite llevar a cabo la implementación técnica de la metodología propuesta en el capítulo 3, diferenciando claramente tres entornos o elementos dentro del ecosistema NDT, el entorno del gemelo físico, el sistema de gestión NDT y el entorno de gemelo digital. Para ello se toma como punto de partida la arquitectura de referencia

propuesta por el IETF, haciendo uso de tecnologías Open Source que permitan tanto el correcto funcionamiento, como cubrir todas y cada una de las fases planteadas en la metodología, alineadas al ciclo de vida del ecosistema basado en DT.

4.1. Diseño de la plataforma

En esta sección se determinan los requisitos definidos para la plataforma. A partir de ellos y de la metodología planteada en el capítulo anterior, se describe la arquitectura , identificando cada uno de los subsistemas que la conforman.

4.1.1. Requisitos de la plataforma

En este apartado se presentan los principales requisitos funcionales asociados a la plataforma capaz de llevar a cabo la implementación técnica de la metodología propuesta. Estos requisitos se han extraído principalmente en base al estudio del estado del arte relacionado, incluyendo los diferentes drafts propuestos por el IETF, los trabajos incipientes realizados sobre la temática y las tecnologías habilitadoras disponibles.

A partir de estas fuentes, se clasifican los requisitos en cinco grupos:

- **Requisitos relacionados al paradigma de gemelos digitales**

La **Tabla 12** refleja los requisitos relacionados al paradigma de gemelos digitales en base a las características y necesidades específicas identificadas en la **sección 0** , del estado del arte.

Tabla 12. Requisitos relacionados al paradigma de gemelos digitales

Id.	Descripción
R1.a	La plataforma debe ser capaz de llevar a cabo el modelado y adaptación de la información para desplegar el gemelo digital de la red objetivo
R1.b	La plataforma debe ser capaz de gestionar el ciclo de vida completo de las instancias DT
R1.c	La plataforma debe ser capaz de aprovisionar de manera básica los DT, dando la opción de llevar a cabo un aprovisionamiento profundo tras la fase de despliegue
R1.d	La plataforma debe de ser capaz de desplegar una réplica digital de la red objetivo completamente virtualizada y computacionalmente más simplificada
R1.e	La plataforma debe permitir la comunicación bidireccional entre el entorno del gemelo físico y el entorno del gemelo digital
R1.f	La plataforma debe permitir la obtención de datos sobre el estado de ambos gemelos, así como del tráfico cursado en cada entorno

- **Requisitos relacionados con la aplicabilidad a redes de comunicación**

La **Tabla 13** muestra los requisitos relacionados con la aplicabilidad del paradigma de gemelos digitales al caso específico de redes de comunicaciones.

Tabla 13. Requisitos relacionados a la aplicabilidad

Id.	Descripción
R2.a	La plataforma debe hacer uso de tecnologías habilitadoras capaces de desplegar una red física de manera virtualizada
R2.b	La plataforma debe permitir la obtención de la información mínima necesaria para llevar a cabo el modelado, independientemente de la topología presente en la red física, ya sea topología real o lógica
R2.c	La plataforma debe permitir la replicabilidad de los NDT a través de los descriptores NFV asociados a cada escenario.
R2.d	La plataforma debe cubrir la gestión de los NDT, permitiendo el despliegue, así como la eliminación de los escenarios.
R2.e	La plataforma debe ser independiente al tipo de red o caso de estudio a desplegar en forma de NDT

- **Requisitos relacionados con los flujos de comunicación**

La **Tabla 14** muestra los requisitos ligados a los flujos de comunicación presentes en el ecosistema completo. Cubriendo desde aspectos como la definición de los diversos canales, así como el almacenamiento y control de la información intercambiada sobre el canal bidireccional.

Tabla 14. Requisitos relacionados con los flujos de comunicación

Id.	Descripción
R3.a	La plataforma debe ser capaz de configurar de manera automática los distintos flujos de comunicación necesarios para garantizar el correcto intercambio de información entre el gemelo físico y el gemelo digital.
R3.b	La plataforma debe permitir la diferenciación de flujos de comunicación en base a las necesidades de tiempo real y/o tamaño de los datos intercambiados
R3.c	La plataforma debe ser capaz de implementar un subsistema encargado de controlar el intercambio de comunicación, evitando en todo lo posible la comunicación directa sin supervisión entre gemelos, con el fin de evitar la propagación de problemas o ataques de un entorno a otro

- **Requisitos relacionados con la seguridad**

La **Tabla 15** hace referencia a la seguridad necesaria en el canal de comunicación que interconecta el gemelo físico y el gemelo digital. Tratando de garantizar la confidencialidad, integridad y autenticidad de los datos.

Tabla 15. Requisitos relacionados con la seguridad

Id.	Descripción
R4.a	El canal utilizado para la comunicación bidireccional entre gemelos debe estar correctamente securizado, con el fin de garantizar la confidencialidad e integridad de la información intercambiada
R4.b	Debido a que la aplicabilidad del paradigma DT está conectando un sistema real, típicamente en producción, con un entorno virtual, es necesario establecer mecanismos que permitan analizar la información antes de ser propagada hacia cada uno de los entornos de gemelo
R4.c	Los mecanismos de seguridad deben adaptarse a los estándares existentes en la medida de lo posible, permitiendo si fuera posible, establecer mecanismos experimentales capaces de ofrecer una capa superior de seguridad frente a las soluciones tradicionales

- **Otros requisitos de la plataforma**

La **Tabla 16** muestra una serie de requisitos adicionales a los presentados anteriormente, centrándose en aspectos puramente funcionales de la solución.

Tabla 16. Otros requisitos

Id.	Descripción
R5.a	La plataforma debe poder ser operada de manera semi automática. Es decir, el grueso de la operación debe estar automatizado, dejando pequeños detalles ajustables de manera manual
R5.b	La plataforma debe diseñarse de manera modular, permitiendo la integración de nuevos módulos que enriquezcan el funcionamiento del ecosistema NDT, como pudiera ser la inclusión de sistemas basados en IA o ML para el procesamiento de la información
R5.c	La plataforma debe ofrecer la posibilidad de interoperabilidad, tanto para la obtención de nueva información, como para compartir información recolectada en el ecosistema NDT con otros sistemas

4.1.2. Diseño de la arquitectura

Los requisitos planteados previamente, junto con la necesidad de definir una plataforma genérica, extensible e interoperable en la medida de lo posible capaz de llevar a cabo la implementación de la metodología propuesta, llevan a conclusión de que el enfoque idóneo en cuanto a diseño de la arquitectura es la creación de un marco de referencia que permita estandarizar, en la medida de lo posible, el diseño aplicado al caso de uso específico de gemelos digitales de red.

Por ello, tras evaluar las distintas posibilidades existentes en cuanto a tecnologías habilitadoras que permitieran llevar esta tarea a cabo, se opta por realizar un diseño desde cero, el cual hace uso de ciertas tecnologías de referencia en el ámbito de las redes de nueva generación, concretamente en la tecnología NFV, la cual se perfila como un habilitador necesario para llevar a cabo la implementación de gemelos digitales de red.

A continuación, se pasará a desglosar los tres grandes bloques de la arquitectura diseñada que conforman el ecosistema NDT encargados de cubrir todo lo relacionado con el entorno de gemelo físico, de gemelo digital, así como el bloque encargado de la gestión de ambos entornos.

4.1.2.1. Bloque I. Entorno Gemelo Físico

Dentro de la diferenciación por bloques realizada dentro del ecosistema NDT controlado por la plataforma, el entorno del gemelo físico, o red objetivo a modelar en forma de gemelo digital, es el elemento a través del cual comienza todo el proceso de diseño y despliegue del NDT.

Tal y como se presentó en el apartado **3.1.1**, en este bloque se despliega el denominado agente inteligente encargado de la recolección de datos sobre la infraestructura de red real para su posterior modelado, adaptación y despliegue en forma de gemelo digital de red. Este agente está diseñado de manera modular, como se observa en la **Figura 34**, lo cual permite su evolución o inclusión de nuevas funcionalidades. A continuación, se revisarán los distintos subsistemas que conforman el agente inteligente desplegado sobre el entorno de gemelo físico.

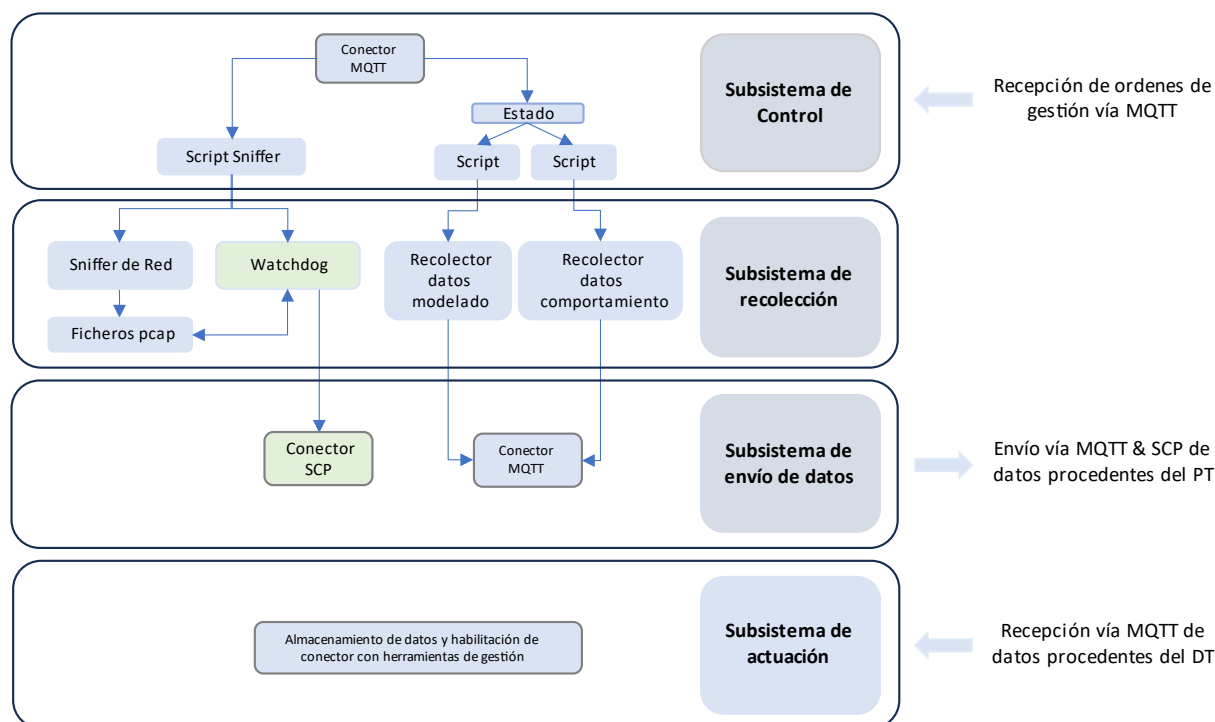


Figura 34. Diagrama bloque I de la plataforma de soporte al NDT

- **Subsistema de control.** Es el subsistema encargado de controlar el funcionamiento global del agente inteligente, pudiendo controlar el modo de recolección de información en base al estado en el que se encuentre el ecosistema NDT, diferenciando entre la fase inicial, en la cual no existe aún la réplica digital en forma de DT, de la fase de realimentación, la cual habilita la recolección de datos de comportamiento que permitan realizar una coevolución entre el gemelo físico y el gemelo digital.

Este subsistema se basa en el procesamiento de determinados flags booleanos enviados desde el bloque II, sistema de gestión NDT, el cual es un elemento centralizado de la arquitectura que permite al operador de la plataforma indicar el modo de funcionamiento que debe activar el agente, permitiendo, por ejemplo, poder auto adaptarse ante un cambio sustancial en la topología de red física. Estos flags activan automáticamente la ejecución de los scripts específicos encargados de iniciar la tarea de recolección de datos.

Adicionalmente a esas dos tipologías de datos, el agente inteligente controla de igual manera un script encargado de activar la tarea de sniffing de red, con el objetivo de obtener datasets de tráfico capturados en la red real. Este sniffer de red

almacena los ficheros pcap generados en el sistema de ficheros del subsistema de recolección, el cual está continuamente monitorizado por un submódulo denominado *watchdog*, el cual está encargado de detectar la generación de nuevos pcaps, para llevar a cabo su envío vía SCP hacia el sistema de gestión donde serán almacenados.

- **Subsistema de recolección.** Se trata del subsistema encargado de recolectar la información establecida en cada uno de los dos modos de funcionamiento contemplados, diferenciados en base a estados, siendo el estado cero el encargado de recolectar información topológica y el estado uno el encargado de recolectar información referente al comportamiento de los diversos elementos de red desplegados.

Tal y como se introduce en la sección **3.1.1**, este subsistema realiza la tarea de sondeo o recepción de reportes sobre los diferentes elementos de red, diferenciando claramente entre dos tipos:

- Elementos de red de propósito específico.
- Elementos de red de propósito general.

Esta diferenciación permite cubrir un amplio espectro de redes de comunicación, las cuales pueden estar basadas completamente en hardware específico como pueden ser routers, concentradores, etc. Los cuales típicamente no permiten la instalación de software adicional en ellos, y únicamente se habilita la recolección de información o estado a través del protocolo SNMP, o basadas en hardware genérico el cual puede albergar funciones o elementos de red basados en virtualización. En este segundo caso, no es necesario limitarse a la información obtenida vía SNMP, sino que es posible la instalación de agentes satélite capaces de recolectar información mucho más específica, capaz de reflejar características físicas y lógicas de estos equipos, lo cual facilita las tareas de modelado y despliegue de este tipo de redes.

Esta aproximación realizada en el subsistema de recolección permite cubrir la mayoría de los escenarios de red actuales, los cuales suelen optar por una topología híbrida, en la cual coexisten elementos de red tradicionales con elementos de red

de última generación basados en virtualización ligera, como pueden ser contenedores o virtualización tradicional como es la basada en máquinas virtuales.

- **Subsistema de envío de datos.** Este subsistema es el encargado de establecer la comunicación entre el entorno del gemelo físico y el sistema de gestión NDT. Por ello, solo es necesario que el equipo alberga el agente inteligente sea el único que tenga conectividad con el equipo que alberga el sistema de gestión NDT, concentrando en él todos los flujos de comunicación que habilitará el sistema de gestión NDT.

Esta decisión de diseño simplifica la configuración y securización de la comunicación, teniendo únicamente dos canales para el intercambio de información, tanto de la generada entre los gemelos, como la información de control que actúa directamente sobre el agente inteligente.

Estos dos canales de comunicación están diferenciados principalmente por el protocolo a utilizar, cuya decisión surge principalmente en base a dos características asociadas a los datos: el tamaño de los mensajes intercambiados y la necesidad de comunicación en tiempo real. Referente a las características, se opta por la utilización del protocolo MQTT, el cual permite la segmentación del canal en flujos de información asignados a topics específicos, los cuales son utilizados para el envío de información topológica, así como el intercambio de información de comportamiento para cada elemento modelado como DT. En este caso, los datos no requieren comunicación en tiempo real, ya que el agente inteligente se encarga de recopilar la información de manera periódica y adaptarla según modelo de datos en formato JSON definido en el apartado 3.1.1.3. el cual no excede en el tamaño máximo para el *payload* soportado por MQTT, 256 Megabytes. No obstante, el agente inteligente permite la recolección de otro tipo de datos, los cuales difieren sustancialmente de los datos enviados a través de MQTT. Estos datos son capturas de tráfico, en formato pcap, los cuales son utilizados tanto para la generación de datasets con el fin de ser utilizados por modelos de detección de anomalías basadas en IA o ML, así como para la reinyección en el gemelo digital del tráfico capturado en la red real. Debido principalmente al tamaño de estos datos, se opta por utilizar como segundo canal de comunicación basado en SCP, el cual sigue una filosofía similar a MQTT, habilitando un módulo TX (transmisor) y RX

(receptor), concentran los flujos de información en un sistema de ficheros, el cual se despliega en el bloque II de la plataforma.

- **Subsistema de actuación.** Debido a que la operación sobre una red real típicamente se realiza a través de herramientas específicas, este subsistema únicamente implementa la recepción de la información, habilitando la opción de interconectar con esos sistemas, los cuales están fuera del alcance del diseño del ecosistema NDT debido a que no existe una solución única y generalmente recae en el uso de herramientas propietarias ligadas al equipamiento hardware desplegado en la red.

4.1.2.2. Bloque II. Sistema de gestión NDT

En este bloque conforma el elemento central de la plataforma del ecosistema NDT, permitiendo interconectar entre el entorno físico y el entorno digital. Se trata de un sistema encargado de albergar toda la lógica de gestión, operación y comunicación asociada al ecosistema NDT, conformado por diferentes subsistemas desarrollados de manera modular, garantizando la correcta interconexión entre todos ellos y obteniendo una arquitectura escalable capaz de albergar nuevas funcionalidades en función de la demanda o evolución futura de la plataforma.

A continuación, se pasará a desglosar cada uno de los subsistemas, reflejados en la **Figura 35**, los cuales conforman el sistema de gestión o gestor de manera completa.

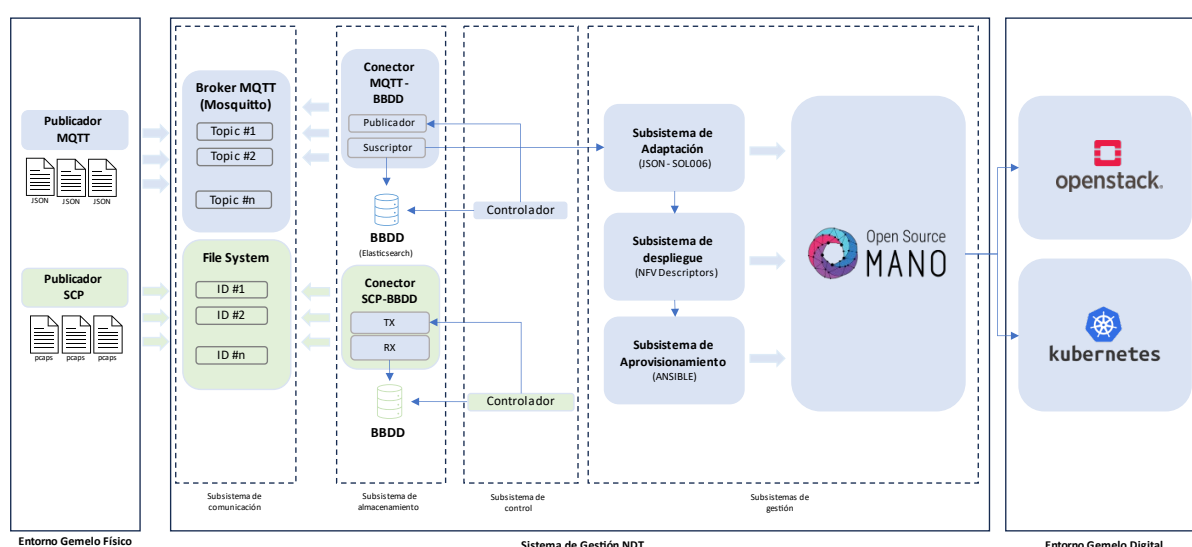


Figura 35. Plataforma para el soporte del NDT

- **Subsistema de comunicación.** Tal y como se ha expuesto en el capítulo 0, en referencia al paradigma de gemelos digitales, una de las características más relevantes es la comunicación bidireccional entre el gemelo físico y su contraparte digital. Por ello, es necesario abordar este requerimiento a la hora de diseñar y desarrollar el sistema de gestión NDT. De igual manera, este aspecto ha sido introducido, tanto en bloque anterior, como en el apartado 3.1.1.3, tomando el entorno físico como entorno inicial de referencia para presentar el sistema de comunicación diseñado, teniendo su parte dual en el entorno digital.

Esta comunicación, se basa en el protocolo de mensajería, MQTT, siguiendo un patrón publicador-suscriptor y permitiendo la diferenciación de los flujos de información. Ambos hacen uso de clientes que pueden funcionar de manera simultánea tanto como publicadores para enviar información desde los entornos físico o digital hacia el sistema de gestión, así como suscriptores para recibir información de gestión o información de su gemelo correspondiente en la fase de realimentación.

En el subsistema de comunicación del sistema de gestión se albergan los componentes denominados “brokers” encargados de gestionar los flujos en base a topics o etiquetas, aplicando adicionalmente configuraciones referentes a aspectos como políticas de reenvío, persistencia de mensajes o seguridad en las comunicaciones. Estas configuraciones son establecidas a través de los correspondientes ficheros de configuración.

Adicionalmente al funcionamiento general de ambos protocolos, el subsistema de comunicación se ha diseñado con un doble objetivo. En primer lugar, permitir al sistema de gestión almacenar toda la información generada y enviada desde cualquiera de los entornos de gemelos, y en segundo, forzar a que el sistema de gestión se encargue de dar paso a la información que fluye de un entorno al otro evitando una comunicación directa sin supervisión. Esta característica permite cortar el flujo de información en caso de detectar alguna anomalía o mal comportamiento de los datos, lo cual podría repercutir negativamente en los entornos en caso de que se permitiera una comunicación directa totalmente desatendida. La **Tabla 17** muestra los cuatro grandes flujos presentes en el sistema de gestión NDT.

Tabla 17. Flujos de comunicación presentes en el sistema de gestión NDT

Id.	Descripción
PT2MANO	Physical Twin to MANO. Habilita la comunicación desde el agente inteligente desplegado en el entorno físico hacia el sistema de gestión MANO. El sistema de gestión se encarga de recoger toda la información enviada desde el entorno físico, almacenarla en base de datos y dar paso a su reenvío hacia el gemelo digital.
MANO2DT	MANO to Digital Twin. Habilita la comunicación desde el sistema de gestión MANO hacia el agente inteligente desplegado en el entorno virtual. Tras guardar la información recibida del entorno físico en base de datos, queda a la espera de que el subsistema de publicación / controlador, de paso a reenviar la información al entorno virtual.
DT2MANO	Digital Twin to MANO. Habilita la comunicación desde el entorno virtual hacia el sistema de gestión MANO. El sistema de gestión se encarga de recoger toda la información enviada desde el entorno digital, almacenarla en base de datos y dar paso a su reenvío hacia el gemelo físico.
MANO2PT	MANO to Physical Twin. Habilita la comunicación desde sistema de gestión MANO hacia el entorno físico. Tras guardar la información recibida del entorno digital en base de datos, queda a la espera de que el subsistema de publicación / controlador, de paso a reenviar la información al entorno físico.

En lo referente a la securización de las comunicaciones entre los diferentes entornos físico – virtual y el sistema de gestión, es este último el que se encarga de la generación de los certificados necesarios para garantizar la privacidad de los datos, a través del cifrado de la información, la integridad ante cualquier tipo de alteración de la información, así como la autenticación entre cliente y servidor.

- **Subsistema de almacenamiento.** En este caso, este subsistema es el encargado de recibir los flujos de información PT2MANO y DT2MANO indicados en la **Tabla 17**, para proceder al almacenado de los datos en base de datos, ya provengan a través de MQTT o SCP. Para ello, se hace uso de una implementación basada en el cliente *python MQTT paho* [164], el cual se encarga de suscribirse a los topics declarados en el broker en la fase de despliegue del ecosistema NDT, y realizar su inserción en formato JSON en base de datos *Elasticsearch* [165], encargándose de agregar el atributo que hace referencia al sentido de la comunicación, PT2MANO o DT2MANO, permitiendo así la posibilidad de realizar consultas a la base de datos de manera más precisa.

Debido a la securización realizada en los brokers de mensajería, los clientes MQTT, requieren de sendos ficheros de configuración en los cuales se indique la ubicación de los certificados necesarios para poder autenticarse contra los brokers.

- **Subsistema de publicación (Controlador).** Una vez realizado el almacenamiento de la información en base de datos o sistema de ficheros por parte del subsistema de almacenamiento a través del conector MQTT/BBDD es el controlador o sistema de publicación el encargado de permitir que la información vaya desde el sistema de gestión hacia sendos entornos de gemelo, cubriendo la comunicación MANO2DT y MANO2PT.

En este caso, se trata de un subsistema atendido, el cual, como se ha comentado anteriormente, evita de manera intencional que la comunicación fluya de manera automática sin posibilidad de realizar una supervisión, manual o automatizada de los datos.

Una vez se da paso a la información, el subsistema hace uso de los mismos clientes desarrollados para el subsistema de almacenamiento, pero esta vez en modo publicador, realizando en primer lugar una acción de consulta, bien una consulta a base de datos para recuperar los datos ingresados o bien una suscripción a los topics correspondientes, esta decisión está ligada a la política de persistencia de los datos asignada a los topics. Una vez tomados los datos por parte del controlador, este hace uso de la funcionalidad de publicación para reinyectar la información en los topics asociados a los flujos MANO2DT y MANO2PT.

Al igual que en el caso del subsistema de almacenamiento, los clientes MQTT, esta vez en modo publicador, hacen uso de los mismos ficheros de configuración establecidos para el subsistema de almacenamiento, dado que se ejecutan en el mismo sistema global, el cual tiene almacenados los certificados requeridos para la autenticación con los brokers. Esto es debido a que, pese a que tanto los brokers como los agentes del subsistema de almacenamiento y publicación se encuentran en la misma máquina, y realizan una comunicación a través de localhost, una vez securizados los brokers, requieren de los certificados para poder realizar la autenticación que permite la conexión, ya sea en modo suscriptor o en modo publicador.

- **Subsistema de adaptación.** Este subsistema es el encargado de recibir el fichero de topología enviado por el agente inteligente desplegado en el entorno físico, el cual sigue el modelo JSON descrito en la sección 3.1.1.3, y llevar a cabo la adaptación al formato definido en la especificación NFV-SOL006 para obtener los descriptores de servicio y de función de red, NSD y VNFD respectivamente, encargados de caracterizar el gemelo físico en forma de VNFs las cuales son desplegadas sobre el entorno digital.

Este subsistema está permanentemente suscrito al topic MQTT encargado de recibir la información topológica de la red física. Debido a que la información topológica es reportada de forma periódica por parte del agente, este subsistema compara la topología desplegada en forma de gemelo digital con las topologías actualizadas, de forma que ante cualquier cambio sustancial en la red real, desencadenaría una nueva readaptación del gemelo digital generando nuevos ficheros NSD y VNFD, permitiendo así al sistema de gestión decidir si desplegar el nuevo gemelo manteniendo el previo, o por el contrario, eliminar o modificar el gemelo digital existente para llevar a cabo la actualización topológica en él.

- **Subsistema de despliegue.** Tras la caracterización del gemelo físico en base a la especificación NFV-SOL006 realizada por el subsistema de adaptación se obtienen los ficheros NSD y VNFD correspondientes al despliegue. Estos ficheros son almacenados en el directorio de ficheros manejado por el subsistema de despliegue permitiendo ordenarlos a través de un identificador basado en la fecha de creación de dichos ficheros.

Este subsistema se encarga de revisar la sintaxis de los descriptores NFV a través de la herramienta OSM. Una vez verificados, el subsistema se encarga de generar los paquetes en formato “.tar.gz”, cargarlos al repositorio interno de Open Source MANO y lanzar las tareas de despliegue del gemelo digital en forma de NS. Una vez se lanza la tarea de despliegue, se llevan a cabo las siguientes tareas específicas por parte de OSM:

- **Validación de la solicitud de despliegue.** Tras la ejecución del despliegue, OSM realiza una serie de verificaciones sobre la disponibilidad de los recursos necesarios establecidos en los ficheros VNFD, centrándose en la capacidad de procesamiento, almacenamiento y conectividad de red.

- **Orquestación de recursos.** Una vez aprobada la solicitud de despliegue, la herramienta OSM se encarga de orquestar los recursos virtuales necesarios, dependiendo de la definición de los ficheros VNFD, estos recursos pueden ser máquinas virtuales o contenedores. Esta orquestación de recursos se realiza sobre los diferentes VIM habilitados en el sistema OSM.
 - **Conexionado de componentes.** De manera automática, OSM se encarga de implementar el conexionado lógico entre los distintos componentes declarados en el NSD, estableciendo la comunicación necesaria entre ellos. Esta tarea implica la configuración de interfaces, rutas de red o políticas de seguridad entre las distintas VNFs que conforman el gemelo digital de red a desplegar.
 - **Configuración y aprovisionamiento.** Tras la fase de instanciación, citada anteriormente como fase Day-0, se realizan una serie de configuraciones específicas para cada componente del NS en función de las especificaciones reflejadas en sus ficheros VNFD asociados, incluyendo tareas como la instalación de software, la configuración de parámetros de red y la asignación de recursos de CPU, memoria RAM y almacenamiento de disco.
 - **Verificación y monitorización.** Una vez desplegado el NS, se realizan una serie de pruebas automatizadas por parte de OSM para verificar el correcto funcionamiento del escenario desplegado. Adicionalmente, se implementan mecanismos de monitorización continua para centralizar métricas de rendimiento, disponibilidad o integridad del NS desplegado, permitiendo la identificación de posibles problemas o mal funcionamiento.
- **Subsistema de gestión (OSM).** Este subsistema, utiliza el software OSM como elemento principal. Tal y como se ha visto anteriormente, tanto el subsistema de gestión como el subsistema de despliegue mantienen una fuerte dependencia entre ambos, debido al uso común que realizan sobre herramienta OSM.

En el caso del subsistema de despliegue, se cubren las fases o primitivas identificadas como Day-0 y Day-1. No obstante, en la fase de gestión realizada desde OSM se abordan las primitivas de tipo Day-2. Estas configuraciones son implementaciones definidas en el descriptor VNF y ejecutables bajo demanda. Tras la finalización del despliegue, estas acciones de tipo Day-2 se habilitan a través de la interfaz de control

de OSM, permitiendo la ejecución de tareas concretas como es la captura de tráfico, la modificación de funcionamiento en las VNFs o la actualización de parámetros específicos.

Adicionalmente a las acciones de tipo Day-2, el subsistema de gestión permite realizar una gestión completa del ciclo de vida del gemelo digital desplegado en forma de NS, a través de la interfaz de control de OSM. Permitiendo obtener en todo momento el estado del despliegue, así como el estado de los recursos asociados a los VIM disponibles. Permitiendo además replicar o escalar el despliegue del gemelo digital, o en caso de que el subsistema de adaptación detectara cambios sustanciales en el entorno físico, a través del subsistema de gestión es posible eliminar o modificar el NS desplegado para alcanzar el nuevo estado.

- **Subsistema de infraestructura Virtualizada (VIM).** Dentro de la plataforma diseñada, tomando como base la arquitectura de referencia NFV MANO, y específicamente el framework OSM, es necesario el uso y configuración del subsistema VIM encargado de la gestión de los recursos de computación, almacenamiento y red, para poder llevar a cabo los despliegues NFV. En la plataforma de soporte a la NDT se maneja el uso de dos tipos diferenciados de VIMs.

En primer lugar, para dar soporte a la orquestación de contenedores, OSM permite VIM basados en clúster de Kubernetes, conocido de forma abreviada como K8s, utilizando en este caso concreto el concepto de *Kubernetes Network Function* (KNF) y *Kubernetes Deployment Unit* (KDU) en vez de VNF y VDU. Kubernetes se define como una plataforma portable y extensible de código abierto para administrar cargas de trabajo y servicios. A diferencia de otros orquestadores, K8s no opera a nivel de hardware, si no de contenedor. Los contenedores se basan en virtualización a nivel de sistema operativo independiente del sistema anfitrión, lo que hace que sean pequeños y rápidos y más fáciles de administrar que una máquina virtual.

En segundo lugar, se hace uso de VIM basado en OpenStack, capaz de controlar grandes grupos de recursos de computación, almacenamiento y red distribuidos en un centro de datos, todos ellos gestionados y aprovisionados a través de APIs, enfocado principalmente a la virtualización en forma de máquinas virtuales.

- **Subsistema de aprovisionamiento.** Por último, dentro del sistema global de gestión, se encuentra el subsistema de aprovisionamiento, el cual está pensado para ser un subsistema de apoyo en caso de necesitar realizar tareas de configuración complejas, las cuales no puedan ser abordadas por las configuraciones básicas realizadas con las acciones de tipo Day-1.

En este caso, se trata de un subsistema basado en ANSIBLE, el cual está conectado a la red MGMT presente en todos los despliegues realizados por la plataforma, permitiendo la ejecución de playbooks específicos que permiten alcanzar un estado específico de configuración en el gemelo digital.

4.1.2.3. Bloque III. Entorno Gemelo Digital

Al igual que el bloque I desplegado en el entorno del gemelo físico, el bloque III está conformado por una serie de subsistemas desplegados en el entorno del gemelo digital capaces de realizar acciones como la recolección de datos o la actuación sobre elementos individuales del gemelo digital de red desplegado sobre la infraestructura virtualizada.

Tal y como se presentó en apartado **3.1.1**, en este bloque se despliega el denominado agente inteligente encargado de la recolección de datos, con la salvedad de que esta vez se despliega sobre la infraestructura de red digital. Este agente está diseñado de manera modular, como se refleja en la **Figura 36**, lo cual permite su evolución o inclusión de nuevas funcionalidades. A continuación, se revisarán los distintos subsistemas que conforman el agente inteligente desplegado sobre el entorno de gemelo digital.

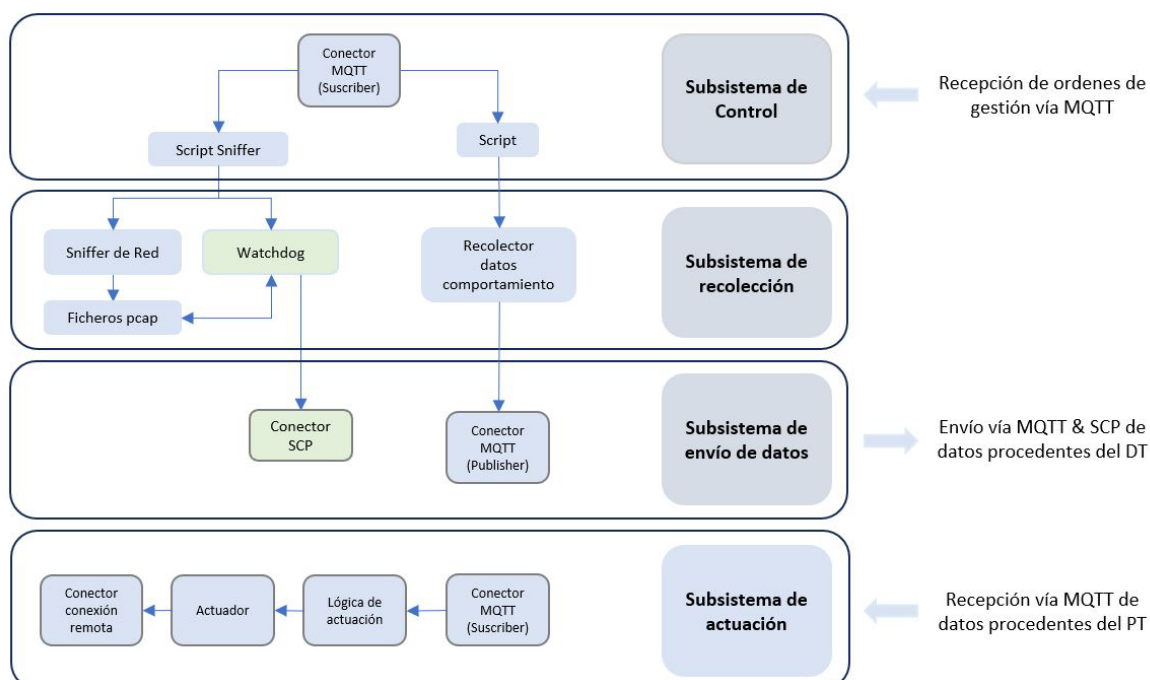


Figura 36. Diagrama bloque III de la plataforma de soporte al NDT

- **Subsistema de control.** Al igual que en el bloque I, este subsistema encargado de controlar el funcionamiento global del agente inteligente. A diferencia del agente desplegado en el entorno físico, en el caso del agente inteligente desplegado en el gemelo digital, no se realiza una captura de la topología de red, dado que la plataforma NDT no contempla la opción que la topología del gemelo digital pueda evolucionar y actuar sobre la replanificación física de la red real. No obstante, si se habilita un canal de gestión vía MQTT, a través del cual se une el agente inteligente con el sistema de gestión.

Al igual que su contraparte física, este subsistema se basa en el procesado de determinados flags booleanos enviados desde el bloque II, sistema de gestión NDT, permitiendo, por ejemplo, utilizar el gemelo digital como generador de *datasets* de tráfico necesarios para el entrenamiento de sistemas expertos capaces de detectar y predecir posibles anomalías o amenazas de seguridad antes de que se materialicen en el entorno real.

Otra funcionalidad inherente a este subsistema es el permitir la reinyección de *datasets* de tráfico real capturados en el entorno físico los cuales están almacenados en el subsistema de almacenamiento descrito en el bloque II y pueden ser reinyectados hacia el gemelo digital mediante el módulo TX del cliente SCP.

- **Subsistema de actuación.** Una vez el ecosistema NDT ha sido desplegado e interconectado a través del protocolo MQTT, los agentes inteligentes pasan al estado uno, o estado de monitorización. En el caso del gemelo físico, se encarga de obtener información de comportamiento, la cual se propaga al sistema de gestión el cual es el encargado tanto de almacenarla como de dar paso al reenvío hacia el gemelo digital. Dada la diferenciación en flujos de información definidos en el broker MQTT, es posible realizar una relación P2P entre un gemelo físico y su contraparte digital. En este punto es donde el subsistema de actuación entra en juego, suscribiéndose automáticamente a los topics relativos a la red modelada en forma de gemelo digital, y encargándose de actuar en base al tipo de información sobre el elemento concreto de la NDT que corresponda, permitiendo así que el gemelo digital sea capaz de evolucionar a través de los datos recopilados de su contraparte física.

Es necesario remarcar un detalle relevante que diferencia los entornos definidos en los bloques I y III. A diferencia de los elementos presentes en el entorno físico, los cuales, tal y como se ha explicado anteriormente, pueden ser equipos de propósito general o específico, en el entorno digital pasan a ser funciones de red o equipos virtualizados, los cuales, desde una perspectiva de actuación y monitorización, se categorizan como equipos de propósito general. En este punto, se da importante diferenciación a la hora de modificar e interactuar con los elementos, permitiendo en el caso del entorno digital, realizar cualquier tipo de modificación o reconfiguración de manera mucho más sencilla que en el caso del entorno físico. Por ejemplo, si se detecta un cambio en la configuración de un equipo físico, el cual únicamente es operable vía SNMP y accesible a través de software propietario, el cambio en un elemento virtual modelado que realice sus mismas operaciones a nivel de red tendría una complejidad menor ya que el subsistema de actuación sería capaz, a través de una conexión remota simple como puede ser la conexión SSH, de acceder al equipo y realizar la operación necesaria. En el caso contrario, involucraría sistemas de gestión que están fuera del alcance de la arquitectura de soporte a la NDT propuesta, ya que sería necesario englobar un amplio abanico de posibilidades en función de la operación y gestión que se realice sobre esa red.

Este subsistema a su vez está formado por tres componentes principales:

- **Lógica de actuación.** En función la información, obtenida a través de los topics MQTT encargados de interconectar los gemelos, la lógica de actuación se encarga de procesar esa información y adaptar la acción a realizar por el módulo actuador sobre el elemento digital de forma que se adecue al comportamiento o estado reportado por el gemelo físico.
 - **Actuador.** Se encarga de gestionar las acciones generadas por la lógica de actuación, identificando a que elemento de la NDT va destinado el cambio o modificación.
 - **Conector conexión remota.** Módulo encargado de realizar la conexión remota desde el agente inteligente desplegado en la NDT, hacia el componente específico que requiera la acción. Debido a que el sistema de gestión tiene pleno conocimiento de las características del NDT, en referencia al direccionamiento IP, usuarios, contraseñas, etc. Este submódulo se encarga de procesar esa información para realizar la conexión remota que permita llevar a cabo la acción necesaria por parte del actuador.
- **Subsistema de recolección.** En el caso del entorno de gemelo digital, el subsistema de recolección difiere en el funcionamiento de su contraparte desplegada en el entorno de gemelo físico, haciéndolo operacionalmente más sencillo. Esto es debido, a que como se ha comentado en el subsistema de actuación, en el entorno digital donde se despliega el NDT no se contempla la existencia de equipamiento específico operable únicamente vía protocolos como SNMP. De esta manera, el subsistema de recolección presente en el entorno digital hace uso únicamente de la opción de recepción de información a través de agentes satélites desplegados por defecto en cada una de las VNF que conforman el NS. Al igual que el agente inteligente del entorno físico, este agente hace uso del modelo de datos definido en la metodología presentada en el capítulo tres. Con la diferencia de que los datos referentes a SNMP van vacíos en el sentido DT2MANO.
 - **Subsistema de envío de datos.** Al igual que lo indicado en el bloque I, este subsistema es el encargado de establecer la comunicación entre el entorno del gemelo digital y el sistema de gestión NDT. Por ello, solo es necesario que el equipo alberga el sistema DANA sea el único que deba tener conectividad con el sistema de gestión NDT.

Capítulo 5

Casos de estudio

En el capítulo anterior se muestra la plataforma encargada de llevar a cabo la implementación técnica de la contribución principal de esta tesis basada en la metodología propuesta en el capítulo tres. Dicha plataforma permite abordar los retos mencionados en la literatura enfocados específicamente a los gemelos digitales aplicados a redes de comunicación. Concretamente se aborda las tareas de recolección de información necesaria para poder desplegar el gemelo digital de red, tanto a nivel topológico, como a nivel comportamiento, estandarizando la caracterización y el despliegue de este en forma de funciones de red virtualizadas, las cuales son desplegadas a través del framework OpenSource MANO, así como los mecanismos necesarios para poder establecer un canal de comunicación entre el mundo físico y el mundo digital, permitiendo gestionar todo ello desde un elemento centralizado encargado de orquestar todos los componentes de gestión.

A lo largo del capítulo anterior, se presentan los diferentes subsistemas y módulos diseñados para poder llevar a cabo el correcto despliegue del NDT cumpliendo los requisitos de este paradigma, así como la respuesta técnica para tratar de solventar los distintos retos relacionados al caso específico de las redes de comunicación.

En este capítulo se presentan tres casos de estudio (revisar si cambiar o no de nombre) relacionados al ecosistema NDT, con los cuales se pretende ilustrar el correcto funcionamiento tanto de la plataforma implementada, como de la metodología propuesta, así como la identificación de posibles limitaciones. Cada caso de estudio (revisar si cambiar o no de nombre) hace énfasis en determinadas fases de la metodología, y en los

subsistemas de la plataforma asociados. En primer lugar, el caso I, se centra en la capacidad de recolección de la información topológica del agente inteligente desplegado en la red física, su posterior adaptación a la especificación NFV-SOL006 y su despliegue sobre el VIM OpenStack a través de OSM. En segundo lugar, el caso II, se centra en la capacidad de modelado de un escenario 5G, en el cual está presente tanto el propio core 5G como el conector STGUTG encargado de interconectar el terminal de usuario con la red virtual, en este caso se aborda tanto el modelado de la red en forma de ficheros VNFD y NSD, así como las acciones de tipo Day-2 que permiten la obtención de datasets de tráfico a través de la herramienta ntop-ng instalada en el NDT. Por último, el caso III, se centra en el canal de comunicaciones basado en MQTT el cual interconecta los distintos gemelos del ecosistema, en él se plantean un análisis desde el punto de vista de la seguridad del canal, utilizando para ello dos modos de securización, la primera basada en TLS/SSL tradicional y la segunda basada en TLS/SSL con soporte a criptografía postcuántica.

5.1. Caso I. Despliegue de NDT a través de red generada en GNS3

Este primer caso de estudio tiene como objetivo validar el correcto funcionamiento del agente inteligente desplegado en el entorno físico, el cual realiza la tarea inicial, correspondiente a la fase I de la metodología, sección 3.1.1, encargada de recolección de información topológica mínima necesaria para generar el NDT.

Para ello se parte de una arquitectura virtualizada a través del software GNS3, representada en la **Figura 37**, que emula una red de agregación simple, la cual interconecta tres UE mediante el uso del conector STG/UTG con el core de red 5G. Dicha red está formada por instancias router Cisco configuradas de forma que permiten la conectividad completa en toda la red.

Una vez desplegada la red que actúa como gemelo físico, se despliega, conectado a uno de los routers, el agente inteligente encargado de la recolección de información, el cual contiene todos los subsistemas descritos en la sección 4.1.2.1. Dado que el sistema se encuentra en un estado inicial en el cual no existe aún el NDT, el bloque II de gestión envía el flag de estado al agente a través del canal correspondiente vía MQTT. El flag es procesado por el subsistema de control presente en el agente el cual se encarga de su procesamiento y automáticamente activa el subsistema de recolección encargado de la recolección de información topológica.

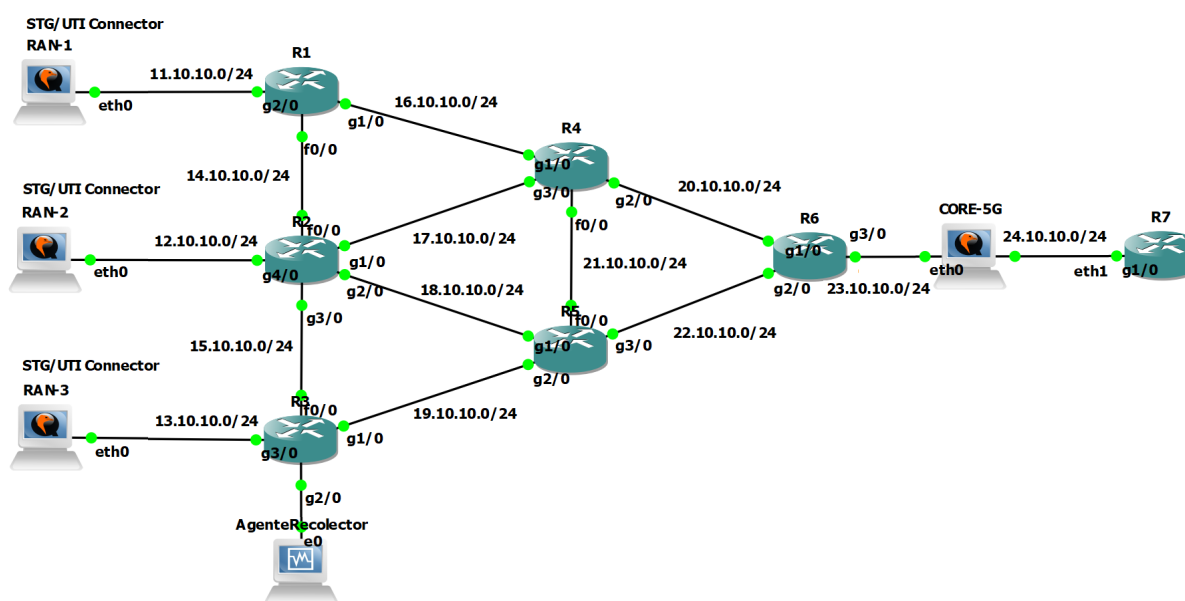


Figura 37. Caso I - Red gemelo físico

Debido a la naturaleza de los equipos desplegados en este caso de estudio, los cuales emulan equipamiento hardware de propósito específico, el agente se encarga de activar el modo de funcionamiento correspondiente a este tipo de equipos, el cual está basado en el protocolo SNMP para la obtención de información. En este momento el agente actúa como elemento centralizado, realizando consultas SNMP en referencia a los OIDs presentes en la **Tabla 18**.

Tabla 18. Listado de Object Identifier utilizados por el agente

OID	Nombre
.1.3.6.1.6.3.10.2.1.1.0	snmpEngineID
.1.3.6.1.2.1.1.5.0	sysName
.1.3.6.1.2.1.1.1.0	sysDescr
.1.3.6.1.2.1.1.2.0	sysObjectID
.1.3.6.1.2.1.1.3.0	sysUpTime
.1.3.6.1.2.1.4.24.4	ipCidrRouteTable
.1.3.6.1.2.1.4.24.4.1.1	ipCidrRouteDest
.1.3.6.1.2.1.4.24.4.1.4	ipCidrRouteNextHop
.1.3.6.1.2.1.4.24.4.1.5	ipCidrRouteifIndex
.1.3.6.1.2.1.4.24.4.1.6	ipCidrRouteType
.1.3.6.1.2.1.4.24.4.1.16	ipCidrRouteStatus
.1.3.6.1.2.1.4.20	ipAddrTable
.1.3.6.1.2.1.4.20.1.1	oidIfacesIP
.1.3.6.1.2.1.4.24.3.0	ipCidrRouterNumber
.1.3.6.1.2.1.2.2.1.2	ifDescr
.1.3.6.1.2.1.2.2.1.4	ifMtu
.1.3.6.1.2.1.2.2.1.5	ifSpeed
.1.3.6.1.2.1.2.2.1.6	ifPhysAddress

En base a las consultas realizadas por el agente inteligente, se pasa a realizar el proceso de modelado de la información para adecuarse al modelo de datos en formato presentado en la sección **3.1.1.3** encargado de obtener la representación semántica de la red. Este modelo genera un fichero en formato JSON el cual contiene un identificador global asociado a la red física que se está modelando en forma de NDT, así como una serie de objetos JSON para cada uno de los elementos de red presentes en dicha red.

Debido al estado del agente, controlado a través del flag enviado desde el bloque de gestión, el modelo de datos únicamente refleja los datos asociados al campo “*snmp_topology_data*”, presentado en la **Tabla 9**, ya que como se ha comentado anteriormente, este caso de estudio se centra en la caracterización de los equipos hardware de propósito general sondeados a través del protocolo SNMP. Una vez la tarea de recolección de datos converge, habiendo obtenido la información necesaria para la representación de la topología, el agente inteligente pasa a activar el subsistema de envío de datos, el cual se encarga a través del conector MQTT de enviar el fichero JSON resultante al topic TOPOLOGY haciendo uso del flujo de comunicación PT2MANO, como se puede observar en la **Tabla 17**.

Llegado a este punto, tras la publicación del fichero JSON en el topic TOPOLOGY del broker MQTT alojado en el sistema de gestión (Bloque II), el agente detiene el subsistema de recolección de datos, quedándose suscrito al topic de gestión a la espera de recibir la siguiente orden a través del canal de gestión controlado por el Bloque II.

De manera dual, tal y como se establece en el apartado **4.1.2.2** con relación al Bloque II, el sistema de gestión implementa un cliente MQTT suscrito al topic “*topology*”, encargado de recibir el fichero JSON reportado por el agente inteligente desplegado en el entorno físico. Tras obtener el fichero JSON que modela la topología del gemelo físico, el sistema de gestión activa el subsistema de adaptación encargado de procesar el fichero JSON para la obtención de los *n* ficheros VNFD, así como el fichero NSD que describirá la implementación del NDT en el entorno digital.

El resultado de la tarea realizada por el subsistema de adaptación es la obtención y subida de los descriptores VNF y NS a la plataforma OpenSourceMANO. En la **Figura 38** y **Figura 39** se observa el resultado de esta iteración, tras haber llevado a cabo la adaptación a la especificación NFV-SOL006 y verificado la correcta sintaxis de los ficheros tar.gz generados.

Product Name	Identifier	Version	Provider	Type	Description	Actions
R1	cd4f2cb1-54c8-40b8-807c-7eba12d77e56	1.0	UPM_tesis	vnfd	Router R1 red agregacion	  Action ▾
R2	afcc9071-ee3b-4fb0-88a4-4616ebf56f28	1.0	UPM_tesis	vnfd	Router R2 red agregacion	  Action ▾
R3	1f207130-df19-4410-abc2-fae23661eaeef	1.0	UPM_tesis	vnfd	Router R3 red agregacion	  Action ▾
R4	77f0e6a3-43b7-4216-9cdf-753944321c72	1.0	UPM_tesis	vnfd	Router R4 red agregacion	  Action ▾
R5	029a428d-7497-4efb-8680-6aa0cf063f93	1.0	UPM_tesis	vnfd	Router R5 red agregacion	  Action ▾
R6	ac8c7b7b-e81f-44a8-b904-5eac30555b3	1.0	UPM_tesis	vnfd	Router R6 red agregacion	  Action ▾

Figura 38. Caso I - Ficheros VNFD cargados en OSM

Desde la interfaz web de gestión de OSM es posible verificar el estado de los ficheros generados, tanto de los VNFD correspondientes a cada router, como el fichero NSD encargado de definir todas las interconexiones de cada uno de los routers modelados como funciones de red virtuales. En este momento se completa la fase III de la metodología, dando paso a las fases IV y V, despliegue y aprovisionamiento respectivamente.

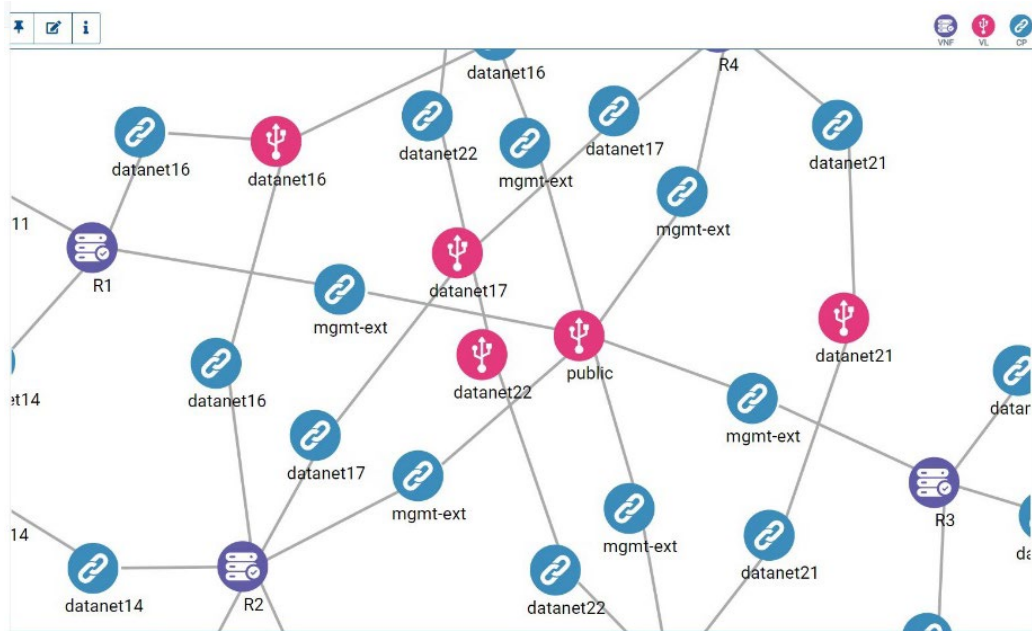


Figura 39. Caso I - Representación gráfica fichero NSD

Una vez terminada la tarea de adaptación a la especificación NFV-SOL006, por parte del subsistema de adaptación del Bloque II, se activa el subsistema de despliegue, el cual se encarga, a través de OSM, de realizar las tareas de orquestación de recursos, conexión de componentes, configuración y aprovisionamiento del NDT. En este caso, la plataforma descrita en el capítulo 4 hace uso del software OpenStack como IaaS, vinculado con OSM como módulo VIM dentro de la arquitectura NFV-MANO. Tras iniciar la creación del NDT en forma de NS, la plataforma toma la definición realizada en el fichero NSD, así como todos los VNFD incluidos para la creación de instancias y redes, así como la configuración básica (Day-0 y Day-1) definida en los ficheros VNFD, las cuales en este caso indican el sistema operativo a utilizar por la instancia, los recursos dedicados a cada instancia (RAM, CPU y almacenamiento), creación de interfaces en cada instancia, así como la asignación de IPs de la subred que corresponda.

Como se puede observar en la **Figura 40** el NDT desplegado en forma de NS está conformado por seis instancias, emulando el comportamiento de un router, conectadas a los trece segmentos de red presentes en el gemelo real. A su vez, estas instancias están conectadas a la red definida como “*public*”, la cual es utilizada tanto para el acceso remoto a las instancias, como para conectar las instancias que actuarán de UE y core 5G. Dado que este caso de estudio se centra en la recolección de información, solo se aborda la red intermedia basada en SNMP, no obstante, en el caso II se abordan los elementos relacionados con el UE, el conector STG/UTG y el core de red 5G.

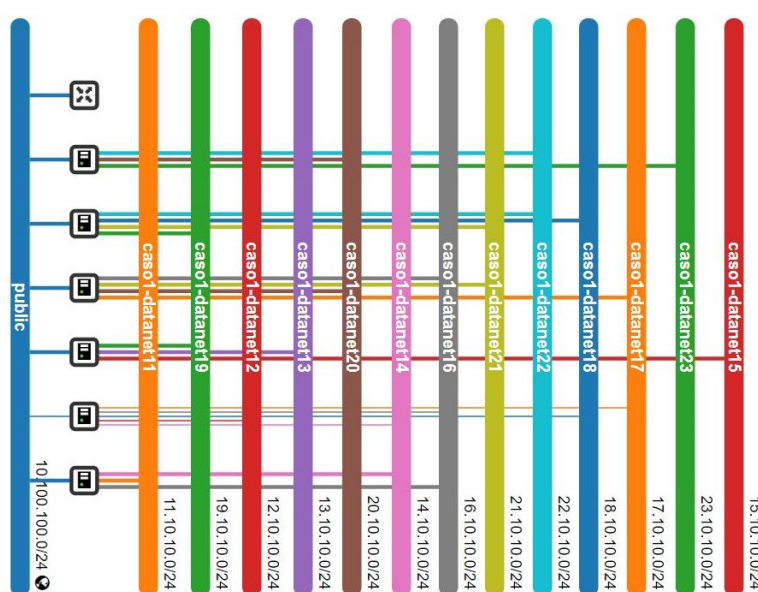


Figura 40. Caso I - Despliegue de NDT sobre OpenStack

Adicionalmente a la **Figura 40**, se puede obtener el detalle de direccionamiento e interfaces de red desplegadas en la **Figura 41** según se muestra en la interfaz web de OpenStack.

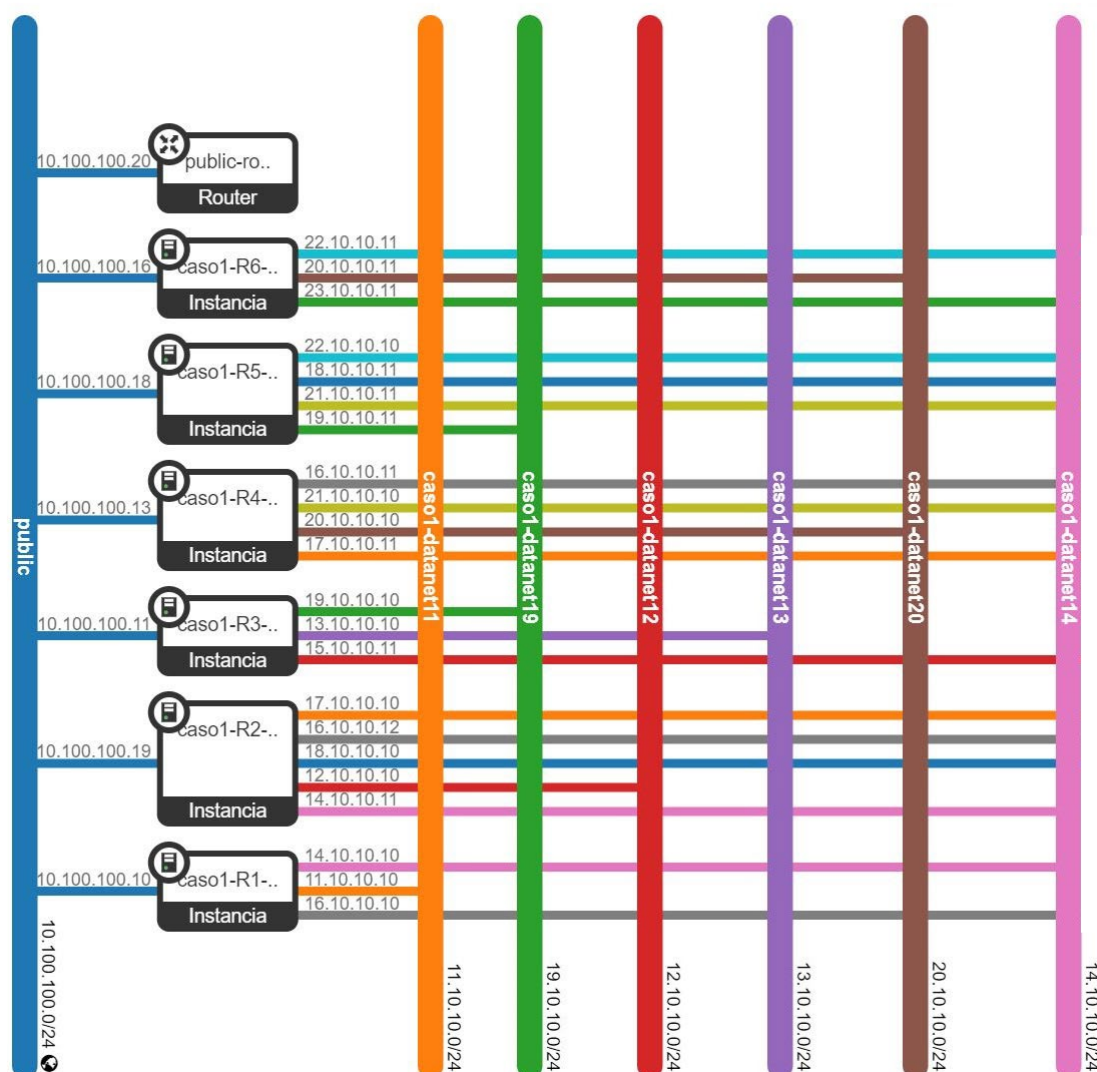


Figura 41. Caso I - Detalle despliegue de NDT sobre OpenStack

Tal y como se ha mencionado anteriormente, el fichero NSD se encarga del establecimiento de las conexiones a través del campo *“virtual-link-profile”* de la especificación NFV-SOL006, en el cual se fijan diferentes atributos como el identificador a gestionar por OSM-OpenStack, o parámetros de capa tres como el CIDR, la versión IP a utilizar, la habilitación del dhcp o la dirección del gateway de esa subred. A continuación, se muestra un pequeño fragmento correspondiente al Day-0 del fichero NSD:

```
virtual-link-profile:
  - id: datanet11
    virtual-link-desc-id: datanet11
    virtual-link-protocol-data:
      l3-protocol-data:
        cidr: 11.10.10.0/24
        ip-version: ipv4
        gateway-ip: 11.10.10.1
        dhcp-enabled: 'true'
  - id: datanet12
    virtual-link-desc-id: datanet12
    virtual-link-protocol-data:
      l3-protocol-data:
        cidr: 12.10.10.0/24
        ip-version: ipv4
        gateway-ip: 12.10.10.1
        dhcp-enabled: 'true'
```

Estos parámetros son interpretados por OSM en el despliegue, encargando a OpenStack la creación de las diferentes subredes presentes en el NDT tal y como se puede observar en la **Figura 42**. Permitiendo poder realizar ajustes en las subredes en caso de ser necesario.

☐	Name	Subnets Associated	Shared	External	Status	Admin State	Availability Zones	Actions
☐	caso1-datanet11	caso1-datanet11-subnet 11.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	public	pubic-subnet 10.100.100.0/24	No	Sí	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet19	caso1-datanet19-subnet 19.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet12	caso1-datanet12-subnet 12.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet13	caso1-datanet13-subnet 13.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet20	caso1-datanet20-subnet 20.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet14	caso1-datanet14-subnet 14.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet16	caso1-datanet16-subnet 16.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet21	caso1-datanet21-subnet 21.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet22	caso1-datanet22-subnet 22.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet18	caso1-datanet18-subnet 18.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet17	caso1-datanet17-subnet 17.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet23	caso1-datanet23-subnet 23.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v
☐	caso1-datanet15	caso1-datanet15-subnet 15.10.10.0/24	No	No	Activo	ARRIBA	-	Editar red v

Figura 42. Caso I - Listado de subredes asociadas al NDT desplegado en OpenStack

De forma complementaria a las subredes, la **Figura 43** muestra el detalle de cada una de las instancias desplegadas sobre OpenStack para conformar el NDT. En este caso, el fichero VNFD es el encargado de reflejar detalles como la imagen a utilizar por OpenStack para el despliegue de la instancia (ubuntu20.04 en este caso), los ficheros cloud-config.txt adjuntos al paquete VNFD.tar.gz, el cual recoge las acciones de configuración de tipo Day-1, las interfaces, así como la asignación de dirección IP o los recursos virtuales solicitados al VIM para el despliegue de la instancia. A continuación, se muestra un pequeño fragmento correspondiente al Day-0 y Day-1 del fichero VNFD:

```
vdu:
-   id: R2-vdu
    name: R2-vdu
    description: R2-vdu
    cloud-init-file: cloud-config.txt
    sw-image-desc: ubuntu20.04
    virtual-compute-desc: R2-vdu-compute
    virtual-storage-desc:
    - R2-vdu-storage
    int-cpd:
    -   id: eth1-int
        virtual-network-interface-requirement:
        -   name: eth1
            ip-address: 12.10.10.10
            virtual-interface:
                type: PARAVIRT
    virtual-compute-desc:
    -   id: R2-vdu-compute
        virtual-cpu:
            num-virtual-cpu: 1
        virtual-memory:
            size: 1.0
```

Estos parámetros son interpretados por OSM en el despliegue tras la tarea de despliegue del NS reflejando automáticamente la reserva de recursos, la creación y el despliegue de las instancias VNF en OpenStack. Permitiendo desde la interfaz realizar acciones como la creación de instantáneas, el apagado/encendido de las máquinas, la asignación de direcciones flotantes, edición de reglas de seguridad, o la conexión vía interfaz web a la interfaz de consola de cada una de las instancias.

☐	Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Age	Actions
☐	caso1-R6-R6-vdu-0	ubuntu20.04	caso1-datanet22 22.10.10.11 caso1-datanet20 20.10.10.11 caso1-datanet23 23.10.10.11 public 10.100.100.16	mgmtVM-vnf1-1-flv	-	Activa	nova	Ninguno	Corriendo	25 minutos	Crear instantánea ▼
☐	caso1-R5-R5-vdu-0	ubuntu20.04	caso1-datanet22 22.10.10.10 caso1-datanet18 18.10.10.11 caso1-datanet21 21.10.10.11 caso1-datanet19 19.10.10.11 public 10.100.100.18	mgmtVM-vnf1-1-flv	-	Activa	nova	Ninguno	Corriendo	25 minutos	Crear instantánea ▼
☐	caso1-R4-R4-vdu-0	ubuntu20.04	caso1-datanet16 16.10.10.11 caso1-datanet21 21.10.10.10 public 10.100.100.13 caso1-datanet20 20.10.10.10 caso1-datanet17 17.10.10.11	mgmtVM-vnf1-1-flv	-	Activa	nova	Ninguno	Corriendo	25 minutos	Crear instantánea ▼
☐	caso1-R3-R3-vdu-0	ubuntu20.04	caso1-datanet19 19.10.10.10 public 10.100.100.11 caso1-datanet13 13.10.10.10 caso1-datanet15 15.10.10.11	mgmtVM-vnf1-1-flv	-	Activa	nova	Ninguno	Corriendo	25 minutos	Crear instantánea ▼
☐	caso1-R2-R2-vdu-0	ubuntu20.04	caso1-datanet17 17.10.10.10 caso1-datanet16 16.10.10.12 caso1-datanet18 18.10.10.10 caso1-datanet12 12.10.10.10 public 10.100.100.19 caso1-datanet14 14.10.10.11	mgmtVM-vnf1-1-flv	-	Activa	nova	Ninguno	Corriendo	25 minutos	Crear instantánea ▼
☐	caso1-R1-R1-vdu-0	ubuntu20.04	caso1-datanet14 14.10.10.10 caso1-datanet11 11.10.10.10 caso1-datanet16 16.10.10.10 public 10.100.100.10	mgmtVM-vnf1-1-flv	-	Activa	nova	Ninguno	Corriendo	25 minutos	Crear instantánea ▼

Figura 43. Caso I - Listado de instancias asociadas al NDT desplegado en OpenStack

5.1.1. Conclusiones

A través de este caso de estudio se pretendía alcanzar el objetivo de obtener un ecosistema basado en gemelos digitales, capaz de modelar una red etiquetada como red real, en forma de gemelo digital de red para poder ser utilizada posteriormente como testbed para la obtención de datasets de tráfico, recuperados tanto desde el gemelo físico, como desde el gemelo digital.

En primer lugar, este caso ha permitido comprobar y validar el correcto funcionamiento del sistema DANA, encargado de recolectar los datos necesarios sobre una red real, en este caso a través de SNMP debido a la naturaleza de los dispositivos desplegados, para posteriormente modelar y adaptar esa información a la recomendación NFV SOL006 generando los descriptores NFV necesarios para desplegar el escenario en forma de gemelo digital. En segundo lugar, y en referencia a la metodología propuesta en el capítulo tres, este caso de estudio ha permitido validar correctamente la implementación de las cinco primeras fases, dejando la fase número seis (interconexión) y fase número siete (realimentación) para su estudio sobre el caso número III. Teniendo en cuenta los requisitos planteados en la sección 4.1.1, este caso de uso valida el cumplimiento de los siguientes requisitos:

- **R1.a. La plataforma debe ser capaz de llevar a cabo el modelado y adaptación de la información para desplegar el gemelo digital de la red objetivo.**

En relación con la recolección que realiza el sistema DANA, la aproximación vía SNMP homogeniza la obtención de datos independientemente del fabricante del dispositivo hardware de propósito específico desplegado en la red. Siendo este uno de los principales problemas identificados en la literatura a la hora de tratar de implantar el paradigma de gemelos digitales como una solución transversal.

- **R1.b. La plataforma debe ser capaz de gestionar el ciclo de vida completo de las instancias DT**

A través de los ficheros VNFD y NSD reflejados en la **Figura 38** y **Figura 39** se habilita el despliegue de la instancia NS formada por las VNFs desde la herramienta OpenSourceMANO. Permitiendo realizar la gestión completa del ciclo de vida del NDT.

- **R1.c. La plataforma debe ser capaz de aprovisionar de manera básica los DT.**

Mediante los descriptores VNFD para cada elemento de la red, es posible incluir el aprovisionamiento de tipo Day-0 y Day-1. En este caso se aprovisiona la imagen de router que utilizará la instancia, así como la configuración a nivel interfaces y direccionamiento completo del escenario.

- **R2.a. La plataforma debe hacer uso de tecnologías habilitadoras capaces de desplegar una red física de manera virtualizada**

Este caso de uso hace uso principalmente de la tecnología NFV para desplegar el NDT, utilizando para ello OSM como elemento MANO y OpenStack como VIM.

- **R2.c. La plataforma debe permitir la replicabilidad de los NDT a través de los descriptores NFV asociados a cada escenario.**

Al igual que el requisito R1.b, el cual permite realizar la gestión del ciclo de vida del NDT, la herramienta OSM permite la replicabilidad y despliegue múltiple del NDT generado. Pudiendo desplegar varios NDT de manera concurrente.

- **R2.d. La plataforma debe cubrir la gestión del NDT, permitiendo el despliegue, así como la eliminación de los escenarios.**

De igual manera que el requisito anterior, OSM permite gestionar completamente los despliegues de NDT, permitiendo pausar los despliegues, eliminarlos o volver a desplegarlos desde su fase inicial.

- **R2.e. La plataforma debe ser independiente al tipo de red o caso de estudio**

Debido al uso del sistema DANA, así como el resto de los submódulos presentes en el bloque de gestión, la plataforma que soporta el NDT es agnóstica al tipo de escenario a modelar y desplegar.

- **R5.a. La plataforma debe poder ser operada de manera semi automática.**

Una vez es desplegado y puesto en funcionamiento el sistema DANA para la recolección de datos, la plataforma comienza el proceso de creación y despliegue del NDT de manera automática, siguiendo un orden secuencial definido por las fases de la metodología propuesta. Únicamente sería opcional la interacción en la fase de aprovisionamiento.

- **R5.b. La plataforma debe diseñarse de manera modular**

Según se establece en el capítulo cuatro, la plataforma engloba tres grandes bloques, los cuales están conformados por diferentes submódulos y subsistemas los cuales se intercomunican a través de interfaces que garantizan la modularidad y posible extensión de la plataforma.

5.2. Caso II. Despliegue NDT red 5G

Este segundo caso de estudio comparte la recolección de datos presentada en el caso anterior, así como el modelado, adaptación y despliegue, pero se centra en la capacidad de ejecución de tareas de tipo Day-2. Estas tareas, como se presentó en la sección 3.1, permiten la ejecución de acciones bajo demanda sobre el NDT desde el sistema de gestión del Bloque II de la plataforma diseñada. Concretamente, una vez desplegado el NDT, se ejecutan las tareas de recolección de datos de tráfico con la finalidad de obtener datasets de tráfico necesarios para tareas posteriores como el entrenamiento de modelos de Machine Learning o sistemas basados en Inteligencia Artificial.

Con la intención de validar las tareas de recolección de información desde el NDT, así como su envío vía MQTT al sistema de gestión alojado en el Bloque II de la plataforma, se despliega el escenario reflejado en la **Figura 44**. Este escenario, desplegado sobre VirtualBox, se utilizará como gemelo físico, formado por cuatro máquinas Ubuntu conectadas entre sí a través de diferentes segmentos de red, cada una de ellas con un agente satélite instalado. En este caso, el escenario a desplegar presenta una topología más simplificada, evitando la red de agregación presentada en el caso I, y utilizando la máquina UE junto al conector STG-UTG para emular el *Radio Access Network* (RAN) de la red 5G. Adicionalmente, esta topología permitirá la validación del funcionamiento del agente DANA en lo referente a la recolección de datos sobre equipos de hardware genérico, los cuales alberga diferentes tecnologías de virtualización como puede ser VirtualBox o Docker.

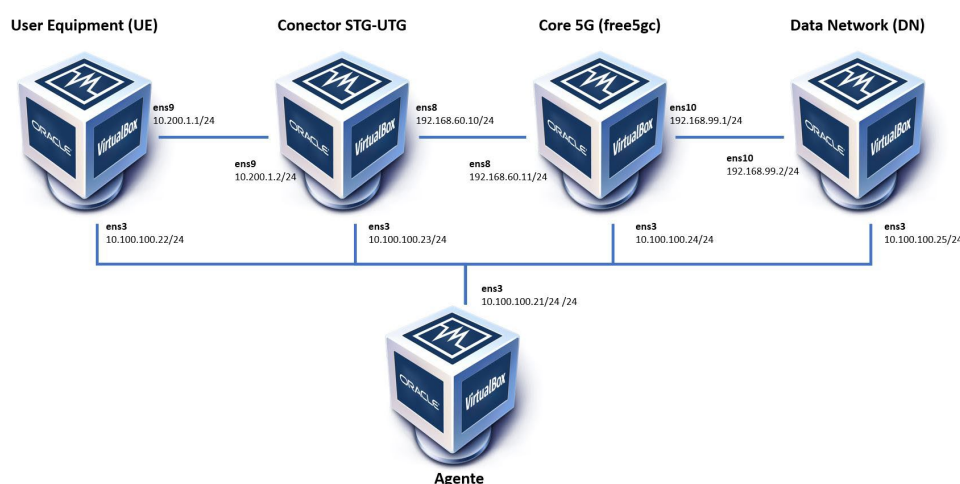


Figura 44. Caso II - Red gemelo físico

- El terminal de usuario o UE corresponde a la máquina desde la cual se ejecutarán las peticiones a la Data Network, haciendo uso de la red 5G. Esta máquina está conectada directamente al conector STGUTG, elemento encargado de establecer la comunicación con las NF del core 5G.
- El conector STGUTG alojado en una máquina Ubuntu 18.04 LTS, según las especificaciones reflejadas en su página de Github [139], y configurada en base al escenario propuesto, fijando como UE la dirección IP de la primera máquina.
- Máquina con la instancia Free5GC desplegada a través de la tecnología Docker [166], en su versión v3.2.1 donde se configura la información necesaria para que

el conector STGUTG pueda acceder a la sesión de datos PDU (*Packet Data Unit*) para el terminal de usuario.

- Data Network, máquina con acceso a internet, la cual tiene diferentes softwares instalado en modo servidor.

Una vez presentado el escenario que será utilizado como gemelo físico, se procede a conectar el agente inteligente, presentado en la sección 3.1.1, a la red común establecida para todas las máquinas del escenario.

De igual manera que en el caso I, el sistema se encuentra en un estado inicial en el cual no existe aún el NDT, por lo que el proceso de creación comienza con el envío del flag de estado por parte del Bloque II a través del canal de gestión correspondiente vía MQTT. Este flag es procesado por el subsistema de control presente en el agente, el cual activa el subsistema de recolección. En este caso, a diferencia del caso anterior, el agente se encarga de la recolección de información sobre equipos de propósito general, tal y como se establece en la sección 3.1.1.2. El agente comienza a recibir la información obtenida por los agentes satélite presentes en cada una de las cuatro máquinas virtuales que conforman el escenario, obteniendo información reflejada en el modelo de datos JSON que genera el agente inteligente. A continuación, se muestra el reporte generado, alineado al modelo de datos propuesto, a través de la información recibida por parte del agente satélite presente en la máquina que alberga el core de red 5G.

```
agent_topology_data:
  uuid:
  name: core5g
  desc: escenario caso II
  so: ubuntu20.04
  disk: 20GB
  cpu: 4
  ram: 8
  ifaces_up:
    ifaces_sw:
      ifaces_sw_ip:
      ifaces_sw_name:
      ifaces_sw_mac:
      ifaces_sw_num:
    ifaces_hw:
      ifaces_hw_ip: [10.100.100.24, 192.168.60.11, 192.168.99.1]
      ifaces_hw_name: [ens3, ens8, ens10]
      ifaces_hw_mac:
```

```
        ifaces_hw_num:[1,2,3]
virtualbox:
    is_installed: false
    is_running:    false
docker:
    is_installed: true
    is_running:    true
kubernetes:
    is_installed: false
    is_running:    false
```

Una vez la tarea de recolección finaliza, tras obtener los reportes de los cuatro agentes satélite presentes en el escenario, el agente inteligente obtiene la información necesaria para la representación topológica y el aprovisionamiento de recursos en el VIM. Tras esta tarea, el agente inteligente pasa a activar el subsistema de envío de datos, el cual se encarga a través del conector MQTT de enviar el fichero JSON resultante al topic TOPOLOGY haciendo uso del flujo de comunicación PT2MANO, como se puede observar en la **Tabla 17**.

Llegado a este punto, tras la publicación del fichero JSON en el topic “topology” del broker MQTT alojado en el sistema de gestión (Bloque II), el agente detiene el subsistema de recolección de datos, quedándose suscrito al topic de gestión a la espera de recibir la siguiente orden a través del canal de gestión controlado por el Bloque II.

De manera dual, tal y como se establece en el apartado **4.1.2.2**, el sistema de gestión implementa un cliente MQTT suscrito al topic TOPOLOGY, encargado de recibir el fichero JSON reportado por el agente inteligente desplegado en el entorno físico. Tras obtener el fichero JSON que

modela la topología del gemelo físico, el sistema de gestión activa el subsistema de adaptación encargado de procesar el fichero JSON para la obtención de los n ficheros VNFD, así como el fichero NSD que describirá la implementación del NDT en el entorno digital.

El resultado de la tarea realizada por el subsistema de adaptación es la obtención y subida de los descriptores VNF y NS a la plataforma OpenSourceMANO. En la **Figura 45**, **Figura 46** y **Figura 47** se observa el resultado de esta iteración, tras haber llevado a cabo la adaptación a la especificación NFV-SOL006 y verificado la correcta sintaxis de los ficheros tar.gz generados.

Product Name	Identifier	Version	Provider	Type	Description	Actions
dn	eb48886a-36ce-4d1b-8f2c-c6b1d703913d	1.0	UPM_tesis	vnfd	Data Network	  Action
free5gc	f4ffc048-0a0c-4151-a31d-f87ea585b234	1.0	UPM_tesis	vnfd	Core 5G basado en free5gc	  Action
stgutg	3b582dbb-414b-41d4-966b-c9644343e3f3	1.0	UPM_tesis	vnfd	Conector STGUTG	  Action
ue	0efe2471-b7dc-4e79-ab64-aea47725d8b4	1.0	UPM_tesis	vnfd	UE - User Equipment	  Action

Figura 45. Caso II - Ficheros VNFD cargados en OSM

En este caso, los cuatro ficheros VNFD generados para desplegar el NDT contienen los ficheros cloud-config.txt, los cuales definen las instrucciones de instalación de software específico, como puede ser Docker, el repositorio STGUTG o la implementación free5gc como acciones de tipo day1, las cuales se llevarán a cabo de manera automática en el momento de instanciar el NS que despliega el NDT. A continuación, se muestra contenido del fichero cloud-config.txt asociado a la VNFD del core 5G.

```
#cloud-config
packages:
  - git
password: escenario_2
chpasswd: { expire: False }
ssh_pwauth: True
ranked:
  - 'curl -fsSL https://get.docker.com | sh'
  - 'cd /home'
  - 'git clone https://github.com/free5gc/free5gc-compose.git'
  - 'cd free5gc-compose && docker compose up'
```

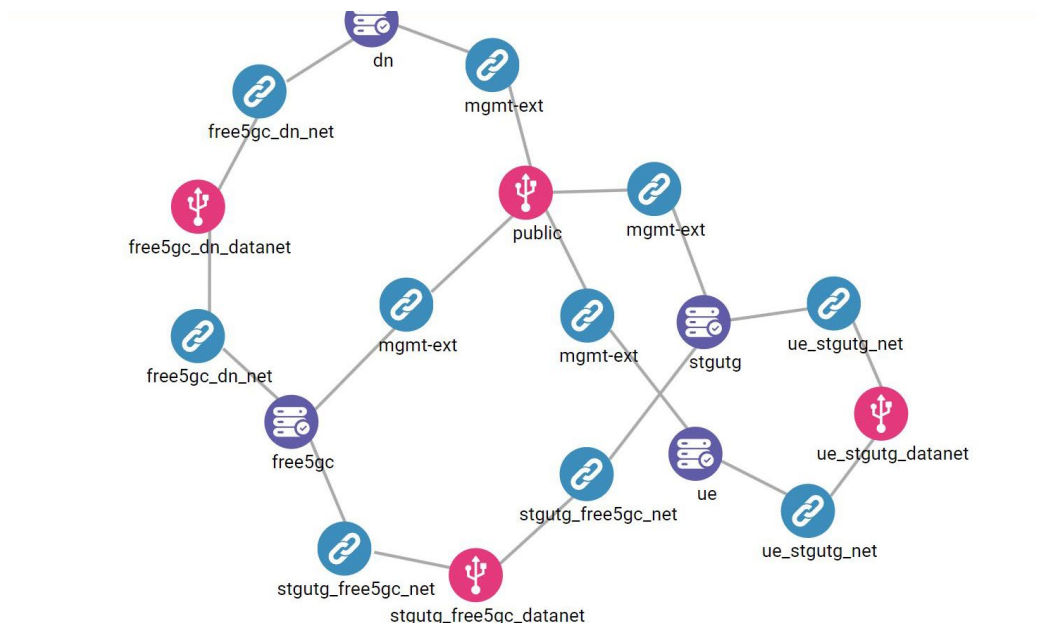


Figura 46. Caso II - Representación gráfica del fichero NSD desde OSM

De igual manera que en el caso I, desde la interfaz web de gestión de OSM es posible verificar el estado de los ficheros generados, tanto de los VNFD correspondientes a cada máquina virtual del gemelo físico, como el fichero NSD encargado de definir todas las interconexiones del escenario. En este momento se completa la fase III de la metodología, dando paso a las fases IV y V, despliegue y aprovisionamiento respectivamente.

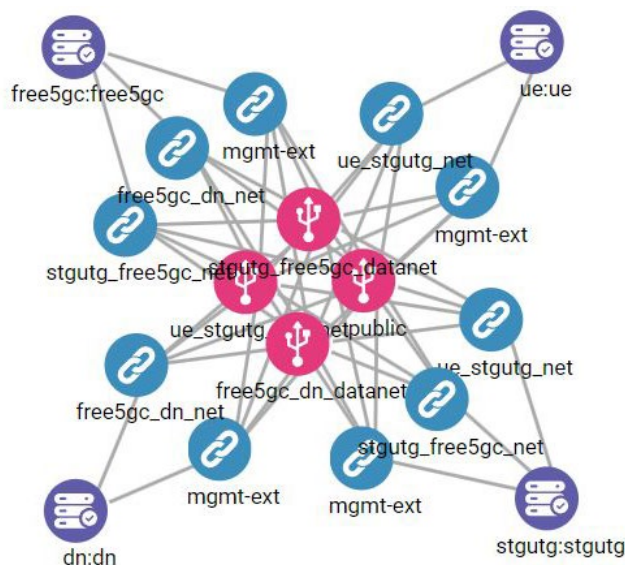


Figura 47. Caso II - Gemelo Digital lanzado como instancia NS lanzada desde OSM

Una vez terminada la tarea de adaptación a la especificación NFV-SOL006, por parte del subsistema de adaptación del Bloque II. Incluyendo en este caso la configuración específica a través de los ficheros cloud-config.txt para cada VNF y la acción de tipo Day-2. Se activa el subsistema de despliegue del NDT en forma de NS, la plataforma toma la definición realizada en el fichero NSD, así como todos los VNFD incluidos para la creación de instancias y redes, así como la configuración básica (Day-0 y Day-1) y configuración específica (Day-2) definida en los ficheros VNFD. A continuación, se muestra un fragmento del fichero VNFD correspondiente al core 5G, el cual habilita la opción de ejecutar la primitiva de Day-2, en forma de chart 'ntopchart' para la activación del sniffer ntop que permitirá la obtención de datasets bajo demanda.

```
vdu-profile:
- id: free5gc-vdu
  min-number-of-instances: 1
lcm-operations-configuration:
  operate-vnf-op-config:
    day1-2:
      - config-access:
          ssh-access:
            default-user: ubuntu
            required: true
        config-primitive:
          - execution-environment-primitive: sniff
            execution-environment-ref: monitor
            name: sniff
            parameter:
              - data-type: STRING
                default-value: /home/ubuntu/ntop.sh
                name: script-path
          - execution-environment-primitive: sleep
            execution-environment-ref: monitor
            name: sleep
        execution-environment-list:
          - external-connection-point-ref: vnf-mgmt-ext
            helm-chart: ntopchart
            id: monitor
      id: free5gc
    initial-config-primitive:
      - execution-environment-ref: monitor
        name: config
        parameter:
          - name: ssh-hostname
            value: <rw_mgmt_ip>
          - name: ssh-username
            value: ubuntu
```

```

- name: ssh-password
  value: escenario_2
  seq: 1
- execution-environment-ref: monitor
  name: sniff
  parameter:
    - name: file-path
      value: sh /home/ubuntu/ntop.sh
    seq: 2

```

Tras la tarea de despliegue por parte del subsistema correspondiente, el resultado se refleja en el entorno digital alojado en el VIM OpenStack, al igual que lo expuesto anteriormente en el caso I. En la particularidad del caso II, se realiza el despliegue de instancias Ubuntu a través de OpenStack, con la creación de las subredes especificadas en el fichero NSD, tal y como se observa en la **Figura 48**. Adicionalmente, durante la fase de instanciación, se ejecuta vía cloud-init las instrucciones reflejadas en los ficheros cloud-config.txt adjuntados a cada uno de los paquetes VNFD.tar.gz generado por el subsistema de adaptación. En este caso concreto se encarga del clonado de los repositorios necesarios, así como la instalación de docker para el posterior despliegue de free5gc a través de Docker Compose.

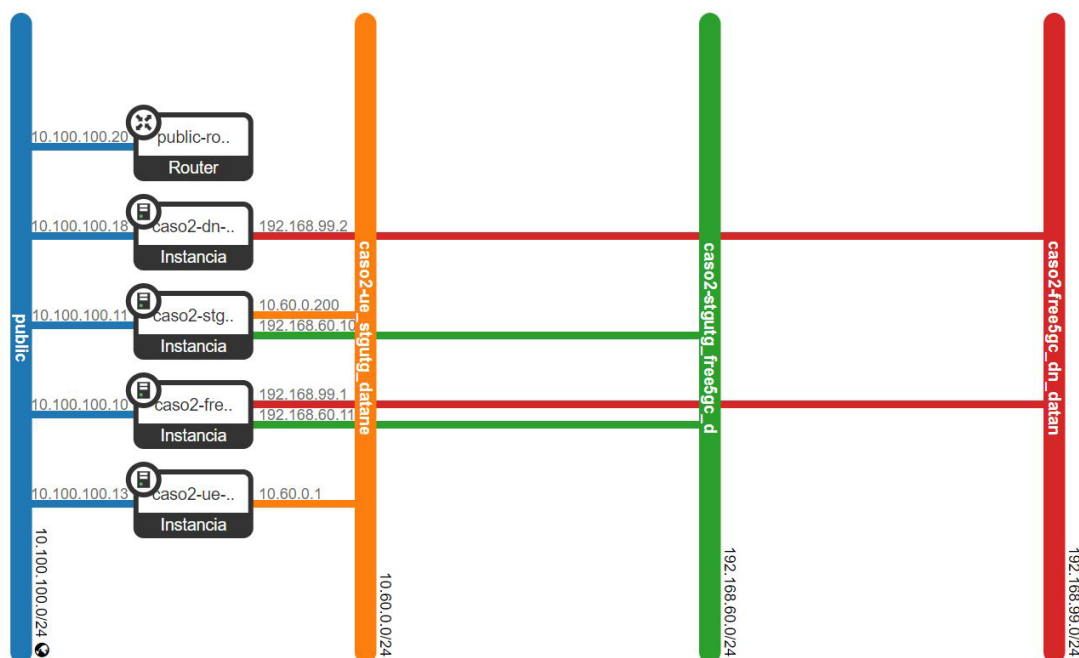


Figura 48. Caso II - Representación gráfica del Gemelo Digital de red 5G desplegado sobre el VIM OpenStack conectado a OSM.

Al igual que en el caso anterior, en la **Figura 49** y **Figura 50** se puede observar el detalle en referencia a las instancias levantas en OpenStack, basadas imágenes Ubuntu, y las subredes necesarias para interconectar el gemelo digital que replica el escenario físico.

☐ Name	Subnets Associated	Shared	External
☐ caso2-ue_stgutg_datane	caso2-ue_stgutg_datane-subnet 10.60.0.0/24	No	No
☐ public	pubic-subnet 10.100.100.0/24	No	Si
☐ caso2-stgutg_free5gc_d	caso2-stgutg_free5gc_d-subnet 192.168.60.0/24	No	No
☐ caso2-free5gc_dn_datan	caso2-free5gc_dn_datan-subnet 192.168.99.0/24	No	No

Figura 49. Caso II - Listado de subredes asociadas al NDT desplegado en OpenStack

☐ Instance Name	Image Name	IP Address	Flavor
☐ caso2-dn-dn-vdu-0	ubuntu20.04	public 10.100.100.18 caso2-free5gc_dn_datan 192.168.99.2	mgmtVM-vnf1-1-flv
☐ caso2-stgutg-stgutg-vdu-0	ubuntu20.04	public 10.100.100.11 caso2-ue_stgutg_datane 10.60.0.200 caso2-stgutg_free5gc_d 192.168.60.10	mgmtVM-vnf1-1-flv
☐ caso2-free5gc-free5gc-vdu-0	ubuntu20.04	public 10.100.100.10 caso2-free5gc_dn_datan 192.168.99.1 caso2-stgutg_free5gc_d 192.168.60.11	mgmtVM-vnf1-1-flv
☐ caso2-ue-ue-vdu-0	ubuntu20.04	caso2-ue_stgutg_datane 10.60.0.1 public 10.100.100.13	mgmtVM-vnf1-1-flv

Figura 50. Caso II - Listado de instancias asociadas al NDT desplegado en OpenStack

Tras la instanciación del NDT sobre OpenStack, el gemelo digital está preparado para poder ser utilizado como ‘*testbed*’ sobre el cual ejecutar diferentes experimentos. A modo de contextualización sobre la necesidad de la obtención de datasets, desde el punto de vista de la ciberseguridad, se contemplan dos tipos de ataques a este tipo de redes, el criptominado y el ataque a infraestructura DNS [151] [167]. Pese a que la realización de este tipo de ataques sobre el NDT queda fuera del alcance de la tesis, si se engloba el mecanismo que posibilita la obtención de datasets desde el NDT con el fin de poder ser utilizados para entrenar modelos de análisis de datos avanzados capaces de detectar patrones anómalos con el fin de tomar medidas preventivas sobre la red real.

A continuación, nos centraremos en la explicación de esta acción de tipo Day-2, tanto a nivel de activación como de funcionamiento de la herramienta ntop que se ejecuta dentro

del NDT. En primer lugar, como se ha presentado en este caso de estudio, el fichero VNFD asociado al core de red 5G implementa una acción de tipo Day-2 la cual hace uso de un paquete Helm-chart (ntopchart). Este paquete se despliega dentro de la VNF, permitiendo activar bajo demanda desde la interfaz OSM la ejecución del software ntop para monitorizar el core 5G. Como se puede observar en la **Figura 51**, el resultado de la ejecución de la acción de tipo Day-2 arranca el sistema ntop el cual ofrece una interfaz web que permite la monitorización en la interfaz 'br-bridge5gc' que despliega la implementación *free5gc* basada en *docker-compose*. A partir de este momento la herramienta está analizando el tráfico cursado a través de la interfaz común a todas las NF que conforman el núcleo de red 5G, obteniendo gráficas en tiempo real sobre el tráfico total cursado, tráfico entrante/saliente, así como el porcentaje de almacenamiento en forma de series temporales, tamaño de MTU o velocidad de la interfaz.

Adicionalmente a esta información, nos centramos en la capacidad de la herramienta para obtener capturas en directo de todo el tráfico cursado a través de esta interfaz, permitiendo obtener capturas en rangos temporales acotados que va desde los 10 minutos a los 30 segundos. Tras la ejecución de dicha acción, el resultado es la obtención de archivos de tráfico en formato pcap ordenados temporalmente.

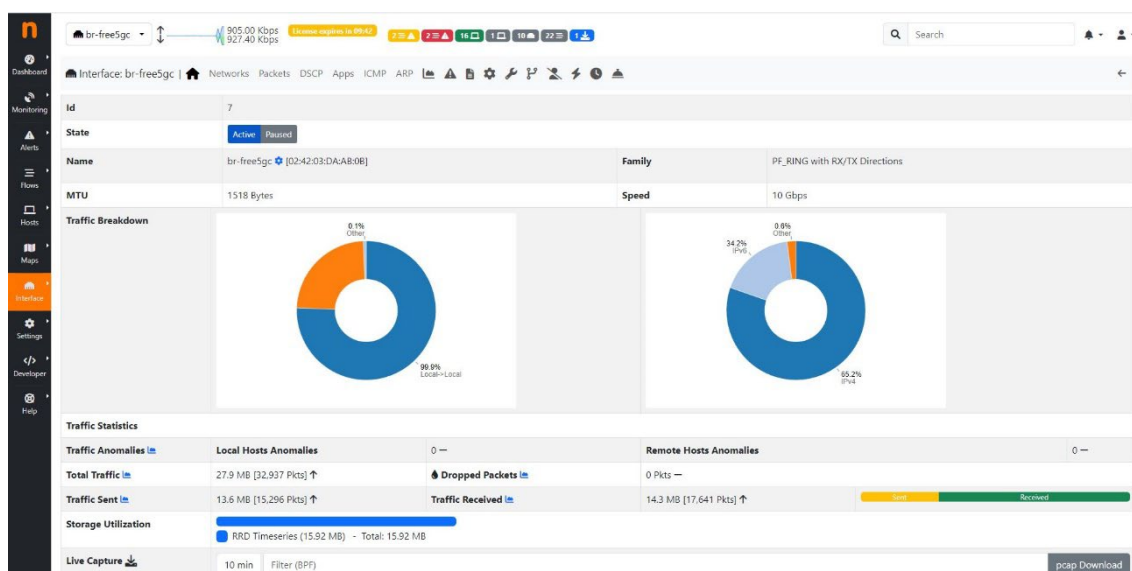


Figura 51. Core5G monitorizado con la herramienta ntop-ng

Pese a que en la figura anterior se expone la capacidad de obtención de todo el tráfico, únicamente centrándose en la interfaz en la que escucha ntop, esta herramienta permite además obtener el listado de máquinas o contenedores que se encuentran conectados a la

interfaz br-free5gc tal y como se muestra en la **Figura 52**. En este caso concreto, dada la implementación desplegada en la VNF del core 5G, estos hosts corresponden a cada uno de los contenedores docker asociados a cada una de las NF que conforman el núcleo de red 5G. De esta manera, es posible obtener información detallada del comportamiento de cada una de las funciones de red, así como su ancho de banda o el total de tráfico cursado para cada una de ellas.

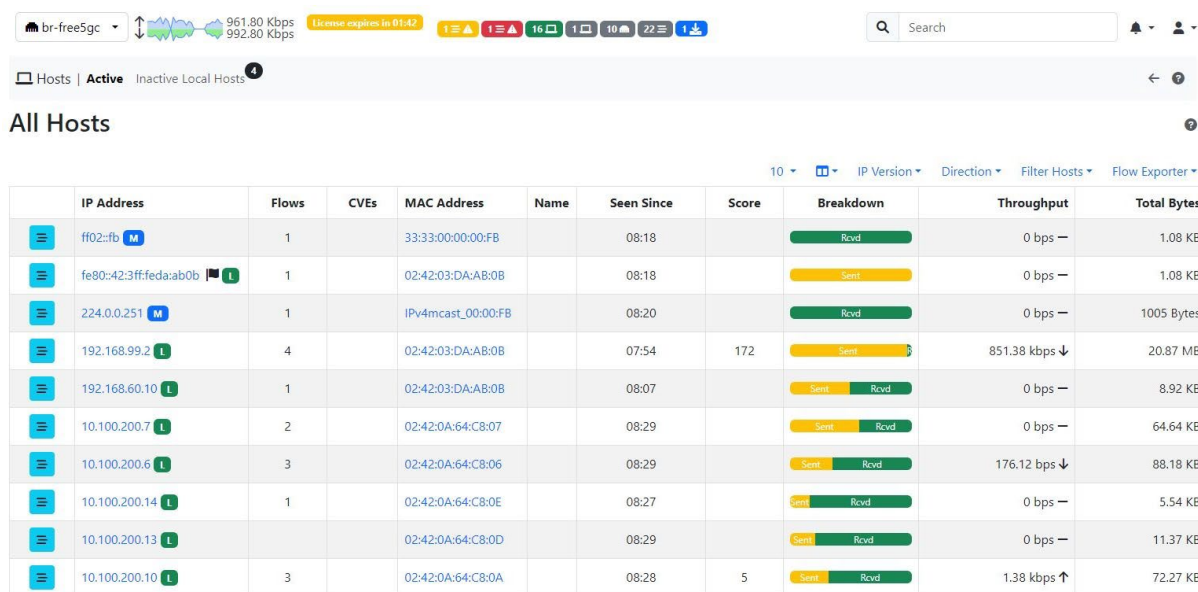


Figura 52. Detalle de la monitorización de cada VNF del core 5G

Esta posibilidad es especialmente útil de cara al análisis independiente del comportamiento, ante determinadas pruebas de concepto o emulación de ataques sobre la red 5G, de funciones de red concretas. A modo de ejemplo, en la **Figura 53**, se puede observar la información disponible para la función de red encargada del plano de usuario (User Plane Function – UPF), la cual se encarga, entre otras tareas de gestionar y dirigir el tráfico de datos desde el UE hacia el exterior, la gestión del handover, la aplicación de políticas de red, o la optimización de tráfico. Al igual que en la visual completa de toda la interfaz br-free5gc, es posible la obtención de pcaps en franjas temporales de los elementos individuales, como el UPF en la **Figura 53**, o el tráfico cursado por el conector STGUTG hacia el core5G reflejado en la **Figura 54**

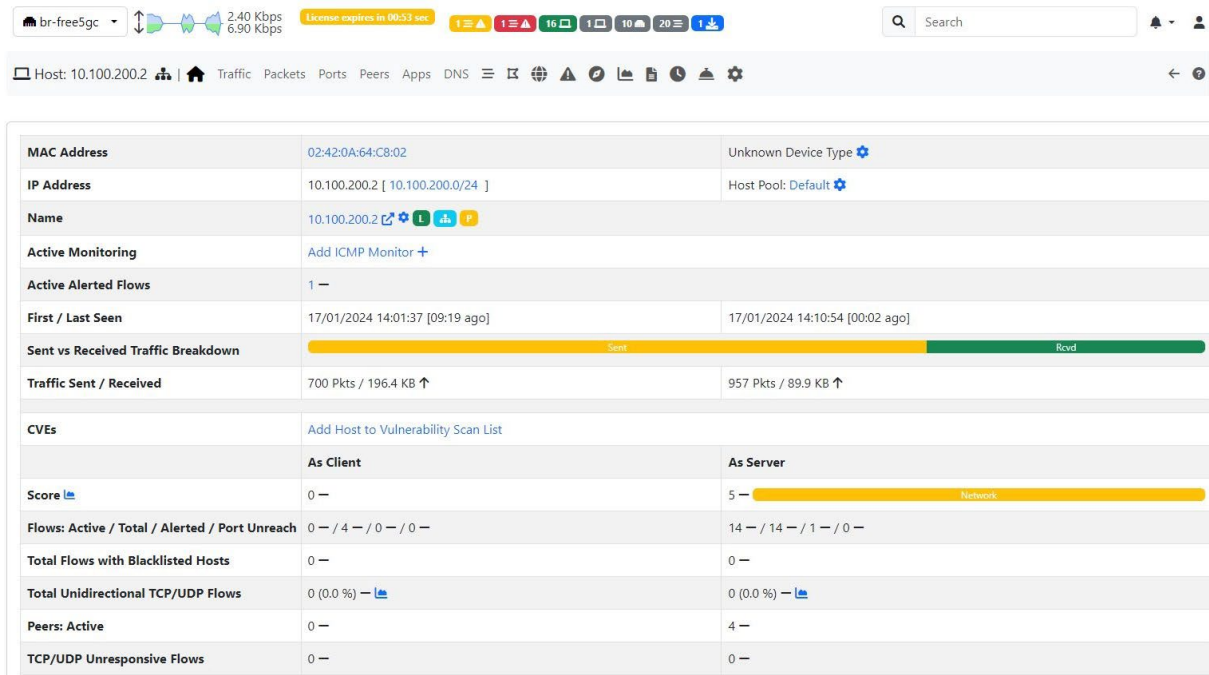


Figura 53. Estadísticas de uso asociadas al UPF

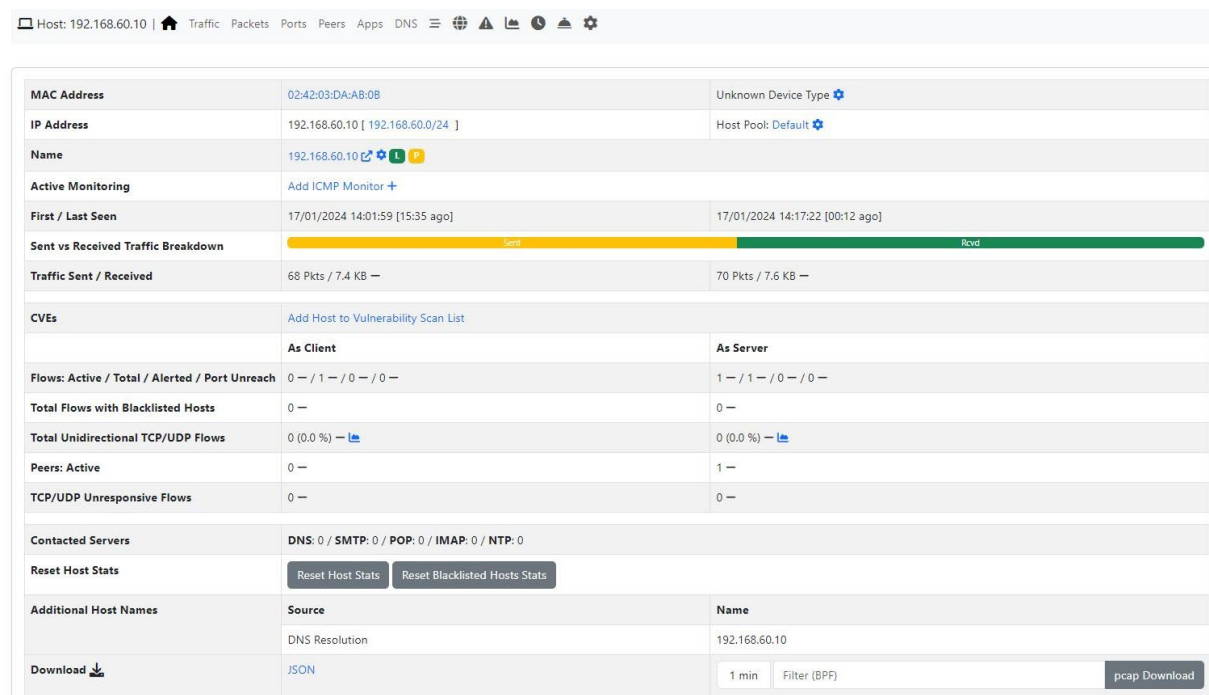
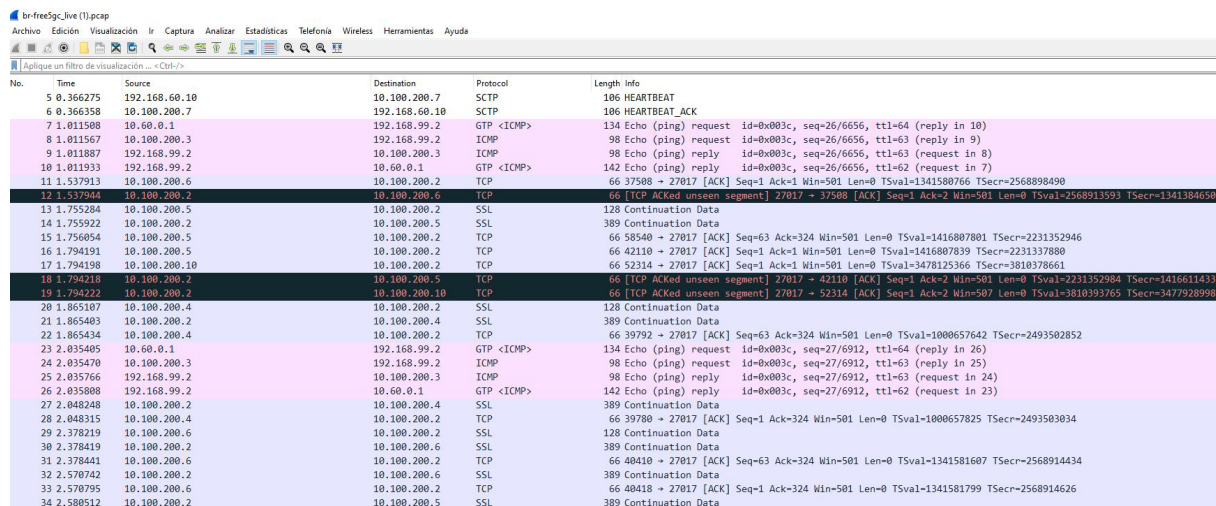


Figura 54. Estadísticas de uso asociadas al conector STGUTG

Finalmente, en la **Figura 55**, se muestra el resultado completo del caso II, tras desplegar el NDT y utilizando el UE, STGUTG, core5G para cursar una serie de peticiones que reflejen tráfico de red capturado por ntop. Obteniendo un pcap que contiene las negociaciones realizadas por parte del UE-STGUTG con el core de red 5G, así como el tráfico cursado a

través del túnel GTP entre el UE y la máquina que actual como Data Network. Llegado a este punto, el ecosistema NDT tiene activos los tres bloques identificados en el capítulo cuatro.



No.	Time	Source	Destination	Protocol	Length	Info
5	0.366275	192.168.60.10	10.100.200.7	Sctp	106	HEARTBEAT
6	0.366358	10.100.200.7	192.168.60.10	Sctp	106	HEARTBEAT_ACK
7	1.011508	10.60.0.1	192.168.99.2	GTP <ICMP>	134	Echo (ping) request id=0x003c, seq=26/6656, ttl=64 (reply in 10)
8	1.011567	10.100.200.3	192.168.99.2	ICMP	98	Echo (ping) request id=0x003c, seq=26/6656, ttl=63 (reply in 9)
9	1.011887	192.168.99.2	10.100.200.3	ICMP	98	Echo (ping) reply id=0x003c, seq=26/6656, ttl=63 (request in 8)
10	1.011933	192.168.99.2	10.60.0.1	GTP <ICMP>	142	Echo (ping) reply id=0x003c, seq=26/6656, ttl=62 (request in 7)
11	1.537913	10.100.200.6	10.100.200.2	TCP	66	37508 → 27017 [ACK] Seq=1 Ack=1 Win=501 Len=0 TSval=1341580766 TSecr=256898490
12	1.537944	10.100.200.2	10.100.200.6	TCP	66	[TCP ACKed unseen segment] 27017 → 37508 [ACK] Seq=1 Ack=2 Win=501 Len=0 TSval=2568913593 TSecr=1341384650
13	1.755284	10.100.200.5	10.100.200.2	SSL	128	Continuation Data
14	1.755922	10.100.200.2	10.100.200.5	SSL	389	Continuation Data
15	1.756054	10.100.200.5	10.100.200.2	TCP	66	58540 → 27017 [ACK] Seq=63 Ack=324 Win=501 Len=0 TSval=1416807801 TSecr=2231352946
16	1.794191	10.100.200.5	10.100.200.2	TCP	66	42110 → 27017 [ACK] Seq=1 Ack=1 Win=501 Len=0 TSval=1416807839 TSecr=2231337880
17	1.794198	10.100.200.10	10.100.200.2	TCP	66	52314 → 27017 [ACK] Seq=1 Ack=1 Win=501 Len=0 TSval=3478125366 TSecr=3810378661
18	1.794218	10.100.200.2	10.100.200.5	TCP	66	[TCP ACKed unseen segment] 27017 → 42110 [ACK] Seq=1 Ack=2 Win=501 Len=0 TSval=12231352884 TSecr=1416611433
19	1.794222	10.100.200.2	10.100.200.10	TCP	66	[TCP ACKed unseen segment] 27017 → 52314 [ACK] Seq=1 Ack=2 Win=501 Len=0 TSval=3810393765 TSecr=3477928998
20	1.865107	10.100.200.4	10.100.200.2	SSL	128	Continuation Data
21	1.865403	10.100.200.2	10.100.200.4	SSL	389	Continuation Data
22	1.865434	10.100.200.4	10.100.200.2	TCP	66	39792 → 27017 [ACK] Seq=63 Ack=324 Win=501 Len=0 TSval=1000657642 TSecr=2493502852
23	2.035405	10.60.0.1	192.168.99.2	GTP <ICMP>	134	Echo (ping) request id=0x003c, seq=27/6912, ttl=64 (reply in 26)
24	2.035470	10.100.200.3	192.168.99.2	ICMP	98	Echo (ping) request id=0x003c, seq=27/6912, ttl=63 (reply in 25)
25	2.035766	192.168.99.2	10.100.200.3	ICMP	98	Echo (ping) reply id=0x003c, seq=27/6912, ttl=63 (request in 24)
26	2.035808	192.168.99.2	10.60.0.1	GTP <ICMP>	142	Echo (ping) reply id=0x003c, seq=27/6912, ttl=62 (request in 23)
27	2.048248	10.100.200.2	10.100.200.4	SSL	389	Continuation Data
28	2.048315	10.100.200.4	10.100.200.2	TCP	66	39780 → 27017 [ACK] Seq=1 Ack=324 Win=501 Len=0 TSval=1000657825 TSecr=2493503034
29	2.378219	10.100.200.6	10.100.200.2	SSL	128	Continuation Data
30	2.378419	10.100.200.2	10.100.200.6	SSL	389	Continuation Data
31	2.378441	10.100.200.6	10.100.200.2	TCP	66	40410 → 27017 [ACK] Seq=63 Ack=324 Win=501 Len=0 TSval=1341581607 TSecr=2568914434
32	2.570742	10.100.200.2	10.100.200.6	SSL	389	Continuation Data
33	2.570795	10.100.200.6	10.100.200.2	TCP	66	40410 → 27017 [ACK] Seq=1 Ack=324 Win=501 Len=0 TSval=1341581799 TSecr=2568914626
34	2.580512	10.100.200.2	10.100.200.5	SSL	389	Continuation Data

Figura 55. Ejemplo del Dataset obtenido desde el NDT vía ntop

En el Bloque III, tal y como se recoge en el apartado 4.1.2.3, sobre el cual se ha desplegado el NDT, se despliega adicionalmente una máquina conectada a la red '*public*' de OpenStack, con conectividad a todas las instancias de gemelo digital del caso II. Esta máquina es la encargada de alojar el agente inteligente encargado de obtener información del NDT y posteriormente propagarla hacia el gemelo físico, en este caso utilizando el flujo de comunicación DT2MANO. Tras realizar los diferentes experimentos o pruebas sobre el NDT de red 5, y haber activado la tarea de sniffer a través de la acción de tipo Day-2, el agente inteligente desplegado en el NDT se encarga de tomar los pcaps de tráfico resultantes para poder enviarlos en primer lugar al sistema de gestión para su almacenamiento y en segundo lugar posibilitar la opción de que el agente inteligente desplegado en el entorno físico lo reciba y pueda realizar una reinyección en la red real.

Para llevar a cabo este intercambio de información referente a capturas de tráfico recogidas por el software ntop-ng, las cuales se almacenan en la instancia que ejecuta el core 5G, el sistema de gestión pública, vía MQTT en el topic de gestión, el flag encargado de activar el script de sniffer presente en el agente, el cual conecta con la instancia del core y comienza a traer los ficheros pcaps generados. Cada vez que el agente obtiene un fichero pcap, este es procesado por el software *watchdog* el cual junto al conector SCP, se

encarga de publicar el fichero en el sistema de ficheros desplegado en el sistema de gestión, haciendo uso del canal de comunicación DT2MANO.

Una vez publicados los diferentes archivos en formato pcap de tráfico obtenidos a través del NDT con la herramienta ntop, el sistema de gestión, a través del controlador se encarga de recuperar el pcap para su almacenamiento y dando la opción al operador del sistema, de reenviar el pcap hacia el gemelo físico a través del canal MANO2PT.

5.2.1. Conclusiones

En este escenario, el objetivo de este caso de estudio es, en primer lugar, la validación del uso del NDT como sistema real para ser usado como ‘testbed’ de experimentación para la obtención de datasets, mediante la ejecución vía acción Day-2 del software de analizador de tráfico ntop, que alimenten algoritmos de machine Learning o Inteligencia Artificial que se integren a la capa de aplicación del NDT y que puedan ser utilizados para tareas como la detección de incidencias de seguridad o la optimización de recursos o tráfico en la red real, y en segundo validar el correcto funcionamiento del canal de comunicación encargado de transferir los distintos pcaps de tráfico hacia el sistema de gestión y al entorno físico.

Teniendo en cuenta los requisitos planteados en la sección 4.1.1, este caso de uso valida el cumplimiento de los siguientes requisitos:

- **R1.a. La plataforma debe ser capaz de llevar a cabo el modelado y adaptación de la información para desplegar el gemelo digital de la red objetivo.**

Al igual que en el caso anterior, el sistema DANA es el encargado de obtener la información necesaria sobre el gemelo físico. La diferencia respecto al caso anterior es que en este caso los equipos a modelar no son equipos de hardware específico sondeables vía SNMP. En este caso, se utiliza la segunda vía de funcionamiento del sistema DANA, el cual recolecta información a través de los agentes satélite instalados en los equipos. De esta forma se valida que el sistema DANA es capaz de realizar la recolección de datos sobre un grupo heterogéneo de sistemas.

- **R1.b. La plataforma debe ser capaz de gestionar el ciclo de vida completo de las instancias DT**

De igual manera al caso anterior, el NDT es caracterizado a través de los ficheros VNFD y NSD, tal y como se puede observar en la **Figura 45**, **Figura 46** y **Figura 47**. Los cuales permiten a OSM realizar el ciclo de vida completo.

- **R1.c. La plataforma debe ser capaz de aprovisionar de manera básica los DT.**

En este caso, al igual que en el anterior, los ficheros VNFD se encargan de cubrir la configuración inicial a través de las acciones de tipo Day-1, así como a través del uso de ficheros cloud-init. En relación con la instancia de tipo core5G, el sistema toma una imagen precargada en el VIM el cual contiene el core ya instalado.

- **R1.f. La plataforma debe permitir la obtención de datos sobre el estado de ambos gemelos, así como del tráfico cursado en cada entorno.**

En este caso específico, se hace uso de acciones de tipo Day-2 las cuales ejecutan el software ntop capaz de obtener capturas del tráfico cursado en los entornos, así como ofrecer información adicional sobre flujos, conexiones, etc.

- **R2.a. La plataforma debe hacer uso de tecnologías habilitadoras capaces de desplegar una red física de manera virtualizada**

Al igual que el caso previo, se hace uso de la tecnología NFV utilizando OSM como componente MANO y OpenStack como VIM.

- **R2.b. La plataforma debe permitir la obtención de la información mínima necesaria para llevar a cabo el modelado, independientemente de la topología presente en la red física**

En este caso, el componente de la plataforma que permite esta obtención es el sistema DANA. Quedando demostrado, junto con el caso anterior, que el sistema es capaz de recolectar la información necesaria para el despliegue independientemente de la tipología de los equipos desplegados en el gemelo físico.

- **R2.c. La plataforma debe permitir la replicabilidad de los NDT a través de los descriptores NFV asociados a cada escenario.**

Al igual que el requisito R1.b, el cual permite realizar la gestión del ciclo de vida del NDT, la herramienta OSM permite la replicabilidad y despliegue múltiple del NDT generado. Pudiendo desplegar varios NDT de manera concurrente.

- **R2.d. La plataforma debe cubrir la gestión del NDT, permitiendo el despliegue, así como la eliminación de los escenarios.**

De igual manera que el requisito anterior, OSM permite gestionar completamente los despliegues de NDT, permitiendo pausar los despliegues, eliminarlos o volver a desplegarlos desde su fase inicial.

- **R2.e. La plataforma debe ser independiente al tipo de red o caso de estudio**

Tomando los casos de estudio I y II, se demuestra que la plataforma es independiente de la red objetivo, ya que es el sistema DANA el que se encarga de adaptarse al tipo de red durante las fases de recolección de datos.

- **R3.b. La plataforma debe permitir la diferenciación de flujos de comunicación**

En este caso, se observa a través de la **Figura 51** y **Figura 55** que una vez ejecuta la acción de tipo Day-2 es posible acceder al software ntop el cual se encarga de obtener los ficheros de tráfico en formato pcap, separando la captura de tráfico del canal, basado en MQTT, utilizado para el intercambio de datos entre los gemelos.

- **R5.a. La plataforma debe poder ser operada de manera semi automática.**

Con la presentación de este caso de estudio, se pone de manifiesto que el grueso de la operación es capaz de realizarse de manera automática, con la salvedad, definida por diseño, de ejecutar las acciones de recolección de tráfico manualmente a través de la ejecución de la acción de tipo Day-2 desde la interfaz de OSM.

5.3. Caso III. Interconexión entre gemelos y securización del canal de comunicaciones

Este tercer caso de estudio parte de un ecosistema NDT configurado y desplegado, centrándose en la implementación del canal de comunicación bidireccional, característica fundamental del paradigma de gemelos digitales, que permite el intercambio de información para una coevolución conjunta de los entornos físico y digital.

Este caso III tiene como objetivo la implementación de las fases VI, sección 3.1.6 y VII, sección 3.1.7, de la metodología propuesta en el capítulo tres. Tal y como se describió en la sección 4.1.2, el ecosistema diseñado se basa en el protocolo MQTT para realizar el intercambio de información entre los distintos entornos. Pese a que, a alto nivel, la literatura de apoyo utilizada para el planteamiento tanto de la metodología como de la plataforma habla de un único canal bidireccional que comunica el gemelo físico con el gemelo digital, en esta implementación, según lo expuesto en el capítulo cuatro, se ha optado por dividir ese canal en cuatro flujos de comunicación reflejados en la **Tabla 17**.

Al igual que lo expuesto en los casos anteriores, como se observa en la **Figura 56**, el sistema de gestión inicia la tarea de recolección efectuando una publicación en el topic de gestión MGMT_AGENT_STATUS, el cual es recibido por el conector MQTT presente en el agente desplegado en el entorno físico. Esta acción inicia la tarea de recolección de datos para caracterización topológica del escenario real, enviando al topic TOPOLOGY el modelo de datos correspondiente. En el caso del agente inteligente presente en el gemelo físico, una vez obtenido el fichero JSON que define la topología, conoce los diferentes elementos presentes en la red. Los identificadores asociados a esos elementos se utilizan para publicar, información de comportamiento una vez cambie el estado del agente, en sus topics específicos con la nomenclatura BEHA_PT2MANO/SNMP/NET_ELE_XXX para los equipos sondeables únicamente a través de SNMP y en el topic BEHA_PT2MANO/AGENT/NET_ELE_XXX para los equipos sondeables a través de agentes satélite.

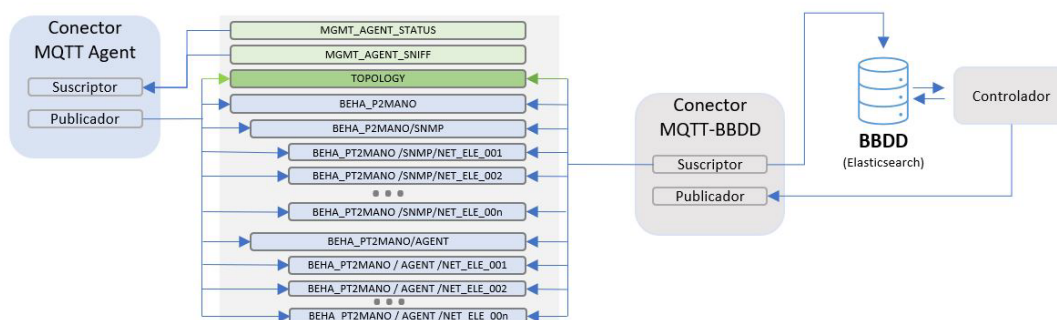


Figura 56. Detalle de la auto suscripción a los topics MQTT de comportamiento por parte del sistema de gestión

Para el correcto funcionamiento del canal de comunicación que conecta el gemelo físico con el gemelo digital, el agente inteligente desplegado en el NDT necesita conocer los topics MQTT a los cuales debe suscribirse para recibir la información de comportamiento tras la fase de despliegue del NDT. Esta tarea recae sobre el sistema de gestión, ya que es el que recibe el fichero JSON que describe la topología, tomando los identificadores de elemento de red, para poder ser utilizados por el suscriptor MQTT presente en el agente inteligente del entorno virtual.

De esta manera, tal y como se refleja de manera esquematizada en la **Figura 57**, tras el despliegue, los elementos MQTT presentes en los entornos físico y digital tienen la información necesaria en referencia a los topics a utilizar para intercomunicar todo el ecosistema. En este punto, únicamente cambia el prefijo asignado al topic, pasando de “BEHA_PT2MANO/SNMP o BEHA_PT2MANO/AGENT” a “BEHA_MANO2DT/SNMP o BEHA_MANO2DT”.

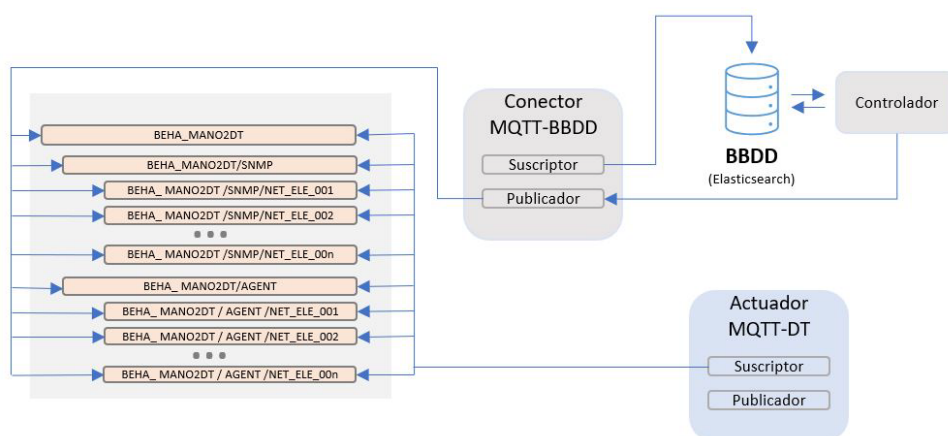


Figura 57. Detalle de la auto suscripción a los topics MQTT de comportamiento por parte del agente inteligente del NDT

A continuación, se muestra el establecimiento del sistema completo, haciendo énfasis en la información almacenada y controlada desde el sistema de gestión. Este sistema, como se ha explicado en el capítulo cuatro, alberga tanto el broker de mensajería MQTT, la base de datos elasticsearch, así como el controlador encargado de gestionar todo el ecosistema.

Una vez los datos correspondientes al modelo JSON son generados y enviados por el agente inteligente desplegado en el gemelo físico, el sistema de gestión se encarga de recibir el fichero y almacenarlo en la base de datos elasticsearch con el objetivo de mantener un histórico en referencia a la topología física.

En la **Figura 58** se muestra el detalle, a través del visualizar Kibana conectado a Elasticsearch, de la información topológica correspondiente al caso I el cual está formado por equipos hardware de propósito específico sondeados a través del protocolo SNMP. En ella se puede observar que el flag_status está asignado a cero, indicando que el agente se encuentra en su fase inicial, en la cual se encarga de obtener información para la caracterización topológica del escenario.

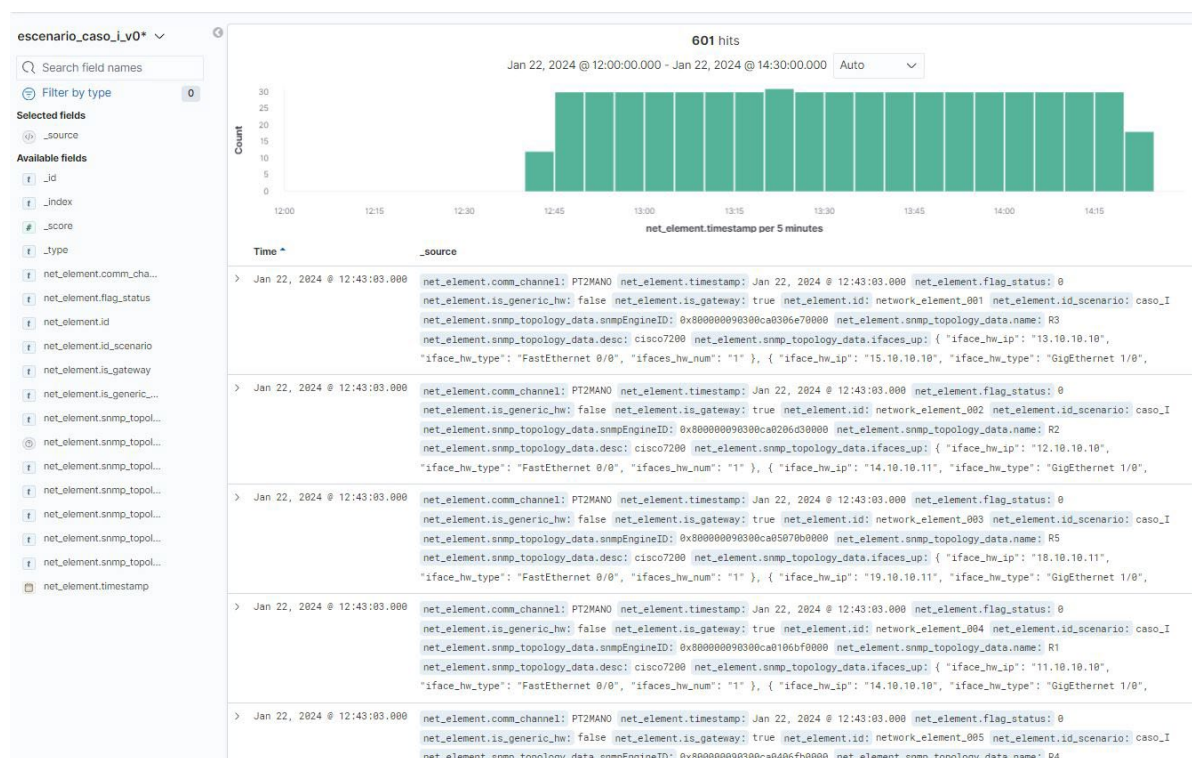


Figura 58. Visual sistema de gestión para la información topológica reportada por el escenario del caso I

Desde la interfaz Kibana, es posible realizar filtrados en base a campos específicos o franjas temporales concretas con el fin de obtener información útil para entender y gestionar correctamente el ecosistema NDT desplegado. En la **Figura 59** se muestra la información asociada al flujo de datos enviado desde el gemelo físico hacia el sistema de gestión, *net_element.comm_channel* correspondiente al valor PT2MANO.

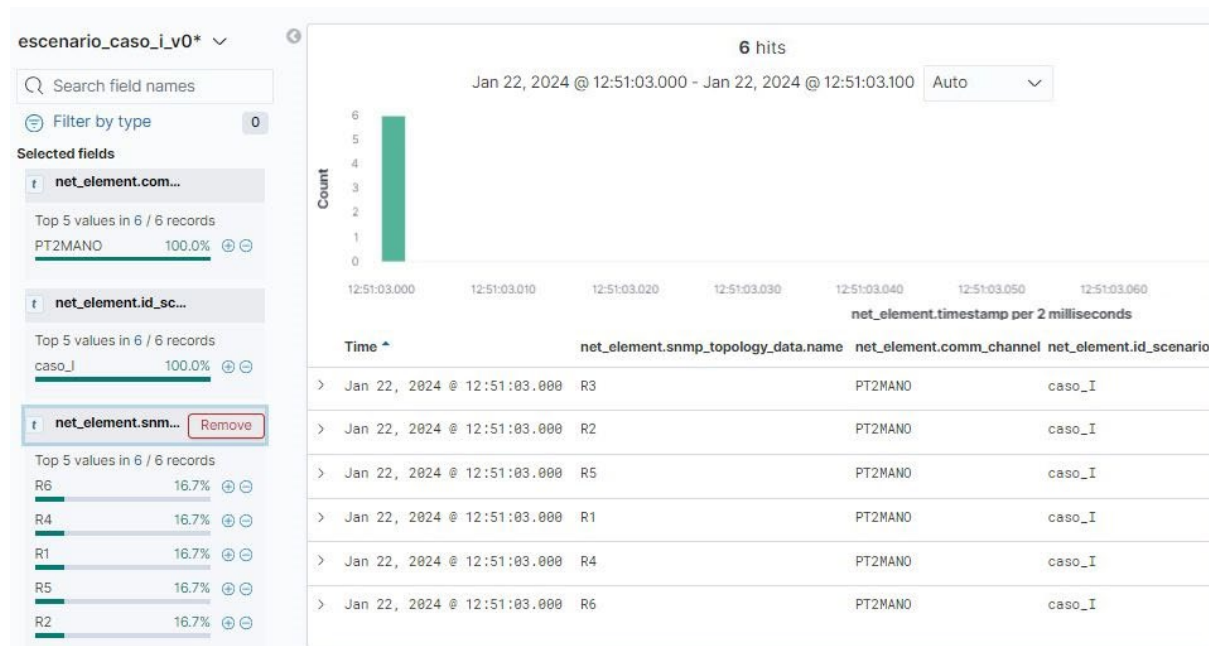


Figura 59. Filtrado de datos topológicos reportados por el caso I

En este caso, la captura muestra los elementos de red correspondientes (routers R1 – R6) asignados al *net_element.id_scenario* caso_I. Pese a que el sistema de gestión toma el fichero JSON directamente desde el topic *TOPOLOGY* para su posterior procesamiento por parte del subsistema de adaptación para la generación de los ficheros VNFD y NSD en formato NFV-SOL006, esta información es almacenada en la base de datos para futuros usos o consultas. Como se muestra en la **Figura 60**, es posible recuperar el JSON generado por el agente inteligente desplegado en el gemelo físico para su posterior uso, modificación o consulta.

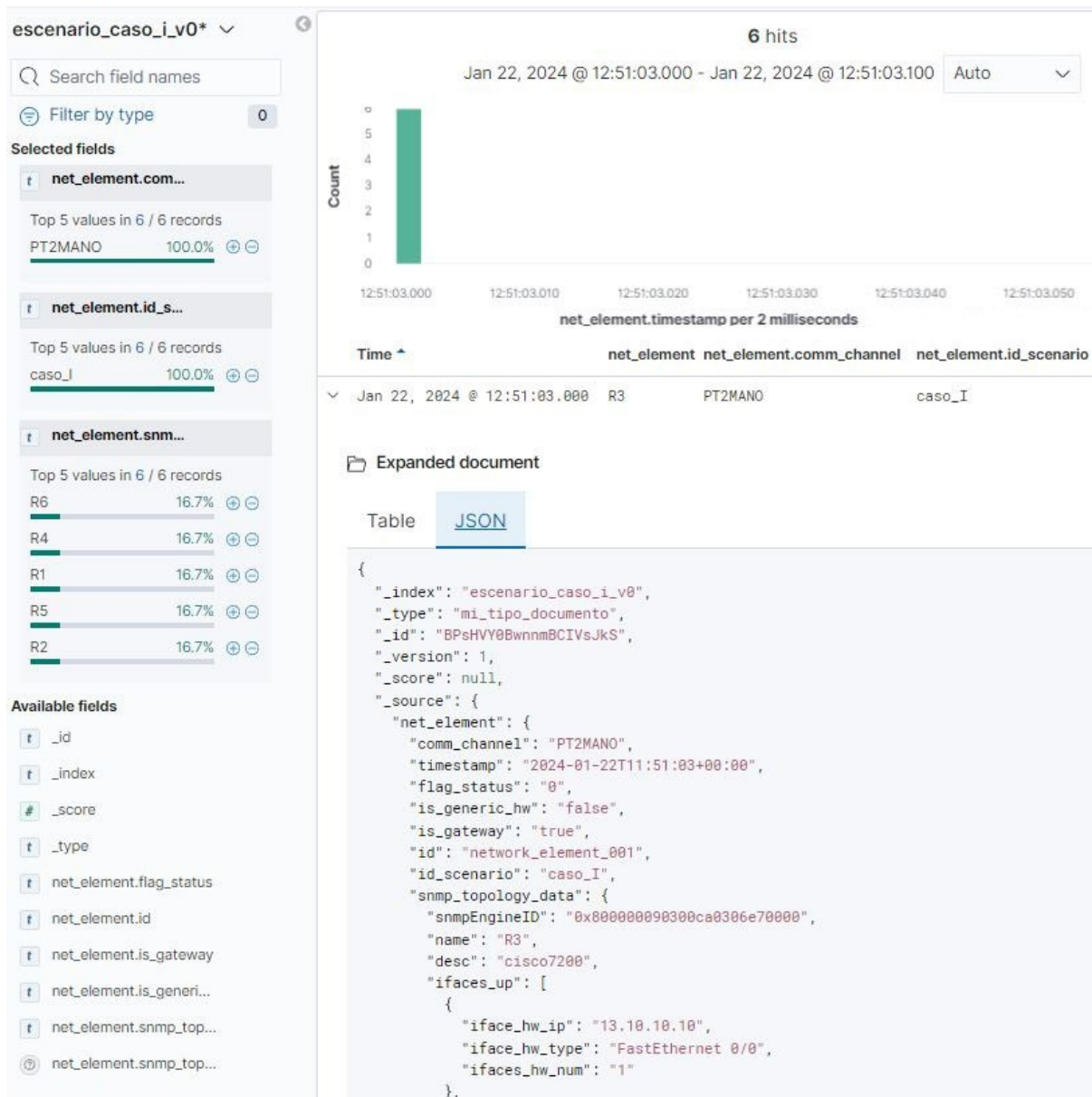


Figura 60. Modelo de datos JSON almacenado para el caso I

De igual manera, la **Figura 61**, **Figura 62** y **Figura 63** refleja la información generada por el agente inteligente en referencia a la topología desplegada como gemelo físico para el caso II. En este caso, la información topológica refleja equipos hardware de propósito general, los cuales hacen uso de software de virtualización como VirtualBox, Docker o Kubernetes. Pudiendo filtrar, como en el caso anterior, desde el visualizador Kibana alojado en el sistema de gestión, para obtener datos específicos ligados a la topología como el *net_element.id*, el *net_element.comm_channel* o el *timestamp* asociado a los datos.

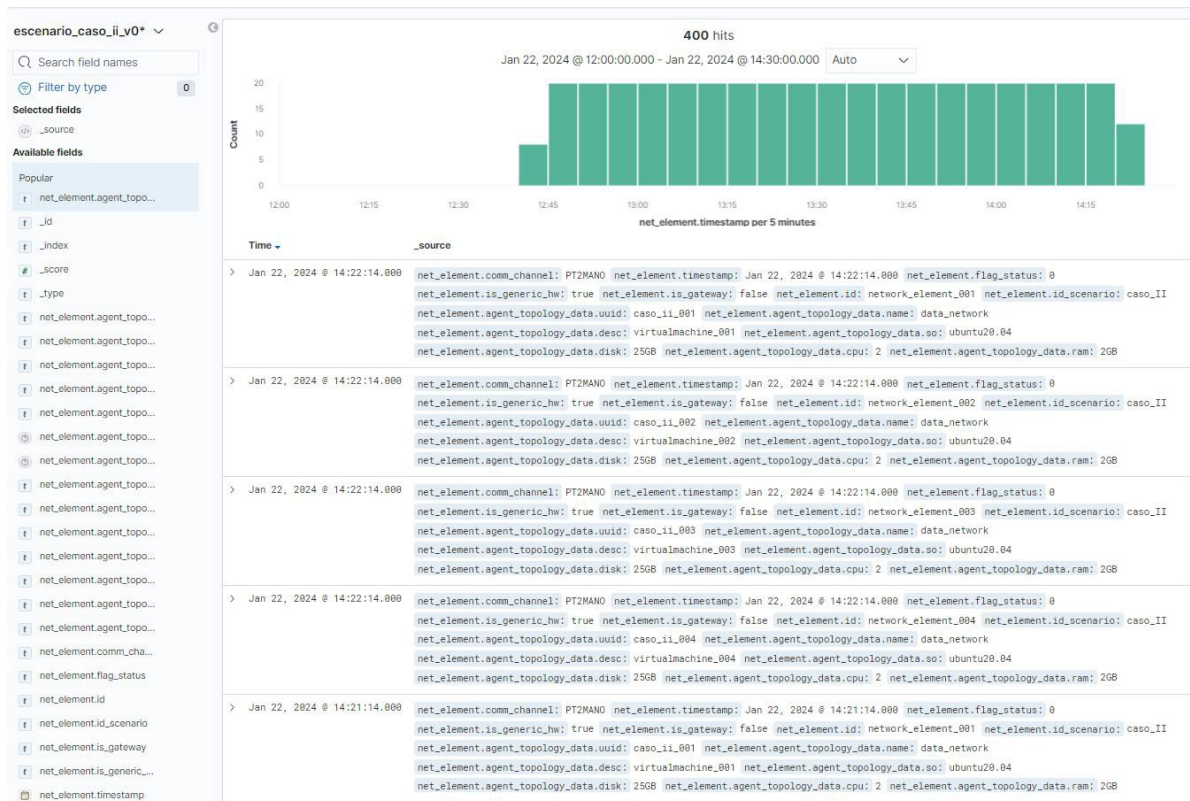


Figura 61. Visual sistema de gestión para la información topológica reportada por el escenario del caso II

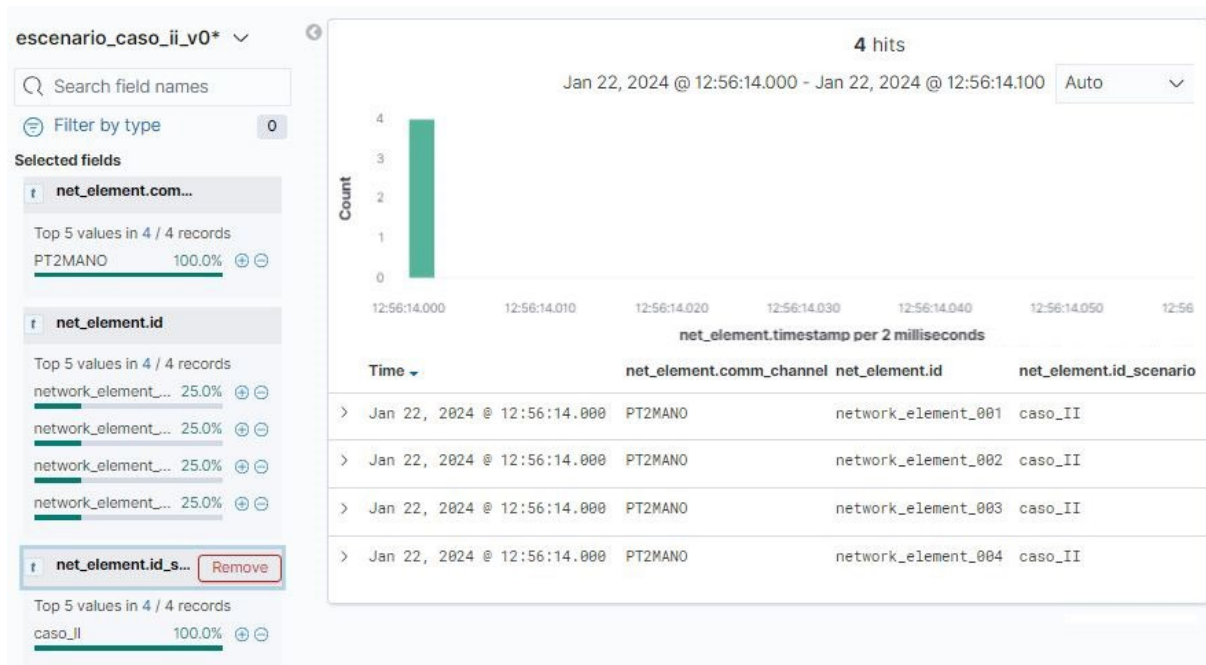


Figura 62. Filtrado de datos topológicos reportados por el caso II

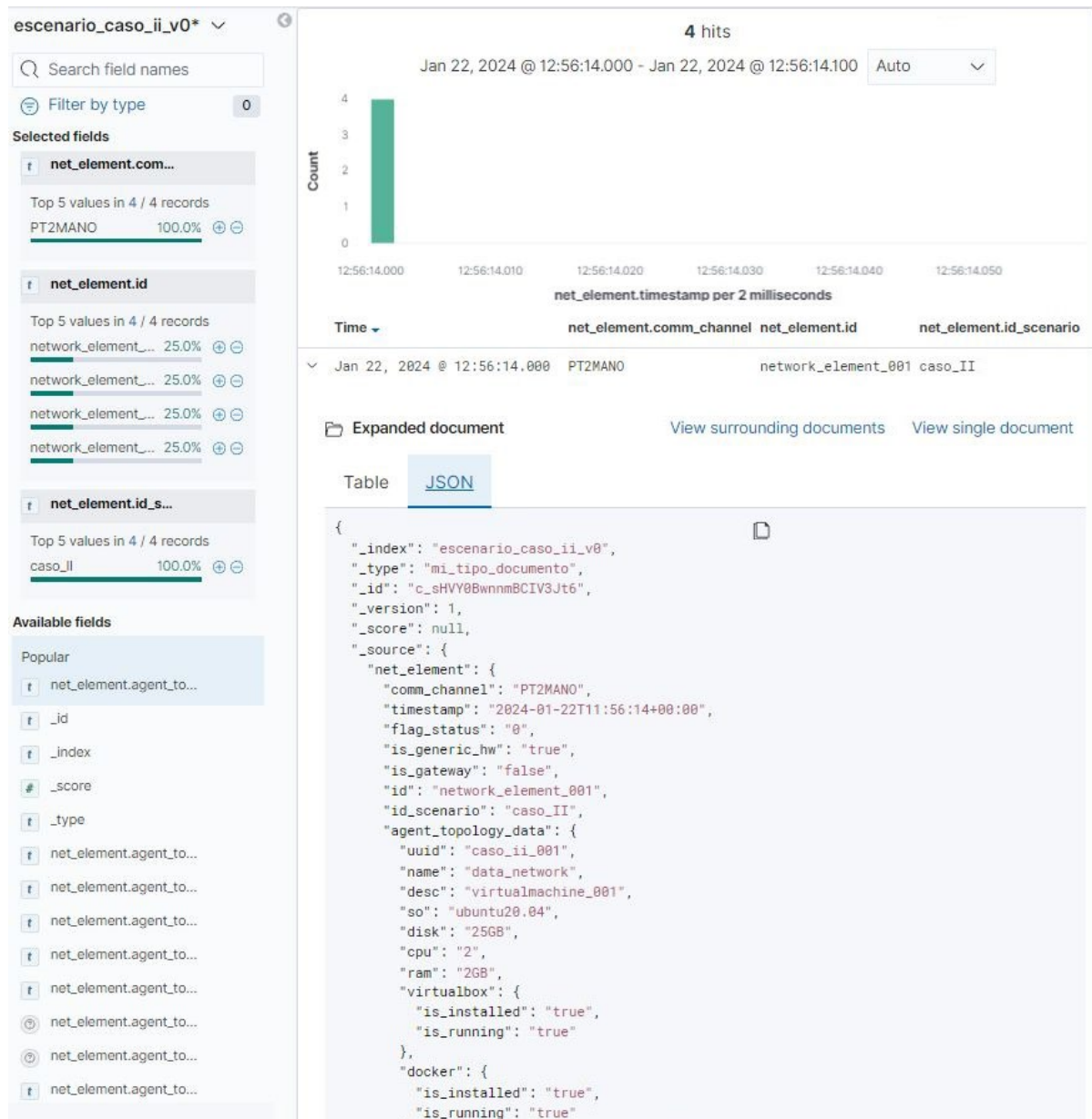


Figura 63. Modelo de datos JSON almacenado para el caso I

Tras la fase de despliegue del NDT, el sistema de gestión publica el flag sobre el topic `MGMT_AGENT_STATUS` que permite el cambio de comportamiento de los agentes inteligentes desplegados tanto en el gemelo físico como en el gemelo digital, pasando al estado 1 el cual activa los módulos para la recolección de información basada en el comportamiento de los elementos de red.

Llegado a este punto, el agente inteligente hace uso del modelo JSON de datos, reflejando los datos establecidos para el bloque *behavior* de cada uno de los elementos de red que conforman el escenario del gemelo físico. A diferencia del envío de datos de topología, en

el cual se enviaba el JSON completo, en este punto el agente inteligente toma cada JSON dentro del JSON global y envía al topic específico siguiendo la etiqueta “BEHA_PT2MANO/AGENT/NET_ELE_XXX”, permitiendo así multiplexar los canales de comunicación PT2MANO/MANO2DT/DT2MANO/MANO2PT para diferenciar los n flujos en base a los n elementos de red presentes en los escenarios.

A diferencia de las figuras previas, donde a través de Kibana se mostraba la información referente a los datos topológicos, la **Figura 64** muestra los datos referentes al comportamiento de los elementos de red presentes en el caso II. Utilizando para ello la estructura de dato manejada por elasticsearch donde se puede apreciar que el *net_element.flag_status* ha pasado a valer 1, indicando la tarea de recolección de datos de comportamiento, y se visualiza la estructura *net.element.agent_behavior_data* que contiene los datos asociados al comportamiento.

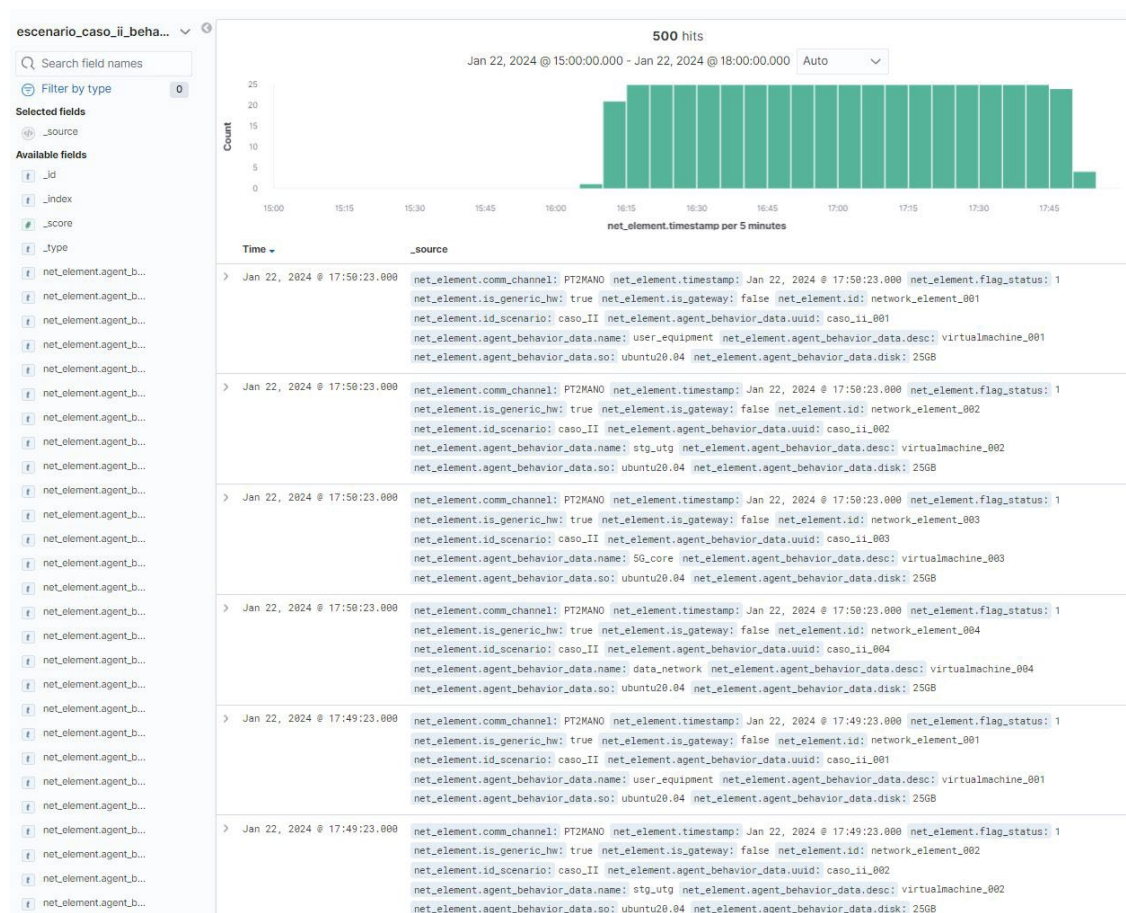


Figura 64. Visual sistema de gestión para la información de comportamiento reportada por el escenario del caso II

A través de diferentes consultas lanzadas a Elasticsearch es posible recuperar el detalle de cada uno de los datos recogidos por el agente inteligente. En la **Figura 65** se muestra el detalle parcial de la estructura de datos en la cual se reflejan datos como el identificador de contenedor Docker desplegado en el elemento de la red física, así como parámetros como su estado, versión, consumo de recursos etc.

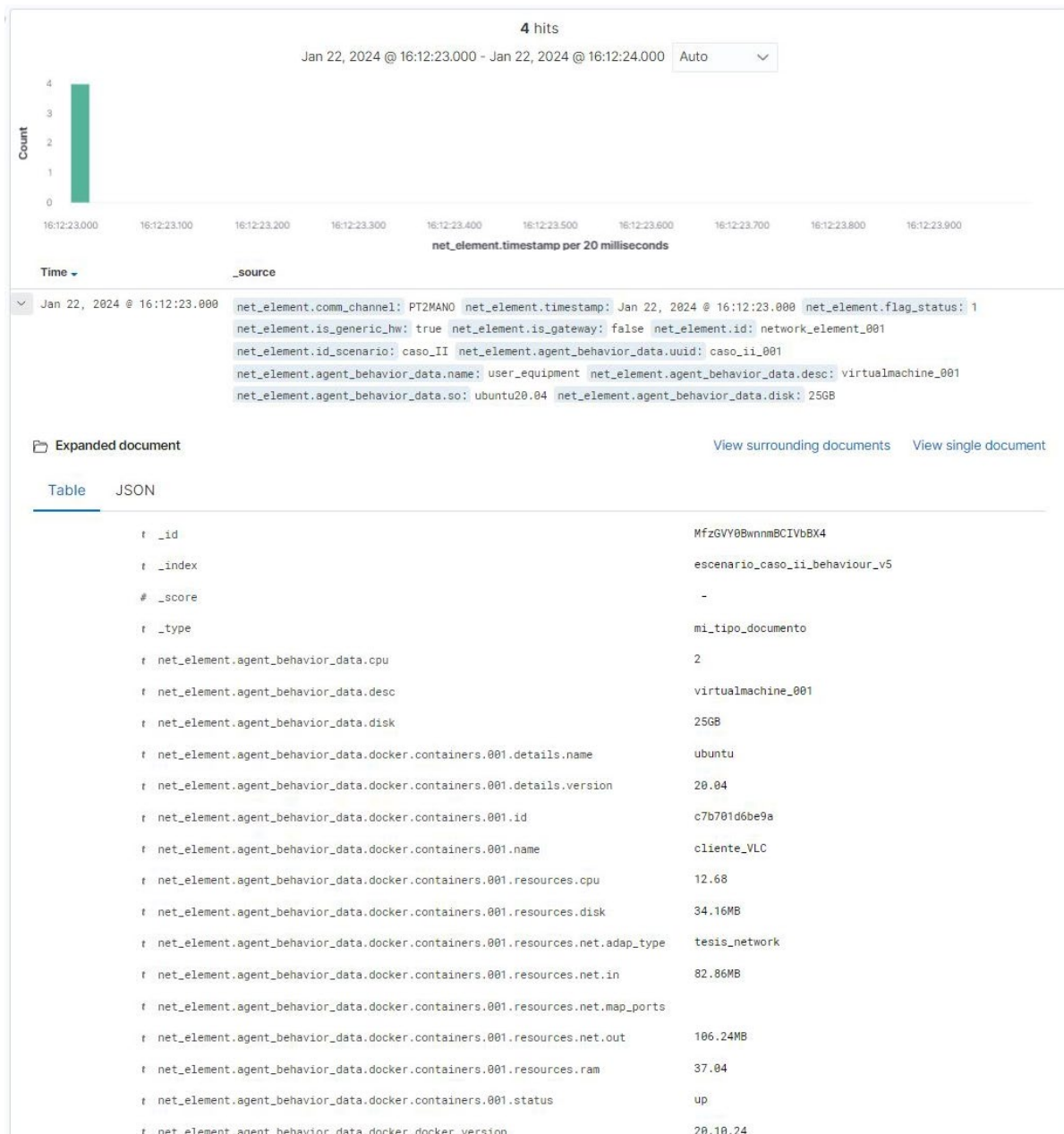


Figura 65. Filtrado de datos de comportamiento reportados por el caso II

Al igual que los datos relacionados con la topología, los datos en referencia al comportamiento son almacenados en Elasticsearch para poder ser utilizados posteriormente en un procesamiento externo al NDT. El procesamiento de la información

almacenada, haciendo uso de algoritmos avanzados, permitiría identificar patrones anómalos en los datos que permitan predecir el estado de la red. Adicionalmente, este almacenamiento puede ser usado como fuente de monitorización de manera que el sistema de gestión actuaría como un NMS (*Network Management System*) avanzado capaz de reflejar el estado de todos los elementos que conforman el ecosistema NDT. Un ejemplo de la evolución temporal ligada a parámetros concretos como el uso de CPU se observa en la **Figura 66**, donde se muestra el consumo de CPU asociado a funciones de red concretas como son el UPF y el SMF del core de red 5G.

Adicionalmente a este potencial uso de los datos almacenados, el elemento controlador alojado en el sistema de gestión es el encargado de condicionar la reinyección de este tipo de datos hacia el entorno del gemelo digital de forma que el agente inteligente desplegando en el NDT trabaje como actuador sobre los diferentes elementos del NDT. Esta característica es potencialmente útil cuando se trata de equipos hardware de propósito general, los cuales típicamente ejecutan software alojado en contenedores, de forma que una vez establecido el sistema, ante un cambio de versión de contenedor en el gemelo físico, o la inclusión de un nuevo software o funcionalidad es capaz de propagar el cambio hacía el NDT, procesando ese cambio por el agente inteligente y actuando directamente sobre el NDT para coevolucionar y adaptarse al gemelo físico.

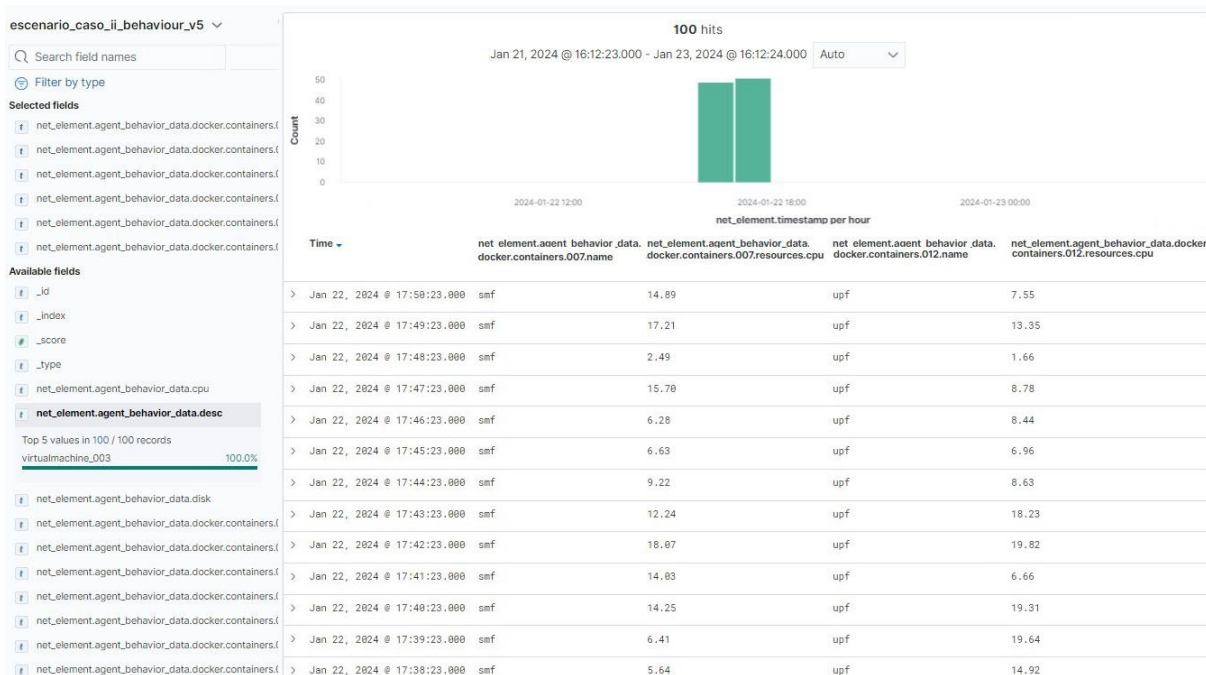


Figura 66. Consulta de la evolución en el consumo de recursos de NF específicas del caso II

Con el objetivo de dotar de seguridad al canal de comunicación para garantizar la confidencialidad, integridad y autenticidad de los datos se implementa el soporte a TLS en los elementos MQTT del sistema, clientes presentes en sistema DANA y conector, así como en el propio broker MQTT alojado en el sistema de gestión.

En primer lugar se generan los certificados y claves necesarios en el sistema de gestión haciendo uso de la herramienta OpenSSL [168]. Para ello se genera una autoridad de certificación (CA) encargada de firmar los certificados tanto para el broker MQTT como para los clientes. Una vez generada la CA, se procede a crear el certificado y la clave para el broker MQTT, haciendo uso de RSA-2048 para posteriormente ser firmado con el certificado CA.

Tras este primer paso, se obtienen los siguientes ficheros:

- ca.crt
- server.crt
- server.key

Los cuales se almacenan en el sistema de gestión, y serán accesibles a través del fichero de configuración *mosquitto.conf*, tal y como se muestra a continuación en el ejemplo:

```
per_listener_settings true

# Restringimos las conexiones inseguras a la red local
listener 1883 127.0.0.1

# Habilitar el cifrado TLS
listener 8883 0.0.0.0

allow_zero_length_clientid false
allow_anonymous false
persistent_client_expiration 1h
use_username_as_clientid false
use_identity_as_username true

require_certificate true
cafile /home/dev/ndt_tesis/certs/ca.crt
certfile /home/ dev /ndt_tesis/certs/server.crt
keyfile /home/ dev /ndt_tesis/certs/server.key
tls_version tlsv1.3
```

En el mismo fichero de configuración se habilita la comunicación securizada a través del puerto 8883 definido para MQTT.

Adicionalmente a la configuración establecida en el broker MQTT, es necesario generar los certificados a utilizar por parte de los clientes MQTT alojados tanto en el sistema DANA como en el conector MQTT presente en el sistema de gestión. Estos certificados de cliente, aparte de establecer la comunicación securizada, evitan que cualquier cliente no autorizado pueda comunicarse con el broker MQTT. Para ello, desde el sistema de gestión se genera un nuevo par clave-certificado, de la misma manera que los generados para el servidor con OpenSSL y se realiza la firma con la CA generada.

Tras este segundo paso se obtienen los siguientes ficheros:

- client.crt
- client.key
- ca.crt

En el caso de los clientes publicadores/suscriptores, los cuales hacen uso del cliente python paho MQTT, implementan la función “*tls_set*” para cargar los certificados y claves como se muestra en el ejemplo:

```
mqttts_client.tls_set( ca_certs = "/home/DANA/certs/ca.crt",  
                       certfile = "/home/DANA/certs/client.crt",  
                       keyfile = "/home/DANA/certs/client.key" )  
mqtt_client.connect ( "Sistema_gestion", port = 8883 )
```

Pese a que TLS, actualmente, proporciona un mecanismo eficiente y robusto para securizar la comunicación MQTT, existe cierta preocupación de que la criptografía tradicional pase a un segundo ante la posible amenaza cuántica. En este contexto, un ataque realizado desde ordenadores cuánticos sería capaz de romper de manera relativamente sencilla la criptografía tradicional, consiguiendo resolver fácilmente la factorización de números primos mediante el algoritmo de Shor [169]. Pese a que la presente tesis doctoral no se centra en resolver esta problemática, si es importante destacar las capacidades del uso de criptografía postcuántica sobre MQTT a través de librerías como *liboqs* [112] la cual implementa una serie de borradores actualmente evaluados por el NIST [170] en relación a mecanismos de encapsulación de clave (KEM) y esquemas de firma como los que se muestran a continuación:

- Mecanismos de encapsulación de clave:
 - **BIKE**: BIKE-L1, BIKE-L3, BIKE-L5
 - **Classic McEliece**: Classic-McEliece-348864, Classic-McEliece-348864f, Classic-McEliece-460896, Classic-McEliece-460896f, Classic-McEliece-6688128, Classic-McEliece-6688128f, Classic-McEliece-6960119, Classic-McEliece-6960119f, Classic-McEliece-8192128, Classic-McEliece-8192128f
 - **FrodoKEM**: FrodoKEM-640-AES, FrodoKEM-640-SHAKE, FrodoKEM-976-AES, FrodoKEM-976-SHAKE, FrodoKEM-1344-AES, FrodoKEM-1344-SHAKE
 - **HQC**: HQC-128, HQC-192, HQC-256
 - **Kyber**: Kyber512, Kyber768, Kyber1024
 - **ML-KEM**: ML-KEM-512-ipd (alias: ML-KEM-512), ML-KEM-768-ipd (alias: ML-KEM-768), ML-KEM-1024-ipd (alias: ML-KEM-1024)
 - **NTRU-Prime**: sntrup761
- Esquemas de firma:
 - **CRYSTALS-Dilithium**: Dilithium2, Dilithium3, Dilithium5
 - **Falcon**: Falcon-512, Falcon-1024, Falcon-padded-512, Falcon-padded-1024
 - **ML-DSA**: ML-DSA-44-ipd (alias: ML-DSA-44), ML-DSA-65-ipd (alias: ML-DSA-65), ML-DSA-87-ipd (alias: ML-DSA-87)
 - **SPHINCS+-SHA2**: SPHINCS+-SHA2-128f-simple, SPHINCS+-SHA2-128s-simple, SPHINCS+-SHA2-192f-simple, SPHINCS+-SHA2-192s-simple, SPHINCS+-SHA2-256f-simple, SPHINCS+-SHA2-256s-simple
 - **SPHINCS+-SHAKE**: SPHINCS+-SHAKE-128f-simple, SPHINCS+-SHAKE-128s-simple, SPHINCS+-SHAKE-192f-simple, SPHINCS+-SHAKE-192s-simple, SPHINCS+-SHAKE-256f-simple, SPHINCS+-SHAKE-256s-simple

5.3.1. Conclusiones

Finalmente, este tercer caso de estudio, partiendo de un ecosistema NDT configurado y desplegado, tiene como objetivo implementar y validar el correcto funcionamiento del canal bidireccional encargado de intercambiar la información entre el gemelo físico y el gemelo digital.

En primer lugar, este caso ha permitido comprobar y validar la correcta implementación de la fase IV de la metodología, así como el funcionamiento del submódulo del sistema DANA encargado de publicar vía MQTT la información referente tanto a topología como a comportamiento. En segundo lugar, en relación con la fase VII de la metodología, este caso ha permitido comprobar y validar la capacidad de adaptación del sistema a través del

canal de gestión establecido en el broker MQTT, el cual permite alternar entre los dos estados de funcionamiento del subsistema DANA, permitiendo alternar el funcionamiento y remodelar la red NDT en caso de detectarse un cambio en la topología del gemelo físico.

Por último, y en relación con la comunicación MQTT, se ha podido comprobar la correcta securización haciendo uso de certificados TLS que permiten tanto la autenticación cliente-broker MQTT como el cifrado de las comunicaciones, así como la posibilidad de implementar algoritmos de criptografía postcuántica sobre MQTT haciendo uso de la librería *liboqs*.

Teniendo en cuenta los requisitos planteados en la sección 4.1.1, este caso de uso valida el cumplimiento de los siguientes requisitos:

- **R1.e. La plataforma debe permitir la comunicación bidireccional entre el entorno del gemelo físico y el entorno del gemelo digital.**

Tal y como se ha presentado en el capítulo cuatro, esta comunicación se realiza a través de MQTT. Implicando el sistema DANA, encargado de establecer los distintos flujos de comunicación en base a los elementos de red presentes en el gemelo físico, así como el sistema de gestión que alberga el broker MQTT junto con el controlador y la base de datos.

- **R1.f. La plataforma debe permitir la obtención de datos sobre el estado de ambos gemelos, así como del tráfico cursado en cada entorno**

A diferencia del caso anterior que se centraba en la validación del canal establecido para la obtención del tráfico cursado, este caso se centra en la validación de la recuperación e intercambio de datos referentes al estado de cada elemento del ecosistema NDT.

- **R3.a. La plataforma debe ser capaz de configurar de manera automática los distintos flujos de comunicación.**

El establecimiento automático de los flujos de comunicación se realiza en dos puntos de la plataforma. En primer lugar, el sistema DANA, durante la fase inicial de adquisición de datos, tiene el conocimiento del número de elementos de red, los cuales tienen asignado un identificador único. Una vez se pase al estado 1, el sistema DANA

publicará los datos correspondientes a cada elemento en un topic específico el cual está conformado por un prefijo común junto al identificador de elemento de red.

De manera dual, el sistema de gestión una vez recibe el fichero JSON generado durante el estado 0 por el sistema DANA, analiza de igual manera el número de elementos presente en el JSON, permitiendo al conector-MQTT presente en el sistema de gestión suscribirse a los topics, que se conforman mediante el prefijo más el identificador, sobre los cuales comenzará el sistema DANA a publicar la información.

- **R4.a. El canal utilizado para la comunicación bidireccional entre gemelos debe estar correctamente securizado.**

Tal y como se muestra en el caso de estudio, los elementos MQTT presentes en la plataforma hacen uso de certificados generados previamente y reflejados en sus ficheros de configuración para garantizar la autenticación de elementos, así como el cifrado de la comunicación entre el sistema DANA y el broker MQTT presente en el sistema de gestión.

- **R4.b. Establecimiento de mecanismos que permitan analizar la información antes de ser propagada hacia cada uno de los entornos del gemelo.**

Según lo expuesto en el capítulo cuatro, por decisión de diseño se pretende evitar que el intercambio de información circule libremente entre los elementos que conforman el ecosistema NDT. Por ello se definen los cuatro subflujos reflejados en la **Tabla 17** y el sistema controlador presente en el sistema de gestión, el cual permite dar paso de forma manual, una vez verificada, a la información intercambiada.

- **R4.c. Los mecanismos de seguridad deben adaptarse a los estándares existentes en la medida de lo posible, permitiendo, establecer mecanismos experimentales capaces de ofrecer una capa superior de seguridad.**

Como mecanismo estándar de seguridad se opta por el uso de certificados X509 para habilitar el uso de TLS/SSL en la plataforma. No obstante, en este caso de estudio se analiza la opción de utilizar la librería liboqs la cual permite el uso de algoritmos de criptografía postcuántica, los cuales pueden dotar al sistema de una seguridad mayor ante amenazas avanzadas.

- **R5.c. La plataforma debe ofrecer la posibilidad de interoperabilidad, tanto para la obtención de nueva información, como para compartir información recolectada en el ecosistema NDT con otros sistemas.**

Debido al diseño modular de la plataforma, así como la definición formal de los modelos de datos a utilizar e intercambiar en ella. Es fácilmente integrable con otras fuentes de información o sistemas que aumenten el funcionamiento del NDT. En relación con la compartición de información, dado que todos los datos se almacenan en la base de datos elasticsearch, y los ficheros pcap en el sistema de ficheros ambos presentes en el sistema de gestión, es fácilmente realizable la compartición con sistemas externos.

Capítulo 6

Conclusiones y líneas futuras

En este capítulo se presenta una visión global del trabajo llevado en la presente Tesis Doctoral. Se abordan las conclusiones más relevantes, utilizando como referencia los objetivos planteados al inicio de la memoria. Finalmente, se presentan las posibles vías de investigación futuras derivadas de los resultados obtenidos en la tesis.

6.1. Análisis de objetivos

La investigación reflejada en la presente tesis doctoral inicia con el objetivo general planteado en la sección 1.2, el cual se basa en la formulación de propuestas que contribuyan al diseño, despliegue e interconexión de gemelos digitales de red aplicados a redes de nueva generación. La motivación detrás de ese objetivo se centra en la hipótesis de que la adopción del paradigma de gemelos digitales sobre redes de comunicación de nueva generación, como 5G o 6G, permitirá crear soluciones novedosas capaces de contribuir en tareas como la optimización de recursos, la gestión de red, o la ciberseguridad. Debido a la amplitud del objetivo principal, abordarlo ha producido diferentes líneas de investigación adicionales, varias de ellas se han desarrollado y validado a lo largo de la tesis, mientras que otras se convierten en candidatas a ser llevadas a cabo de investigaciones futuras. A continuación, se presenta un análisis pormenorizado de cada uno de los objetivos específicos definidos en la sección 1.2

Objetivo 1. Realizar un análisis comprensivo del estado del arte, el cual permita obtener una visión completa en referencia a la unión del paradigma de gemelos digitales unido a la tecnología 5G, centrándose especialmente en el estado actual de los diferentes mecanismos de recolección de información para su posterior modelado y despliegue en forma de gemelo digital, con el objetivo de identificar los retos y limitaciones que plantea la aplicación de tecnología basada en gemelos digitales en este tipo de redes de comunicación de última generación.

Tras el estudio del estado del arte realizado en el capítulo 3, se identifica claramente que la recolección de datos es necesaria para llevar a cabo la implementación del paradigma de gemelos digitales. Dicha necesidad se ve apoyada por el draft del IETF el cual pone de manifiesto los datos como pilar fundamental de un NDT. En relación con las tareas de recolección, la sección 2.2.4 recoge varias líneas actuales que abordan la adquisición de datos en el caso específico de gemelos de red. Se destaca a lo largo de la presente tesis doctoral, la fuerte dependencia de la recolección en función del área de aplicación del paradigma de gemelos digitales. Por ello, se hace una fuerte diferenciación en la recolección en base al tipo de equipo final, destacando equipos de propósito específico, y equipos de propósito general, siendo este uno de los grandes retos a la hora de tratar de diseñar una solución global que permita la recolección de datos independientemente de la red objetivo. Optando para el primer grupo, el sondeo vía

protocolos como SNMP o NetConf y para el segundo se opta por el uso de pequeños agentes instalables que permiten una mayor configuración a la hora de obtener parámetros específicos.

Objetivo 2. Proponer y diseñar principios de implementación y funcionamiento de una metodología que permita obtener un mecanismo estándar para crear e interconectar gemelos digitales de redes. Esta metodología abarca distintas fases, desde la captación de información en los gemelos físicos, hasta la interconexión y realimentación con los gemelos digitales, siguiendo los diferentes modelos de framework propuestos por el IETF, actualmente en formato draft.

Tomando como referencia el estudio del estado del arte, tanto en el paradigma global de gemelos digitales, como en su especificación al caso de uso en redes de comunicación, se identifica la necesidad de establecer una metodología procedimentada que permita cubrir las necesidades para el diseño, desarrollo y despliegue de un NDT. En este punto, la sección 2.2.5 recoge los principales retos que plantea esta tecnología, los cuales sirven de motivación para el diseño y establecimiento de la metodología propuesta en el capítulo 3, la cual cubre el ciclo completo del NDT, desde la fase inicial de recolección de datos, hasta la fase final de realimentación en lazo cerrado. Esta metodología está fuertemente influenciada por el draft publicado por el IETF [ref], el cual trata de establecer las bases de la tecnología NDT.

Objetivo 3. Proponer y diseñar una arquitectura, alineada con el framework NFV establecido por la ETSI, que permita la implementación de la metodología diseñada satisfaciendo los distintos requisitos y requerimientos de las tecnologías implicadas.

Tras la definición de la metodología para el modelado y despliegue de gemelos digitales de red, es necesario realizar una propuesta de diseño de una arquitectura o plataforma capaz de implementar correctamente la metodología propuesta. A través de la definición formal de cada una de las fases en el capítulo 3, se presenta la plataforma para el soporte del NDT en el capítulo 4, la cual hace uso de las tecnologías habilitadoras identificadas previamente, alineándose con el framework NFV propuesto por la ETSI, el cual sirve como marco general para la implementación del NDT basado en elementos puramente virtuales. El diseño de la plataforma identifica claramente tres grandes bloques, los cuales engloban el entorno del gemelo físico, el sistema de gestión y el

entorno del gemelo digital. A lo largo del capítulo 4 se presentan todos y cada uno de los subsistemas encargados de llevar a cabo la implementación de la metodología, teniendo en cuenta, en su fase de diseño los requisitos identificados en la sección 4.2.1.

Objetivo 4. Proponer y diseñar agentes software encargados de la recolección e intercambio de información, así como la definición de un sistema encargado de la gestión e interconexión de los distintos flujos de información presentes en el ecosistema dotando a los mismos de un nivel de seguridad apropiado para la tipología de datos intercambiada entre los gemelos.

Dada la fuerte dependencia del paradigma de gemelos digitales con los datos, el cual se aborda en la fase I de la metodología propuesta. Se presenta el diseño del sistema DANA, el cual se encarga de dos de las tareas más críticas en el contexto del NDT. La primera de estas tareas se centra en de recolección de información, diferenciando entre datos para la representación topológica de la red, así como datos para el modelado del comportamiento de los elementos. A su vez, el sistema es capaz de abordar la recolección de datos, independientemente del tipo de equipo objetivo (equipo hardware de carácter específico o equipo hardware de carácter general). La segunda de estas tareas se centra en la interconexión del gemelo físico con el gemelo digital a través del sistema de gestión. Para ello, el sistema DANA contiene la lógica necesaria para procesar la información recolectada relacionada con la topología, la cual es utilizada para el establecimiento de los diferentes flujos de comunicación basados en el protocolo MQTT que serán utilizados para habilitar el canal bidireccional entre los dos entornos. Adicionalmente a estas dos tareas, el sistema DANA se encarga también de realizar la comunicación contra el sistema de gestión, permitiendo tanto la autenticación del sistema, como la dotación de seguridad en el canal de comunicaciones haciendo uso del protocolo SSL/TLS.

Objetivo 5. Validar las propuestas presentadas en la presente tesis doctoral mediante la implementación de diferentes escenarios virtualizados abordando diversos ámbitos de importancia identificados en el estado del arte como puede ser la ciberseguridad o la gestión de red.

Finalmente, tras el establecimiento de la metodología e implementación de la plataforma de soporte al NDT, el capítulo 5 muestra tres pequeños casos de estudio, con el fin de validar tanto el objetivo principal de la presente tesis doctoral, así como los distintos objetivos específicos.

6.2. Limitaciones

Debido al amplio Stack tecnológico que engloba el paradigma de gemelos digitales enfocado específicamente a redes de comunicación, hay ciertos aspectos y tecnologías que requerirían de una investigación adicional para cubrir todos los aspectos y necesidades concretas, algo que excede el alcance de esta tesis. A continuación, se abordan una serie de limitaciones identificadas a raíz del trabajo de investigación realizado.

Se identifica una fuerte limitación asociada a la heterogeneidad, tanto de tecnologías como de dispositivos presentes en una red de comunicación. Esta heterogeneidad plantea un problema a la hora de utilizar un mecanismo único para la obtención de información, pese a que, en la presente tesis doctoral, se ha optado por el uso del protocolo SNMP, el cual surgió con la motivación de establecer un mecanismo estándar para la monitorización y gestión de equipos heterogéneos, es muy probable que, aplicado a ciertos entornos, esté limitado a la información presente en las MIBs de los equipos. Tratando de cubrir esta problemática identificada se presenta el uso de agentes satélite, los cuales, pese a ser funcionales, requerirían de una investigación adicional para poder alcanzar un desarrollo capaz de cubrir completamente esta limitación asociada a la heterogeneidad.

Otro ejemplo es la limitación ligada a la escalabilidad de los mecanismos de recolección sobre escenarios que manejen volúmenes masivos de tráfico, debido a que no ha podido ser investigado en profundidad sobre escenarios reales que tengan estas características.

Adicionalmente a los ejemplos plantados, la investigación planteada cubre aspectos relacionados con el aprovisionamiento de los gemelos digitales de red, pero no se aborda en profundidad. Un estudio futuro permitiría la identificación de mecanismos avanzados de aprovisionamiento automatizados en base a la información recolectada desde los equipos reales.

Por último, una de las grandes limitaciones identificada tanto en la realización la presente tesis doctoral, como en la literatura científica y en los proyectos de investigación relacionados con esta temática, radica en el grado real de integración de soluciones NDT en arquitecturas reales en producción. Pese a que la tecnología NDT ofrece una serie de ventajas evidentes, su integración en sistemas reales no es trivial, concretamente en el aspecto de actuación sobre el entorno real. De igual manera que existe homogeneidad en los mecanismos de obtención de información de los equipos, no existe en sistemas que

permitan actuar o modificar estos equipos, ya que típicamente suelen utilizar sistemas propietarios para realizar una gestión avanzada.

6.3. Conclusiones

En la presente tesis doctoral se ha presentado un conjunto de propuestas orientadas a la contribución del diseño y despliegue de gemelos digitales de red, las cuales han sido presentadas a lo largo de los capítulos 2, 3, 4 y 5.

La primera parte de la investigación llevada a cabo se centra en un estudio detallado del estado del arte. Iniciando la contextualización del paradigma general de gemelos digitales, identificando los aspectos claves, los distintos esfuerzos en la investigación e implantación, así como la evolución a lo largo de los años motivada por tecnologías habilitadoras que han facilitado su implementación y aplicabilidad da distintos ámbitos.

Una vez revisado el concepto general, se identifica claramente la fuerte dependencia del paradigma en función del caso de aplicabilidad específico, observado que no existe una solución agnóstica al campo de aplicación, tanto a nivel procedimental o metodológico como a nivel infraestructura capaz de llevar a cabo un despliegue basado en DT de manera transparente.

Así mismo, tras esta contextualización, se aborda el concepto de NDT, siendo la aplicabilidad del paradigma DT a redes de comunicación. De igual manera que en el caso general, se realiza analizar las características propias de este campo, los retos que plantean, las tecnologías habilitadoras, así como los distintos esfuerzos reflejados en la literatura científica para tratar de llevar a cabo este tipo de implementaciones. Al igual que en el caso general de DT, se observa una carencia a la hora de establecer mecanismos estándares para el modelado y creación de sistemas basados en NDT, pese a los esfuerzos realizados por organismos como el IETF para tratar de establecer las bases comunes de esta tecnología.

Las soluciones planteadas en literatura científica en realización a NDT, cubren aspectos muy concretos, sin llegar a centrarse completamente en la necesidad de establecer y diseñar mecanismos estándares para la creación y despliegue de gemelos, así como dos

de las características fundamentales de esta tecnología, la recolección de datos y la interconexión entre gemelos físicos y gemelos digitales.

Como resultado dicho estudio sirve como motivación para el establecimiento de las siguientes contribuciones:

Contribución 1. Propuesta de una metodología para el modelado y despliegue de gemelos digitales de red.

La metodología propuesta surge como respuesta a los retos planteados por el paradigma NDT, así como las conclusiones obtenidas a través de la revisión del estado del arte. A través de las siete fases definidas, se cubre todo el ciclo de vida completo de un NDT, iniciando a través de la recolección de información y finalizando en la interconexión o realimentación en bucle cerrado entre el gemelo físico y el gemelo digital.

La definición formal de cada una de las siete fases da respuesta a las necesidades y requisitos identificados para el paradigma NDT. Obteniendo un mecanismo sencillo y procedimental para la implementación técnica del NDT, independientemente de la complejidad o el tiempo de escenario de red objetivo.

Adicionalmente a la definición de cada fase, el capítulo 3, recoge los distintos diseños de los subsistemas encargados de llevar a cabo la implementación técnica de cada fase.

Contribución 2. Propuesta de sistema capaz de llevar a cabo la recolección de los datos necesarios para la creación del NDT.

Esta contribución surge a partir de la necesidad identificada con la primera fase de la metodología propuesta, en relación con la recolección de la información sobre redes de comunicación. Dada la heterogeneidad de dispositivos presentes en una red de comunicación, el sistema DANA pretende ser un sistema escalable, capaz de recolectar información cualquier tipo de red, implementando diferentes mecanismos o protocolos para obtener la información necesaria para el modelado de la red en forma de gemelo digital.

A lo largo de la sección 3.1.1 se presenta el funcionamiento interno del sistema DANA, detallando la arquitectura interna del sistema, organizada a través de los distintos

componentes software desarrollados, con el fin de obtener un sistema capaz de recolectar datos, tanto referentes a topología, como a comportamiento, con independencia del tipo de equipo o tecnología implementada en él.

No obstante, aunque el sistema DANA puede ser visto como una contribución adicional a la creación de gemelos digitales, debido a que puede ser utilizado para diversos propósitos como la prospección tecnológica de red, tareas de autodescubrimiento de equipos, tareas como la monitorización o recolección de datos para otros fines distintos al modelado, reúne una serie de características que lo convierten en una pieza clave del ecosistema NDT. Como ya se ha presentado anteriormente, facilita la tarea de recolección, la cual, debido a su diseño modular, tiene el potencial de seguir creciendo y evolucionar sus capacidades de recolección hacia diferentes tipos de redes o tecnologías de virtualización. Adicionalmente, se posiciona como un sistema capaz de llevar a cabo la implementación de acciones sobre ambos gemelos, ya sea la modificación de parámetros en base al comportamiento de los gemelos, o en base a la decisiones obtenidas a través de algoritmos avanzados basados en Machine Learning o Inteligencia Artificial que impliquen una actuación sobre las entidades finales.

Pese a que dicha característica de actuación es tecnológicamente complicada de implementar, el sistema DANA, debido a su presencia en ambos entornos, se posiciona como un elemento a tener en cuenta para llevar a cabo dichas acciones, o actuar como pasarela capaz de traducirlas hacia aplicaciones específicas encargadas de llevar a cabo esas tareas. Concretamente en este último caso, se pone de manifiesto la problemática de la heterogeneidad de dispositivos en una red, ya que, aunque sea posible la obtención de parámetros a través de protocolos estandarizados como SNMP, las acciones que implican modificaciones sobre los equipos suelen utilizar herramientas propietarias, lo cual dificulta la implementación de dichas acciones de forma global y automatizada.

Contribución 3. Propuesta de una plataforma de soporte encargada del despliegue, gestión y utilización del NDT.

Durante el establecimiento del diseño de la metodología presentada en la contribución 1 surge la necesidad de desarrollar una plataforma capaz de llevarla a cabo.

Esta contribución profundiza en la propuesta de una plataforma tecnológica capaz de llevar a cabo la implementación del paradigma NDT cubriendo las necesidades y requerimientos técnicos derivados, tanto del estudio del arte realizado, como de la metodología diseñada.

Se presenta una plataforma conformada por tres grandes bloques, diferenciando entre entorno de gemelo físico, sistema de gestión y entorno de gemelo digital. La unión de estos tres bloques ofrece mecanismos para la creación, gestión y despliegue del ecosistema NDT, cubriendo las siete fases identificadas en la metodología. Concretamente la plataforma ofrece las capacidades tanto de desplegar, permitiendo acciones como la replicabilidad de experimentos o la gestión completa del ciclo de vida del NDT, como de almacenar todos los datos intercambiados dentro del ecosistema.

Adicionalmente a la implementación técnica de las distintas fases a través de la plataforma, se implementa un sistema controlador desplegado dentro del bloque de gestión, el cual se encarga de controlar los flujos de información intercambiados entre la red física y la red gemela digital. Esta implementación no surge como un requisito propio del paradigma DT, sino que pretende facilitar la gestión integral del sistema, así como evitar el intercambio no supervisado de información entre los extremos del ecosistema. Principalmente tratando de evitar posibles propagaciones de problemas, mal funcionamiento, o configuraciones erróneas de un entorno a otro sin ningún tipo de medida de correctiva o análisis previo.

Contribución 4. Propuesta de canal de comunicación bidireccional integrado en el ecosistema NDT.

Esta contribución surge como respuesta a la requisito específico del paradigma DT, basado en la necesidad de establecer un canal de comunicación bidireccional, lo cual lo diferencia de aproximaciones tradicionales como las simulaciones o emulaciones.

En la sección 4.1.2.2 se detalla la aproximación planteada para la presente tesis doctoral, la cual se implementa tanto en el sistema de gestión como en los sistemas DANA desplegados en los entornos de gemelo físico y digital. Para ello se parte de la propuesta de cuatro flujos globales presentes en el ecosistema NDT, los cuales guardan relación con el controlador expuesto al final de la contribución 3.

Concretamente el canal de comunicación se basa en protocolos de mensajería capaces de ofrecer una gestión eficiente para el intercambio de mensajes a través de flujos independientes, permitiendo desplegar diferentes publicadores / suscriptores o productores / consumidores de información en los módulos encargados de recolectar, procesar, almacenar o enviar datos. De esta manera, es posible llevar a cabo el intercambio de información entre un elemento físico y su contraparte digital, permitiendo adicionalmente utilizar sistemas auxiliares encargados de procesar la información o almacenarla en base de datos con el objetivo de mantener un histórico de la comunicación cursada a través del ecosistema NDT desplegado.

Adicionalmente a las características propias de la tecnología basada en mensajería, el canal de comunicación se ha desarrollado teniendo en cuenta la sensibilidad de la información intercambiada en un NDT o en un DT, ya que, debido a la bidireccionalidad, el gemelo físico, el cual típicamente es un sistema en producción, se conecta para poder enviar la información. Esta conexión implica los riesgos y amenazas típicas de cualquier flujo de información enviado sobre redes IP, de forma que, para poder cumplir con la confidencialidad, autenticidad e integridad de los datos, es necesario que el canal sea capaz de soportar mecanismos que garanticen la seguridad. En este sentido, el canal propuesto implementa TLS, con la posibilidad de integrar algoritmos de criptografía postcuántica como capa adicional, pero no obligatoria, de seguridad.

6.4. Trabajos futuros

Esta sección presenta un conjunto de posibles vías de investigación futuras derivadas del trabajo realizado en la presente Tesis Doctoral. Algunas, debido a su envergadura y limitaciones de tiempo no han podido ser llevadas a cabo completamente durante la investigación, y otras surgen a raíz de las conclusiones obtenidas al final de la Tesis. Concretamente se propone:

- *Validación de las propuestas desarrolladas sobre nuevos escenarios:* Pese a que en el capítulo 5 se presentan una serie de casos de estudio, los cuales han servido para validar el correcto funcionamiento, tanto de la metodología propuesta como los distintos sistemas y subsistemas desarrollados, se plantea la inclusión de los desarrollos sobre una arquitectura real con requisitos diferentes, como escenarios

con topologías de red dinámicas, o escenarios con un alto requerimiento de modificaciones en tiempo real. Este tipo de implementación, ligada a entornos de producción, permitirían además la evaluación de como integrar un ecosistema NDT en las herramientas y procedimientos de monitorización y gestión de red.

- *Implementación de trabajos relacionados con IA y ML como módulo integrado dentro de la plataforma de soporte al NDT:* A lo largo de la presente tesis doctoral, concretamente en el capítulo 3, se pone de manifiesto la importancia de la tecnología basada en inteligencia artificial o machine Learning aplicada a entornos de gemelos digitales. Basando su aplicación en el análisis de los datos generados en el entorno del gemelo digital, con el fin de obtener datasets que permitan entrenar este tipo de algoritmos. Esta característica se aborda en el capítulo 5, concretamente en el caso de estudio II, permitiendo la obtención de datasets a través de la plataforma de soporte al NDT. No obstante, futuras investigaciones y desarrollos podrían extender el ámbito de la plataforma incorporando sistemas basados en AI / ML de manera que las salidas generadas por estos algoritmos pudieran integrarse sobre el sistema de gestión el cual actuaría directamente sobre el gemelo físico.
- *Extender la arquitectura a entornos Edge en el contexto de la futura generación de redes móviles 6G:* esta línea de investigación surge a raíz de la futura necesidad tecnológica presentada en el capítulo 3, concretamente relacionada a la inclusión de la tecnología de gemelos digitales en las primeras fases de definición de la tecnología 6G, tal y como se recoge en el IMT2030. En este caso específico, se propone la descentralización de la plataforma de soporte al NDT, con el objetivo de poder desplegar los distintos gemelos digitales como nodos en la red Edge, persiguiendo la disminución de la latencia, así como una reducción en los tiempos de transmisión de información a través del canal bidireccional que uniría los equipos finales con su NDT. Adicionalmente, el acercamiento del NDT a los equipos físicos finales, permitiría mejorar la eficiencia computacional de los algoritmos basados en IA / ML, alimentado con datos generados desde los distintos NDT,

Bibliografía

- [1] "X.700 : Management framework for Open Systems Interconnection (OSI) for CCITT applications." Accessed: Mar. 22, 2024. [Online]. Available: <https://www.itu.int/rec/T-REC-X.700-199209-I/en>
- [2] "M.3010 : Principles for a telecommunications management network." Accessed: Mar. 22, 2024. [Online]. Available: <https://www.itu.int/rec/T-REC-M.3010-200002-I/en>
- [3] U. Mulligan, "Network Functions Virtualisation (NFV)," ETSI. Accessed: Feb. 21, 2024. [Online]. Available: <https://www.etsi.org/technologies/nfv>
- [4] H. Zafar, "Evolving NFV towards the next decade".
- [5] M. Sanz Rodrigo, D. Rivera, J. I. Moreno, M. Álvarez-Campana, and D. R. López, "Digital Twins for 5G Networks: A Modeling and Deployment Methodology," *IEEE Access*, vol. 11, pp. 38112–38126, 2023, doi: 10.1109/ACCESS.2023.3267548.
- [6] "Gartner Survey Reveals Digital Twins Are Entering Mainstream Use," Gartner. Accessed: Oct. 25, 2023. [Online]. Available: <https://www.gartner.com/en/newsroom/press-releases/2019-02-20-gartner-survey-reveals-digital-twins-are-entering-mainstream-use>
- [7] "Gartner Top Strategic Technology Trends 2023" Accessed: Oct. 25, 2023. [Online]. Available: <https://emtemp.gcom.cloud/ngw/globalassets/en/publications/documents/2023-gartner-top-strategic-technology-trends-ebook.pdf>
- [8] "Digital Twin Market Size & Share | Growth Forecasts, 2032," Global Market Insights Inc. Accessed: Oct. 25, 2023. [Online]. Available: <https://www.gminsights.com/industry-analysis/digital-twin-market>
- [9] "Digital Twin Market Size, Share & Trends [2023 Report]." Accessed: Oct. 25, 2023. [Online]. Available: <https://www.technavio.com/report/digital-twin-market-size-share-trends-2023>
- [10] M. Grieves, "Product lifecycle management: the new paradigm for enterprises," *Int. J. Prod. Dev. - Int J Prod Dev*, vol. 2, Jan. 2005, doi: 10.1504/IJPD.2005.006669.
- [11] K. Främling, J. Holmström, T. Ala-Risku, and M. Kärkkäinen, "Product Agents for handling information about physical objects," *Hels. Univ. Technol.*
- [12] E. Tuegel, A. Ingraffea, T. Eason, and S. Spottswood, "Reengineering Aircraft Structural Life Prediction Using a Digital Twin," *Int. J. Aerosp. Eng.*, vol. 2011, Jan. 2011, doi: 10.1155/2011/154798.
- [13] "Global Horizons Final Report" Accessed: Oct. 25, 2023. [Online]. Available: <https://defenseinnovationmarketplace.dtic.mil/wp-content/uploads/airforce/GlobalHorizonsFINALREPORT6-26-13.pdf>
- [14] M. Grieves and J. Vickers, "Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems," 2017, pp. 85–113. doi: 10.1007/978-3-319-38756-7_4.
- [15] B. Barricelli, E. Casiraghi, and D. Fogli, "A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications," *IEEE Access*, vol. PP, pp. 1–1, Nov. 2019, doi: 10.1109/ACCESS.2019.2953499.
- [16] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital Twin in manufacturing: A categorical literature review and classification," *IFAC-Pap.*, vol. 51, pp. 1016–1022, Jan. 2018, doi: 10.1016/j.ifacol.2018.08.474.

-
- [17] E. Negri, L. Fumagalli, and M. Macchi, "A Review of the Roles of Digital Twin in CPS-based Production Systems," *Procedia Manuf.*, vol. 11, pp. 939–948, Jan. 2017, doi: 10.1016/j.promfg.2017.07.198.
 - [18] Y. Zheng, S. Yang, and H. Cheng, "An application framework of digital twin and its case study," *J. Ambient Intell. Humaniz. Comput.*, vol. 10, no. 3, pp. 1141–1153, Mar. 2019, doi: 10.1007/s12652-018-0911-3.
 - [19] "Definition of a Digital Twin," Digital Twin Consortium. Accessed: Oct. 25, 2023. [Online]. Available: <https://www.digitaltwinconsortium.org/initiatives/the-definition-of-a-digital-twin/>
 - [20] "Digital Transformation in Industry White Paper". Industrial IoT Consortium. Accessed: Oct. 30, 2023 [Online]. Available: https://www.iiconsortium.org/pdf/Digital_Transformation_in_Industry_Whitepaper_2020-07-23.pdf
 - [21] "About the Centre for Digital Built Britain." University of Cambridge. Accessed: Oct. 25, 2023. [Online]. Available: <https://www.cdabb.cam.ac.uk/AboutCDBB>
 - [22] "Digital Twin | Siemens," Siemens Digital Industries Software. Accessed: Oct. 25, 2023. [Online]. Available: <https://www.plm.automation.siemens.com/global/en/our-story/glossary/digital-twin/24465>
 - [23] "Digital Twin Software | GE Digital." Accessed: Feb. 21, 2024. [Online]. Available: <https://www.ge.com/digital/applications/digital-twin>
 - [24] R. Söderberg, K. Wärmefjord, J. S. Carlson, and L. Lindkvist, "Toward a Digital Twin for real-time geometry assurance in individualized production," *CIRP Ann.*, vol. 66, no. 1, pp. 137–140, Jan. 2017, doi: 10.1016/j.cirp.2017.04.038.
 - [25] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *Int. J. Adv. Manuf. Technol.*, vol. 94, no. 9, pp. 3563–3576, Feb. 2018, doi: 10.1007/s00170-017-0233-1.
 - [26] Z. Liu, N. Meyendorf, and N. Mrad, "The role of data fusion in predictive maintenance using digital twin," *AIP Conf. Proc.*, vol. 1949, no. 1, p. 020023, Apr. 2018, doi: 10.1063/1.5031520.
 - [27] S. Boschert and R. Rosen, "Digital Twin—The Simulation Aspect," 2016, pp. 59–74. doi: 10.1007/978-3-319-32156-1_5.
 - [28] M. Segovia and J. Garcia-Alfaro, "Design, Modeling and Implementation of Digital Twins," *Sensors*, vol. 22, no. 14, Art. no. 14, Jan. 2022, doi: 10.3390/s22145396.
 - [29] E. Ferko, A. Bucaioni, and M. Behnam, "Architecting Digital Twins," *IEEE Access*, vol. 10, pp. 50335–50350, 2022, doi: 10.1109/ACCESS.2022.3172964.
 - [30] M. Grieves, "Digital Twin: Manufacturing Excellence through Virtual Factory Replication," Mar. 2015.
 - [31] F. Tao, M. Zhang, Y. Liu, and A. Nee, "Digital twin driven prognostics and health management for complex equipment," *CIRP Ann.*, vol. 67, May 2018, doi: 10.1016/j.cirp.2018.04.055.
 - [32] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, "Characterising the Digital Twin: A systematic literature review," *CIRP J. Manuf. Sci. Technol.*, vol. 29, pp. 36–52, May 2020, doi: 10.1016/j.cirpj.2020.02.002.
 - [33] "Digital Twins Made Easy | AWS IoT TwinMaker | Amazon Web Services," Amazon Web Services, Inc. Accessed: Oct. 25, 2023. [Online]. Available: <https://aws.amazon.com/es/iot-twinmaker/>

-
- [34] “Digital Twins: modelado y simulaciones | Microsoft Azure.” Accessed: Oct. 25, 2023. [Online]. Available: <https://azure.microsoft.com/es-es/products/digital-twins>
 - [35] “Digital Supply Chain | Google Cloud.” Accessed: Oct. 25, 2023. [Online]. Available: <https://cloud.google.com/solutions/supply-chain-twin>
 - [36] “IoT digital twin platform - BaseN.” Accessed: Feb. 21, 2024. [Online]. Available: <https://www.basen.net/>
 - [37] “Akila.” Accessed: Feb. 21, 2024. [Online]. Available: <https://www.akila3d.com/>
 - [38] “Digital Twin Simulation-Based Software | Ansys.” Accessed: Feb. 21, 2024. [Online]. Available: <https://www.ansys.com/products/digital-twin>
 - [39] “Digital twin – AVEVA.” Accessed: Feb. 21, 2024. [Online]. Available: <https://www.aveva.com/en/solutions/digital-transformation/digital-twin/>
 - [40] “IBM Documentation.” Accessed: Feb. 21, 2024. [Online]. Available: <https://www.ibm.com/docs/en/announcements/www.ibm.com/docs/en/announcements/digital-twin-exchange>
 - [41] O. Corporation, “About the IoT Digital Twin Framework,” Oracle Help Center. Accessed: Feb. 21, 2024. [Online]. Available: <https://docs.oracle.com/en/cloud/paas/iot-cloud/iotgs/iot-digital-twin-framework.html>
 - [42] M. Attaran and B. G. Celik, “Digital Twin: Benefits, use cases, challenges, and opportunities,” *Decis. Anal. J.*, vol. 6, p. 100165, Mar. 2023, doi: 10.1016/j.dajour.2023.100165.
 - [43] “Digital Twin: Enabling Technologies, Challenges and Open Research | IEEE Journals & Magazine | IEEE Xplore.” Accessed: Feb. 21, 2024. [Online]. Available: <https://ieeexplore.ieee.org/document/9103025>
 - [44] Q. Qi *et al.*, “Enabling technologies and tools for digital twin,” *J. Manuf. Syst.*, vol. 58, pp. 3–21, Jan. 2021, doi: 10.1016/j.jmsy.2019.10.001.
 - [45] L. Li, B. Lei, and C. Mao, “Digital twin in smart manufacturing,” *J. Ind. Inf. Integr.*, vol. 26, p. 100289, Mar. 2022, doi: 10.1016/j.jii.2021.100289.
 - [46] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, “Digital Twin in Industry: State-of-the-Art,” *IEEE Trans. Ind. Inform.*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019, doi: 10.1109/TII.2018.2873186.
 - [47] T. Erol, A. F. Mendi, and D. Doğan, “The Digital Twin Revolution in Healthcare,” in *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Oct. 2020, pp. 1–7. doi: 10.1109/ISMSIT50672.2020.9255249.
 - [48] T. Bergs, S. Gierlings, T. Auerbach, A. Klink, D. Schraknepper, and T. Augspurger, “The Concept of Digital Twin and Digital Shadow in Manufacturing,” *Procedia CIRP*, vol. 101, pp. 81–84, Jan. 2021, doi: 10.1016/j.procir.2021.02.010.
 - [49] H. Elayan, M. Aloqaily, and M. Guizani, “Digital Twin for Intelligent Context-Aware IoT Healthcare Systems,” *IEEE Internet Things J.*, vol. 8, no. 23, pp. 16749–16757, Dec. 2021, doi: 10.1109/JIOT.2021.3051158.
 - [50] M. Jacoby and T. Usländer, “Digital Twin and Internet of Things—Current Standards Landscape,” *Appl. Sci.*, vol. 10, no. 18, Art. no. 18, Jan. 2020, doi: 10.3390/app10186519.
 - [51] A. Kychkin and A. Nikolaev, “IoT-based Mine Ventilation Control System Architecture with Digital Twin,” in *2020 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*, May 2020, pp. 1–5. doi: 10.1109/ICIEAM48468.2020.9111995.

-
- [52] S. AbdulRahman, S. Otoum, O. Bouachir, and A. Mourad, "Management of Digital Twin-Driven IoT Using Federated Learning," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 11, pp. 3636–3649, Nov. 2023, doi: 10.1109/JSAC.2023.3310102.
 - [53] A. Saad, S. Faddel, and O. Mohammed, "IoT-Based Digital Twin for Energy Cyber-Physical Systems: Design and Implementation," *Energies*, vol. 13, no. 18, Art. no. 18, Jan. 2020, doi: 10.3390/en13184762.
 - [54] A. Canedo, "Industrial IoT lifecycle via digital twins," in *Proceedings of the Eleventh IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis*, in CODES '16. New York, NY, USA: Association for Computing Machinery, Oct. 2016, p. 1. doi: 10.1145/2968456.2974007.
 - [55] S. Ahelerooff, X. Xu, R. Y. Zhong, and Y. Lu, "Digital Twin as a Service (DTaaS) in Industry 4.0: An Architecture Reference Model," *Adv. Eng. Inform.*, vol. 47, p. 101225, Jan. 2021, doi: 10.1016/j.aei.2020.101225.
 - [56] P. Stavropoulos and D. Mourtzis, "Chapter 10 - Digital twins in industry 4.0," in *Design and Operation of Production Networks for Mass Personalization in the Era of Cloud Technology*, D. Mourtzis, Ed., Elsevier, 2022, pp. 277–316. doi: 10.1016/B978-0-12-823657-4.00010-5.
 - [57] A. To *et al.*, "Drone-Based AI and 3D Reconstruction for Digital Twin Augmentation," in *Social Computing and Social Media: Experience Design and Social Network Analysis*, G. Meiselwitz, Ed., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021, pp. 511–529. doi: 10.1007/978-3-030-77626-8_35.
 - [58] N. Grigoropoulos and S. Lalis, "Simulation and Digital Twin Support for Managed Drone Applications," in *2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, Sep. 2020, pp. 1–8. doi: 10.1109/DS-RT50469.2020.9213676.
 - [59] O. Zheng, M. Abdel-Aty, L. Yue, A. Abdelraouf, Z. Wang, and N. Mahmoud, "CitySim: A Drone-Based Vehicle Trajectory Dataset for Safety Oriented Research and Digital Twins," *Transp. Res. Rec. J. Transp. Res. Board*, p. 03611981231185768, Jul. 2023, doi: 10.1177/03611981231185768.
 - [60] Y. Yigit *et al.*, "TwinPort: 5G drone-assisted data collection with digital twin for smart seaports," *Sci. Rep.*, vol. 13, no. 1, Art. no. 1, Jul. 2023, doi: 10.1038/s41598-023-39366-1.
 - [61] R. Zhang, F. Wang, J. Cai, Y. Wang, H. Guo, and J. Zheng, "Digital twin and its applications: A survey," *Int. J. Adv. Manuf. Technol.*, vol. 123, no. 11, pp. 4123–4136, Dec. 2022, doi: 10.1007/s00170-022-10445-3.
 - [62] E. Shahat, C. T. Hyun, and C. Yeom, "City Digital Twin Potentials: A Review and Research Agenda," *Sustainability*, vol. 13, no. 6, Art. no. 6, Jan. 2021, doi: 10.3390/su13063386.
 - [63] R. G. Alves *et al.*, "A digital twin for smart farming," in *2019 IEEE Global Humanitarian Technology Conference (GHTC)*, Oct. 2019, pp. 1–4. doi: 10.1109/GHTC46095.2019.9033075.
 - [64] R. Dong, C. She, W. Hardjawana, Y. Li, and B. Vucetic, "Deep Learning for Hybrid 5G Services in Mobile Edge Computing Systems: Learn From a Digital Twin," *IEEE Trans. Wirel. Commun.*, vol. 18, no. 10, pp. 4692–4707, Oct. 2019, doi: 10.1109/TWC.2019.2927312.
 - [65] Y. Dai, K. Zhang, S. Maharjan, and Y. Zhang, "Deep Reinforcement Learning for Stochastic Computation Offloading in Digital Twin Networks," *IEEE Trans. Ind. Inform.*, vol. 17, no. 7, pp. 4968–4977, Jul. 2021, doi: 10.1109/TII.2020.3016320.

-
- [66] H. Hong *et al.*, “NetGraph: An Intelligent Operated Digital Twin Platform for Data Center Networks,” in *Proceedings of the ACM SIGCOMM 2021 Workshop on Network-Application Integration*, Virtual Event USA: ACM, Aug. 2021, pp. 26–32. doi: 10.1145/3472727.3472802.
 - [67] C. Zhou *et al.*, “Digital Twin Network: Concepts and Reference Architecture,” Internet Engineering Task Force, Internet Draft draft-irtf-nmrg-network-digital-twin-arch-04, Oct. 2023. Accessed: Feb. 21, 2024. [Online]. Available: <https://datatracker.ietf.org/doc/draft-irtf-nmrg-network-digital-twin-arch>
 - [68] D. R. R. Raj, T. A. Shaik, A. Hirwe, P. Tamma, and K. Kataoka, “Building a Digital Twin Network of SDN Using Knowledge Graphs,” *IEEE Access*, vol. 11, pp. 63092–63106, 2023, doi: 10.1109/ACCESS.2023.3288813.
 - [69] H. Adhami, M. Alja’afreh, M. Hoda, J. Zhao, Y. Zhou, and A. El Saddik, “Suitability of SDN and MEC to facilitate digital twin communication over LTE-A,” *Digit. Commun. Netw.*, Jun. 2023, doi: 10.1016/j.dcan.2023.05.008.
 - [70] P. Kumar, R. Kumar, A. Aljuhani, D. Javeed, A. Jolfaei, and A. K. M. N. Islam, “Digital twin-driven SDN for smart grid: A deep learning integrated blockchain for cybersecurity,” *Sol. Energy*, vol. 263, p. 111921, Oct. 2023, doi: 10.1016/j.solener.2023.111921.
 - [71] M. Kherbache, M. Maimour, and E. Rondeau, “When Digital Twin Meets Network Softwarization in the Industrial IoT: Real-Time Requirements Case Study,” *Sensors*, vol. 21, p. 8194, Dec. 2021, doi: 10.3390/s21248194.
 - [72] Y. Hu, G. Min, J. Li, Z. Li, Z. Cai, and J. Zhang, “VNF Migration in Digital Twin Network for NFV Environment,” *Electronics*, vol. 12, no. 20, Art. no. 20, Jan. 2023, doi: 10.3390/electronics12204324.
 - [73] M. Sanz Rodrigo, D. Rivera, J. Moreno, and X. Larriva-Novo, *Diseño de un ecosistema DTN basado en NFV para el análisis de ciberseguridad en redes 5G*. 2022.
 - [74] “Designing data center networks using bottleneck structures | Proceedings of the 2021 ACM SIGCOMM 2021 Conference.” Accessed: Feb. 21, 2024. [Online]. Available: <https://dl.acm.org/doi/10.1145/3452296.3472898>
 - [75] J. Ros-Giralt *et al.*, “On the Bottleneck Structure of Congestion-Controlled Networks,” *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 3, no. 3, p. 59:1–59:31, Dec. 2019, doi: 10.1145/3366707.
 - [76] S. Mihai *et al.*, “Digital Twins: A Survey on Enabling Technologies, Challenges, Trends and Future Prospects,” *IEEE Commun. Surv. Tutor.*, vol. 24, no. 4, pp. 2255–2291, 2022, doi: 10.1109/COMST.2022.3208773.
 - [77] Y. Wu, K. Zhang, and Y. Zhang, “Digital Twin Networks: A Survey,” *IEEE Internet Things J.*, vol. 8, no. 18, pp. 13789–13804, Sep. 2021, doi: 10.1109/JIOT.2021.3079510.
 - [78] “Eclipse Ditto™ • open source framework for digital twins in the IoT.” Accessed: Mar. 14, 2023. [Online]. Available: <https://www.eclipse.org/ditto/>
 - [79] “FIWARE - Open APIs for Open Minds.” Accessed: Mar. 14, 2023. [Online]. Available: <https://www.fiware.org/>
 - [80] P. Almasan *et al.*, “Digital Twin Network: Opportunities and Challenges.” arXiv, Jan. 07, 2022. doi: 10.48550/arXiv.2201.01144.
 - [81] Y. Zhu, D. Chen, C. Zhou, L. Lu, and X. Duan, “A knowledge graph based construction method for Digital Twin Network,” in *2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI)*, Jul. 2021, pp. 362–365. doi: 10.1109/DTPI52967.2021.9540177.

-
- [82] H. X. Nguyen, R. Trestian, D. To, and M. Tatipamula, "Digital Twin for 5G and Beyond," *IEEE Commun. Mag.*, vol. 59, no. 2, pp. 10–15, Feb. 2021, doi: 10.1109/MCOM.001.2000343.
 - [83] H. Wang, Y. Wu, G. Min, and W. Miao, "A Graph Neural Network-Based Digital Twin for Network Slicing Management," *IEEE Trans. Ind. Inform.*, vol. 18, no. 2, pp. 1367–1376, Feb. 2022, doi: 10.1109/TII.2020.3047843.
 - [84] W. Sun, H. Zhang, R. Wang, and Y. Zhang, "Reducing Offloading Latency for Digital Twin Edge Networks in 6G," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12240–12251, Oct. 2020, doi: 10.1109/TVT.2020.3018817.
 - [85] G. N. Schroeder, C. Steinmetz, R. N. Rodrigues, R. V. B. Henriques, A. Rettberg, and C. E. Pereira, "A Methodology for Digital Twin Modeling and Deployment for Industry 4.0," *Proc. IEEE*, vol. 109, no. 4, pp. 556–567, Apr. 2021, doi: 10.1109/JPROC.2020.3032444.
 - [86] P. Janda, Z. Hajicek, and P. Bernardin, "Implementation of The Digital Twin Methodology," in *DAAAM Proceedings*, 1st ed., vol. 1, B. Katalinic, Ed., DAAAM International Vienna, 2019, pp. 0533–0538. doi: 10.2507/30th.daaam.proceedings.072.
 - [87] C. Gehrman and M. Gunnarsson, "A Digital Twin Based Industrial Automation and Control System Security Architecture," *IEEE Trans. Ind. Inform.*, vol. 16, no. 1, pp. 669–680, Jan. 2020, doi: 10.1109/TII.2019.2938885.
 - [88] C.-C. Chung, C.-C. Pai, F.-S. Ching, C. Wang, and L.-J. Chen, "When post-quantum cryptography meets the internet of things: an empirical study," in *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services*, Portland Oregon: ACM, Jun. 2022, pp. 525–526. doi: 10.1145/3498361.3538766.
 - [89] D. J. Bernstein and T. Lange, "Post-quantum cryptography," *Nature*, vol. 549, no. 7671, Art. no. 7671, Sep. 2017, doi: 10.1038/nature23461.
 - [90] W. Hu, T. Zhang, X. Deng, Z. Liu, and J. Tan, "Digital twin: a state-of-the-art review of its enabling technologies, applications and challenges," *J. Intell. Manuf. Spec. Equip.*, vol. 2, pp. 1–34, Aug. 2021, doi: 10.1108/JIMSE-12-2020-010.
 - [91] E. Haleplidis, K. Pentikousis, S. Denazis, J. Salim, D. Meyer, and O. Koufopavlou, "RFC 7426: Software-Defined Networking (SDN): Layers and Architecture Terminology," *IRTF*, Jan. 2015.
 - [92] "¿Qué son las redes definidas por software (SDN)? | Glosario de VMware | ES." Accessed: Feb. 21, 2024. [Online]. Available: <https://www.vmware.com/es/topics/glossary/content/software-defined-networking.html>
 - [93] "What is SDN: Software Defined Networking » Electronics Notes." Accessed: Feb. 21, 2024. [Online]. Available: <https://www.electronics-notes.com/articles/connectivity/data-networks/sdn-what-is-it-basics-technology.php>
 - [94] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network Function Virtualization: State-of-the-Art and Research Challenges," *IEEE Commun. Surv. Tutor.*, vol. 18, no. 1, pp. 236–262, 2016, doi: 10.1109/COMST.2015.2477041.
 - [95] K. Kaur, V. Mangat, and K. Kumar, "A review on Virtualized Infrastructure Managers with management and orchestration features in NFV architecture," *Comput. Netw.*, vol. 217, p. 109281, Nov. 2022, doi: 10.1016/j.comnet.2022.109281.
 - [96] "Network Functions Virtualisation" ETSI white Paper. Accessed: Feb. 21, 2024. [Online]. Available: https://portal.etsi.org/NFV/NFV_White_Paper2.pdf

-
- [97] "Network Functions Virtualisation (NFV); Management and Orchestration". ETSI Group Specification. Accessed: Feb. 21, 2024. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/nfv-man/001_099/001/01.01.01_60/gs_nfv-man001v010101p.pdf
 - [98] "Open Source MANO." OSM Official Page. Accessed: Dec. 01, 2022. [Online]. Available: <https://osm.etsi.org/>
 - [99] "OSM Release fourteen" Notes about OSM release fourteen. Accessed: Feb. 21, 2024. [Online]. Available: https://osm-download.etsi.org/ftp/osm-14.0-fourteen/OSM_Release_FOURTEEN_Release_Notes.pdf
 - [100] "OSM Information Model" Accessed: Dec. 01, 2022. [Online]. Available: <https://osm.etsi.org/docs/user-guide/latest/11-osm-im.html>
 - [101] "Open Source Cloud Computing Platform Software," OpenStack. Accessed: Feb. 21, 2024. [Online]. Available: <https://www.openstack.org/software/>
 - [102] "Vision for OpenStack Clouds — OpenStack Technical Committee Governance Documents." Accessed: Feb. 21, 2024. [Online]. Available: https://governance.openstack.org/tc/reference/technical-vision.html?_ga=2.74946322.1056786473.1685611747-909653073.1680082729
 - [103] "DevStack — DevStack documentation." Accessed: Feb. 21, 2024. [Online]. Available: <https://docs.openstack.org/devstack/latest/>
 - [104] "¿Qué es Kubernetes?" Accessed: Feb. 21, 2024. [Online]. Available: <https://kubernetes.io/es/docs/concepts/overview/what-is-kubernetes/>
 - [105] "MicroK8s - Zero-ops Kubernetes for developers, edge and IoT | MicroK8s," microk8s.io. Accessed: Feb. 21, 2024. [Online]. Available: <http://microk8s.io>
 - [106] "Creating a cluster with kubeadm." Accessed: Feb. 21, 2024. [Online]. Available: <https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/create-cluster-kubeadm/>
 - [107] "Welcome to Open Source MANO's VNF Onboarding guide! — Open Source MANO 6.0 documentation." Accessed: Feb. 21, 2024. [Online]. Available: <https://osm.etsi.org/docs/vnf-onboarding-guidelines/index.html#>
 - [108] "The simplest way to deploy and maintain applications in the cloud," Juju. Accessed: Feb. 21, 2024. [Online]. Available: <https://juju.is>
 - [109] "Helm." Official site. Accessed: Feb. 21, 2024. [Online]. Available: <https://helm.sh/>
 - [110] "MQTT - The Standard for IoT Messaging." Official site. Accessed: Dec. 01, 2022. [Online]. Available: <https://mqtt.org/>
 - [111] "ISO/IEC 20922:2016, MQTT v3.11" ISO. Accessed: Mar. 18, 2024. [Online]. Available: <https://www.iso.org/standard/69466.html>
 - [112] "Open Quantum Safe", Mar. 17, 2024. Accessed: Mar. 18, 2024. [Online]. Available: <https://github.com/open-quantum-safe/liboqs>
 - [113] I. T. L. Computer Security Division, "Selected Algorithms 2022 - Post-Quantum Cryptography | CSRC | CSRC," CSRC | NIST. Accessed: Mar. 18, 2024. [Online]. Available: <https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022>
 - [114] "Apache Kafka," Official site. Accessed: Dec. 01, 2022. [Online]. Available: <https://kafka.apache.org/>
 - [115] "RabbitMQ: One broker to queue them all." Official site. Accessed: Mar. 18, 2024. [Online]. Available: <https://www.rabbitmq.com/>
 - [116] "AMQP." Official site. Accessed: Mar. 18, 2024. [Online]. Available: <https://www.amqp.org/>

-
- [117] A. A. Abdul Ghaffar, A. Mahmoud, T. Sheltami, and M. Abu-Amara, "A Survey on Software-Defined Networking-Based 5G Mobile Core Architectures," *Arab. J. Sci. Eng.*, vol. 48, no. 2, pp. 2313–2330, Feb. 2023, doi: 10.1007/s13369-022-07183-9.
 - [118] J. Yao, Z. Han, M. Sohail, and L. Wang, "A Robust Security Architecture for SDN-Based 5G Networks," *Future Internet*, vol. 11, no. 4, Art. no. 4, Apr. 2019, doi: 10.3390/fi11040085.
 - [119] "Software Defined 5G and 6G Networks: a Survey | Mobile Networks and Applications." Accessed: Feb. 21, 2024. [Online]. Available: <https://dl.acm.org/doi/abs/10.1007/s11036-019-01397-2>
 - [120] S. A. Syed *et al.*, "QoS Aware and Fault Tolerance Based Software-Defined Vehicular Networks Using Cloud-Fog Computing," *Sensors*, vol. 22, no. 1, Art. no. 1, Jan. 2022, doi: 10.3390/s22010401.
 - [121] "5G Network Slicing for Vehicle-to-Everything Services | IEEE Journals & Magazine | IEEE Xplore." Accessed: Feb. 21, 2024. [Online]. Available: <https://ieeexplore.ieee.org/document/8246845/keywords#keywords>
 - [122] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," *IEEE Commun. Mag.*, vol. 53, no. 2, pp. 90–97, Feb. 2015, doi: 10.1109/MCOM.2015.7045396.
 - [123] A. A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines, "5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges," *Comput. Netw.*, vol. 167, p. 106984, Feb. 2020, doi: 10.1016/j.comnet.2019.106984.
 - [124] "5G PPP Architecture Working Group, 'View on 5G Architecture' #4." Accessed: Feb. 21, 2024. [Online]. Available: <https://5g-ppp.eu/5g-ppp-architecture-working-group-view-on-5g-architecture-4/>
 - [125] "IMT-2020 (a.k.a. '5G')." ITU. Accessed: Mar. 20, 2024. [Online]. Available: <https://www.itu.int:443/en/ITU-R/study-groups/rsg5/rwp5d/imt-2020/Pages/default.aspx>
 - [126] "Minimum requirements related to technical performance for IMT-2020 radio interface(s)," ITU. Accessed: Mar. 20, 2024. [Online]. Available: <https://www.itu.int:443/en/publications/ITU-R/Pages/publications.aspx>
 - [127] "5G Deployment: Standalone vs. Non-Standalone from the Operator Perspective | IEEE Journals & Magazine | IEEE Xplore." Accessed: Feb. 21, 2024. [Online]. Available: <https://ieeexplore.ieee.org/document/9269939>
 - [128] "Github repository dukl/5gCore." Jan. 06, 2024. Accessed: Feb. 21, 2024. [Online]. Available: <https://github.com/dukl/5gCore>
 - [129] "Github repository free5gc/free5gc: Open source 5G core network base on 3GPP R15." Accessed: Feb. 21, 2024. [Online]. Available: <https://github.com/free5gc/free5gc>
 - [130] "Github repository open5gs/open5gs: Open5GS is a C-language Open Source implementation for 5G Core and EPC, i.e. the core network of LTE/NR network (Release-17)." Accessed: Feb. 21, 2024. [Online]. Available: <https://github.com/open5gs/open5gs>
 - [131] "Github repository cn5g · GitLab," GitLab. Accessed: Feb. 21, 2024. [Online]. Available: <https://gitlab.eurecom.fr/oai/cn5g>
 - [132] "Github repository bubblecounter/Internship-5GCN: Implementation of RESTful Web Services between 5G Control Plane Nodes(AMF,NRF,SMF,UDM)." Accessed: Feb. 21, 2024. [Online]. Available: <https://github.com/bubblecounter/Internship-5GCN>

-
- [133] "Github repository Orange-OpenSource/towards5gs-helm." Orange, Feb. 15, 2024. Accessed: Feb. 21, 2024. [Online]. Available: <https://github.com/Orange-OpenSource/towards5gs-helm>
 - [134] "Github repository free5G/free5GRAN." free5G, Jan. 21, 2024. Accessed: Feb. 21, 2024. [Online]. Available: <https://github.com/free5G/free5GRAN>
 - [135] "Github repository 5G Test tool - gNBSim." Mar. 2022. Accessed: Feb. 21, 2024. [Online]. Available: <https://github.com/omec-project/gnbsim>
 - [136] "Github repository oai / openairinterface5G · GitLab," GitLab. Accessed: Feb. 21, 2024. [Online]. Available: <https://gitlab.eurecom.fr/oai/openairinterface5g>
 - [137] "Github repository srsran/srsRAN_4G." srsRAN, Feb. 21, 2024. Accessed: Feb. 21, 2024. [Online]. Available: https://github.com/srsran/srsRAN_4G
 - [138] "Github repository aligungr/UERANSIM." Feb. 20, 2024. Accessed: Feb. 21, 2024. [Online]. Available: <https://github.com/aligungr/UERANSIM>
 - [139] "Github repository UPM-RSTI/STGUTG: Signaling and user traffic injection tools for 5G testing platforms." Accessed: Dec. 09, 2022. [Online]. Available: <https://github.com/UPM-RSTI/STGUTG>
 - [140] D. Rivera, J. I. Moreno, M. S. Rodrigo, D. R. López, and A. Mozo, "Providing Heterogeneous Signaling and User Traffic for 5G Core Network Functional Testing," *IEEE Access*, vol. 11, pp. 2968–2980, 2023, doi: 10.1109/ACCESS.2022.3233412.
 - [141] "IMT towards 2030 and beyond," ITU. Accessed: Mar. 20, 2024. [Online]. Available: <https://www.itu.int:443/en/ITU-R/study-groups/rsg5/rwp5d/imt-2030/Pages/default.aspx>
 - [142] "Recommendation ITU-R M.2160-0 (11/2023) - Framework and overall objectives of the future development of IMT for 2030 and beyond".
 - [143] "Y.3090 : Digital twin network - Requirements and architecture." Accessed: Mar. 20, 2024. [Online]. Available: <https://www.itu.int/rec/T-REC-Y.3090>
 - [144] X. Lin, L. Kundu, C. Dick, E. Obiodu, T. Mostak, and M. Flaxman, "6G Digital Twin Networks: From Theory to Practice," *IEEE Commun. Mag.*, vol. 61, no. 11, pp. 72–78, Nov. 2023, doi: 10.1109/MCOM.001.2200830.
 - [145] H. Shin, S. Oh, A. Isah, I. Aliyu, J. Park, and J. Kim, "Network Traffic Prediction Model in a Data-Driven Digital Twin Network Architecture," *Electronics*, vol. 12, no. 18, Art. no. 18, Jan. 2023, doi: 10.3390/electronics12183957.
 - [146] "Digital Twin-Assisted Real-Time Traffic Data Prediction Method for 5G-Enabled Internet of Vehicles | IEEE Journals & Magazine | IEEE Xplore." Accessed: Feb. 21, 2024. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/9440709>
 - [147] Z. Zhou *et al.*, "Secure and Latency-Aware Digital Twin Assisted Resource Scheduling for 5G Edge Computing-Empowered Distribution Grids," *IEEE Trans. Ind. Inform.*, vol. 18, no. 7, pp. 4933–4943, Jul. 2022, doi: 10.1109/TII.2021.3137349.
 - [148] S. Vakaruk, A. Mozo, A. Pastor, and D. R. López, "A Digital Twin Network for Security Training in 5G Industrial Environments," in *2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI)*, Jul. 2021, pp. 395–398. doi: 10.1109/DTPI52967.2021.9540146.
 - [149] K. Gai *et al.*, "Digital Twin-enabled AI Enhancement in Smart Critical Infrastructures for 5G," *ACM Trans. Sens. Netw.*, vol. 18, no. 3, p. 45:1-45:20, Sep. 2022, doi: 10.1145/3526195.

-
- [150] "Placement of Social Digital Twins at the Edge for Beyond 5G IoT Networks | IEEE Journals & Magazine | IEEE Xplore." Accessed: Feb. 21, 2024. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/9828402>
 - [151] F. Rebecchi *et al.*, "A Digital Twin for the 5G Era: the SPIDER Cyber Range," in *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Jun. 2022, pp. 567–572. doi: 10.1109/WoWMoM54355.2022.00088.
 - [152] J. Santa *et al.*, "MIGRATE: Mobile Device Virtualisation Through State Transfer," *IEEE Access*, vol. 8, pp. 25848–25862, 2020, doi: 10.1109/ACCESS.2020.2971090.
 - [153] "A Digital Twin for the 5G Era: the SPIDER Cyber Range | IEEE Conference Publication | IEEE Xplore." Accessed: Feb. 21, 2024. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/9842774>
 - [154] C. Grasselli, A. Melis, L. Rinieri, D. Berardi, G. Gori, and A. A. Sadi, "An Industrial Network Digital Twin for enhanced security of Cyber-Physical Systems," in *2022 International Symposium on Networks, Computers and Communications (ISNCC)*, Jul. 2022, pp. 1–7. doi: 10.1109/ISNCC55209.2022.9851731.
 - [155] C. Contoli, D. Rossi, G. Tontini, D. Borsatti, and F. Callegati, "Demonstration of Digital Twins for 5G Connectivity in Industry 4.0," in *2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Nov. 2021, pp. 102–103. doi: 10.1109/NFV-SDN53031.2021.9665099.
 - [156] J. Baranda *et al.*, "Demo: AIML-as-a-Service for SLA management of a Digital Twin Virtual Network Service," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, May 2021, pp. 1–2. doi: 10.1109/INFOCOMWKSHPS51825.2021.9484610.
 - [157] S. M. Raza, R. Minerva, N. Crespi, and M. Karech, "Definition Of Digital Twin Network Data Model in The Context of Edge-Cloud Continuum," in *2023 IEEE 9th International Conference on Network Softwarization (NetSoft)*, Madrid, Spain: IEEE, Jun. 2023, pp. 402–407. doi: 10.1109/NetSoft57336.2023.10175444.
 - [158] "Especificación NFV-SOL006." Accessed: Feb. 21, 2024. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/NFV-SOL/001_099/006/03.05.01_60/gs_NFV-SOL006v030501p.pdf
 - [159] S. Shalunov *et al.*, "Application-Layer Traffic Optimization (ALTO) Protocol," Internet Engineering Task Force, Request for Comments RFC 7285, Sep. 2014. doi: 10.17487/RFC7285.
 - [160] "OSM repository about NSD information model" Accessed: Feb. 21, 2024. [Online]. Available: https://osm-download.etsi.org/repository/osm/debian/ReleaseFOURTEEN/docs/osm-im/osm_im_trees/etsi-nfv-nsd.tree.txt
 - [161] "OSM repository about VNFD information model" Accessed: Feb. 21, 2024. [Online]. Available: https://osm-download.etsi.org/repository/osm/debian/ReleaseFOURTEEN/docs/osm-im/osm_im_trees/etsi-nfv-vnfd.tree.txt
 - [162] F. Tao, M. Zhang, Y. Liu, and A. Y. C. Nee, "Digital twin driven prognostics and health management for complex equipment," *CIRP Ann.*, vol. 67, no. 1, pp. 169–172, Jan. 2018, doi: 10.1016/j.cirp.2018.04.055.
 - [163] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, "Characterising the Digital Twin: A systematic literature review," *CIRP J. Manuf. Sci. Technol.*, vol. 29, pp. 36–52, May 2020, doi: 10.1016/j.cirpj.2020.02.002.

-
- [164] "Paho MQTT." Official site. Accessed: Feb. 21, 2024. [MacOS :: MacOS X, Microsoft :: Windows, POSIX]. Available: <http://eclipse.org/paho>
- [165] "Elasticsearch." Official site. Accessed: Feb. 21, 2024. [Online]. Available: <https://www.elastic.co/es/elasticsearch>
- [166] "Github repository free5gc/free5gc-compose." free5GC, Feb. 20, 2024. Accessed: Feb. 21, 2024. [Online]. Available: <https://github.com/free5gc/free5gc-compose>
- [167] A. Mozo, A. Pastor, A. Karamchandani, L. de la Cal, D. Rivera, and J. I. Moreno, "Integration of Machine Learning-Based Attack Detectors into Defensive Exercises of a 5G Cyber Range," *Appl. Sci.*, vol. 12, no. 20, Art. no. 20, Jan. 2022, doi: 10.3390/app122010349.
- [168] "OpenSSL." Official site. Accessed: Mar. 25, 2024. [Online]. Available: <https://www.openssl.org/>
- [169] P. W. Shor, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," *SIAM J. Comput.*, vol. 26, no. 5, pp. 1484–1509, Oct. 1997, doi: 10.1137/S0097539795293172.
- [170] I. T. L. Computer Security Division, "Round 3 Submissions - Post-Quantum Cryptography | CSRC | CSRC," CSRC | NIST. Accessed: Mar. 25, 2024. [Online]. Available: <https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions>