

Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingenieros de Telecomunicación



**ESTUDIO DE LA TECNOLOGÍA NFV
APLICADA AL NÚCLEO EPC EN REDES
LTE Y ANÁLISIS DE ENTORNO DE
ORQUESTACIÓN OSM PARA SU GESTIÓN**

TRABAJO FIN DE MÁSTER

Jaime Andrés Cárdenas Córdoba

2016

Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingenieros de Telecomunicación

**Máster Universitario en
Ingeniería de Redes y Servicios Telemáticos**

TRABAJO FIN DE MÁSTER

**ESTUDIO DE LA TECNOLOGÍA NFV
APLICADA AL NÚCLEO EPC EN REDES
LTE Y ANÁLISIS DE ENTORNO DE
ORQUESTACIÓN OSM PARA SU GESTIÓN**

Autor

Jaime Andrés Cárdenas Córdova

Director

David Fernández Cambrero

Departamento de Ingeniería de Sistemas Telemáticos

2016

Resumen

El foco álgido de las comunicaciones desde sus inicios ha sido el direccionamiento de los datos. Se han implantado técnicas y métodos para mejorar la eficiencia de su transmisión, pero para ello se necesita de equipos dedicados cada vez más sofisticados y potentes que tienen características funcionales especializadas para el tratamiento de la información, lo que ha generado que las distintas infraestructuras de red se vuelvan cada vez más complicadas, tanto para operarla como para su mantenimiento. Esto conlleva gastos en adquisición de equipos, soporte de profesionales y mantenimiento de red. Por tal motivo, la convergencia actual de las redes de comunicaciones ha enfocado su objetivo en virtualizar su funcionamiento con el fin de tener una gestión mediante software, dando paso a tecnologías como NFV y SDN.

NFV se ha planteado como solución actual al crecimiento desmedido del tráfico de datos. Se necesita agilizar su control y a la vez se busca abaratar costos en implementación y gestión de servicios por parte de los operadores de red. En la actualidad las redes móviles son el medio de transmisión de datos más utilizado, siendo los principales interesados en innovar sus arquitecturas de red con el fin de mejorar sus servicios y prestaciones pero con una gestión flexible a costos más bajos, prestando atención a las redes LTE como el objetivo de implementación de NFV.

Con esta visión general se plantea el estudio del estado del arte de las tecnologías NFV, analizando profundamente los conceptos básicos para su aplicación teórica sobre los componentes del núcleo de red en entornos LTE. Con esto se explican y analizan los métodos viables para su implementación, así como el despliegue, buscando la factibilidad del uso de la tecnología de virtualización.

Además el proyecto se adentra en el análisis de herramientas actuales desarrolladas para el uso, despliegue y gestión de redes con funciones virtualizadas. Se estudian las funcionalidades que cada una presenta, así como su arquitectura y compatibilidad con distintos entornos. Todas las herramientas detalladas están basadas en código libre lo cual permite tener un mejor control sobre ellas al momento de su utilización o modificación a conveniencias de estudio.

Para entender el trabajo que realizan estas herramientas, se ha seleccionado una de las más actuales del mercado conocida como OSM y junto con ella se planteó un escenario sencillo virtualizado en VNX donde se puede poner en práctica el estado del arte investigado. De este modo se tiene un acercamiento más tangible sobre el uso y despliegue de funciones de red virtualizadas, estudiando paso a paso todos los procedimientos que se efectúan al momento de ejecutar un entorno de red con NFVs.

Abstract

The main goal of communications since its beginning has been the data addressing. Have been implemented techniques and methods to improve the efficiency of transmission, but this requires increasingly sophisticated and powerful dedicated computers that have characteristics and specialized functions for information processing, which has meant that the various infrastructures network become highly complicated, both to operate and to maintain. This involves costs in acquiring equipment, professional support and network maintenance. Therefore, the current convergence of communications networks has focused in virtualizing its operation in order to have a management by software, leading to technologies such as NFV and SDN.

Has been proposed NFV as the current solution to uncontrolled growth of data traffic. It is necessary expedite its control and at the same time seeks to decrease costs in implementation and services management by network operators. Today mobile networks are the most used way of data transmission, being the primarily interested in innovating their network architectures in order to improve their services and performance but with a lower flexible management and costs, being the LTE networks the NFV deployment goal.

With this overview, it is projected technology NFV state of the art study, having a deep analysis of the basic concepts in order to its theoretical application over the LTE core networks environments. With this, they explain and analyze all the viable methods for implementation and deployment, and thus seek the feasibility of using virtualization technology.

The project also focus in current tools analyzing, developed for use, deployment and management of virtualized networks functions. The features that each one presents as well as its architecture and compatibility with different environments are studied. All detailed tools works with open source code, which allows better control over them at the time of use or modification to study conveniences.

To understand the work of these tools, we have selected one of the current market known as OSM, and while have used a virtualized VNX simple scenario where you can implement state of the art investigation. Therefore, you have a more tangible approach with the use and deployment of virtualized network functions; studying systematically, all the procedures performed when running a network environment with NFVs.

Índice general

Contenido

Resumen	i
Abstract.....	ii
Índice general.....	iii
Índice de figuras.....	vi
Siglas	viii
1 Introducción.....	1
1.1 Motivación.....	2
1.2 Objetivos	3
1.3 Estructura del Documento	4
2 Sistemas de Comunicación Móvil.....	6
2.1 Redes Móviles de Telecomunicación.....	6
2.1.1 Introducción.....	6
2.1.2 Redes Móviles de Nueva Generación.....	8
2.1.3 Antecedentes de la Evolución 3G.....	9
2.1.4 Evolución a UMTS.....	10
2.2 Redes Móviles LTE.....	12
2.3 Arquitectura de Red LTE	14
2.3.1 Arquitectura E-UTRAN.....	15
2.3.2 Arquitectura de EPC.....	16
2.3.3 Serving Gateway (S-GW)	17
2.3.4 PDN Gateway (P-GW).....	18
2.3.5 Arquitectura de Protocolos de Comunicación a través de LTE.....	19
3 Estudio de la Tecnología NFV	22
3.1 Virtualización.....	22
3.2 Virtualización de Redes.....	24

3.3	Virtualización de Funciones de Red (NFV)	24
3.3.1	NFV ISG (Grupo de Especificación de la Industria).....	26
3.3.2	Marco Arquitectónico NFV.....	26
3.3.3	Infraestructura de NFV.....	27
3.3.4	NFV MANO (Gestión y Orquestación).....	28
3.3.5	NFV Casos de Uso.....	29
3.4	Beneficios de NFV	31
4	Desarrollo	33
4.1	Aplicación de NFV a Redes Móviles LTE.....	33
4.2	VNF de EPC	34
4.2.1	Coexistencia entre redes Virtualizadas y No-Virtualizadas de EPC	37
4.2.2	Virtualización Parcial de EPC.....	37
4.2.3	Virtualización de la Entidad de Gestión de Movilidad (vMME)	39
4.3	Agrupamiento de Entidades EPC en entorno NFV.....	40
4.3.1	Segmento Uno.....	41
4.3.2	Segmento Dos	41
4.3.3	Segmento Tres.....	42
4.3.4	Segmento Cuatro	42
4.4	Tipos de Escenarios de Comunicación con vEPC.....	43
4.4.1	Escenario de Comunicación End to End	43
4.4.2	Escenario con Arquitectura de Plano de Usuario sin Estado.....	44
4.4.3	Uso del Protocolo OpenFlow en Arquitectura LTE-EPC.....	45
4.5	Recomendaciones en Implementación de Virtualización de EPC.....	48
4.6	Retos y Aplicaciones de la Gestión y Orquestación de NFV.....	49
4.6.1	Entorno MANO desarrollado por la ETSI	49
4.6.2	Retos y Puntos de Investigación.....	53
4.6.3	Proyectos Desarrollados Sobre NFV MANO	54
5	Caso Demostrativo.....	61
5.1	Componentes Básicos de Escenario	61
5.1.1	Herramienta VNX	62
5.1.2	Herramienta OPENMANO.....	64

5.1.3	Herramienta OPENSATACK	66
5.1.4	Herramienta JUJU	70
5.1.5	Herramienta RIFT.ware.....	72
5.2	Propuesta de Escenario Práctico	76
5.2.1	Escenario Openstack	77
5.2.2	Escenario OSM.....	80
5.2.3	Detalles del Modelo de Datos.....	82
5.2.4	Proceso de Interconexión de Componentes.....	86
5.2.5	Flujo de Trabajo de OSM.....	92
5.3	Pasos Para Creación de Descriptores VNF y NS.....	97
6	Conclusiones y Líneas Futuras de Investigación.....	99
6.1	Conclusiones	99
6.2	Líneas Futuras de Investigación.....	101
	Bibliografía	102
	Anexos	105

Índice de figuras

Figura 2.1. Evolución de la Telefonía Móvil.....	8
Figura 2.2. Arquitectura de Red GSM e introducción de GPRS.	11
Figura 2.3. Arquitectura de Red LTE. [5].....	15
Figura 2.4. Red de Acceso E-UTRAN [5].....	16
Figura 2.5. Arquitectura de Núcleo de Red EPC[5].....	17
Figura 2.6. Funciones Principales y Conexiones Lógicas de S-GW [4].....	18
Figura 2.7. Funciones Principales y Conexiones Lógicas de P-GW [4].....	19
Figura 2.8. Pila de Protocolos de Plano de Usuario en LTE. [4].....	21
Figura 2.9. Pila de Protocolos de Plano de Control en LTE.[4].....	21
Figura 3.10. Pila de Protocolos de Plano de Control en LTE.[11].....	23
Figura 3.11. Pila de Protocolos de Plano de Control en LTE.[11].....	23
Figura 3.12. Pila de Protocolos de Plano de Control en LTE. [11].....	23
Figura 3.13. ETSI visión de NFV. [14].....	25
Figura 3.14. Marco Arquitectónico de tecnología NFV. [14].....	27
Figura 3.15. NFVI Multiplicidad y Campos de Aplicación NFV [16].....	28
Figura 3.16. Arquitectura de Gestión y Orquestación desarrollada por NFV. [17] ...	29
Figura 3.17. Pila de Protocolos de Plano de Control en LTE. [19].....	30
Figura 4.18. Virtualización General de EPC. [18].....	35
Figura 4.19. Creación y Reubicación de Instancias VNF.[18].....	36
Figura 4.20. Coexistencia entre Redes Virtualizadas y No Virtualizadas.[18]	37
Figura 4.21. .Virtualización Parcial de Componentes EPC.[18].....	38
Figura 4.22. Virtualización según Servicio Específico. [18].....	38
Figura 4.23. Sistema General vMME [21].....	39
Figura 4.24. Agrupamiento de entidades EPC en entorno NFV. [22].....	40
Figura 4.25. Agrupación de componentes vEPC. [22].....	43
Figura 4.26. Ejemplo de escenario vEPC con túnel tradicional GTP[23].....	44
Figura 4.27. Arquitectura Propuesta escenario plano de usuario sin estado. [24].....	45
Figura 4.28. Arquitectura vEPC Basada en OpenFlow [25].....	46
Figura 4.29. Proceso de unión inicial OF.[25].....	47
Figura 4.30. Proceso de configuración de portadora OF.[25].....	47
Figura 4.31. Arquitectura MANO propuesta por ETSI NFV. [27].....	50
Figura 4.32. Arquitectura funcional de BrahmaPutra.....	55
Figura 4.33. Arquitectura OpenMANO de Telefónica [27]......	56
Figura 4.34. Vista simplificada de la arquitectura OSM[28].....	57

Figura 4.35. Arquitectura de OpenStack.....	59
Figura 5.36. Arquitectura funcional de VNX [29].....	63
Figura 5.37. Pasos para creación de escenarios en VNX.....	64
Figura 5.38. Arquitectura NFVI y aplicación de Openmano.	65
Figura 5.39. Componente Nova con hipervisor KVM.	67
Figura 5.40. Diagrama de red Neutrón.	68
Figura 5.41. Arquitectura de Horizon.	68
Figura 5.42. Arquitectura básica de OpenStack con Neutrón.[31]	70
Figura 5.43. Estructura básica de juju.....	70
Figura 5.44. Arquitectura de funcionamiento de Juju.....	72
Figura 5.45. Estructura funcional de software Rift.ware. [32]	73
Figura 5.46. Arquitectura por capas de escenario experimental.	77
Figura 5.47. Escenario OpenStack Kilo.	78
Figura 5.48. Arquitectura mínima de Openstack con agentes.....	80
Figura 5.49. Arquitectura mínima de plataforma OSM con Switch físico.	81
Figura 5.50. Escenario virtualizado de OSM junto con Openstack.....	82
Figura 5.51. Especificación de estructura interna de VNFs.....	83
Figura 5.52. Estructura interna de Descriptor VNF (VNFD).....	84
Figura 5.53. Estructura interna de un descriptor de servicio de red.....	85
Figura 5.54. Estructura interna de Descriptor NS (NSD).....	85
Figura 5.55. Estructura interna de escenarios de red. Simple y complejo.....	86
Figura 5.56. Acoplamiento entre plataformas OSM y Openstack.....	87
Figura 5.57. Acoplamiento entre RO y Openstack.	89
Figura 5.58. Modelo de datos del paquete VNF. [34].....	90
Figura 5.59. Acoplamiento de SO con componentes CM y RO[35].....	91
Figura 5.60. Acoplamiento entre SO y RO en dashboard OSM.....	92
Figura 5.61. Acoplamiento SO y CM en dashboard OSM.....	92
Figura 5.62. Arquitectura actual del modelo OSM.....	93
Figura 5.63. Carga de ficheros NSD y VNFD en OSM.....	94
Figura 5.64. Gestión del ciclo de vida de red de servicio (NS). [36].....	95
Figura 5.65. Despliegue NSD en GUI de OSM.....	96
Figura 5.66. Proceso de Instanciación del despliegue de NS.	97

Siglas

3GPP	3 rd Generation Partnership Project
ARQ	Automatic Repeat Request
BSC	Base Station Controller
BSS	Base Station System
BTS	Base Transceiver Station
CDMA	Code Division Multiple Access
CP	Connection Points
E2E	End to End
EM	Element Management
EMS	Element Managers System
EPA	Enhanced Platform Awareness
EPC	Evolved Packet Core
ETSI	European Telecommunications Standards Institute
E-UTRAN	Evolved - UMTS Terrestrial Radio Access Network
GERAN	GSM/EDGE - Radio Access Network
GPRS	General Packet Radio system
GSM	Global System for Mobile Communications
GTP	GPRS Tunneling Protocol
HLR	Home Location Register
HSS	Home Subscriber Server
IaaS	Infrastructure as a Service
IETF	Internet Engineering Task Force

IMS	IP Multimedia System
ISG	Industry Specification Group
ITU	International Telecommunications Union
KVM	Kernel-based Virtual Machine
LTE	Long Term Evolution
M2M	Machine to Machine
MANO	Management and Orchestration
MIMO	Multiple Input Multiple Output
MME	Mobility Management Entity
MSC	Mobile Switching Centre
NFV	Network Function Virtualization
NFVI	NFV - Infrastructure
NFVO	NFV Orchestrator
NMS	Network Management System
NS	Network Service
NSD	NS Descriptor
NSS	Network Switching Subsystem
OF	Open Flow
OFC	OpenFlow Controller
OFDMA	Orthogonal frequency Domain Multiple Access
OSS/BSS	Operational Support systems - Business Support Systems
PaaS	Platform as a Service
PCEF	Policy and Charging Enforcement Functions
PCRF	Policy and Charging Rules Function
P-GW	Packet Data Network Gateway

RLC	Radio Link Control
RNC	Radio Network Controller
SAE	System Architecture Evolution
SDN	Software Defined Networking
S-GW	Serving Gateway
SMS	Short Message Service
UE	User Equipment
UMTS	Universal Mobile telecommunications
VDU	Virtual Deployment Units
vEPC	Virtual EPC
VIM	Virtualized Infrastructure Manager
VL	Virtual Link
VM	Virtual Machine
vMME	Virtual MME
VNF	Virtual Network Function
VNFD	VNF Descriptor
VNFM	VNF Manager

1 Introducción

La telefonía móvil desde sus inicios ha brindado facilidades a sus usuarios, generando un cambio en el paradigma de comunicación, a tal punto de convertirse en una herramienta indispensable en la actualidad, comparándose como cualquier servicio básico requerido por la sociedad. Esto ha obligado a las distintas operadoras de telefonía a mantener una evolución constante sobre sus arquitecturas y servicios. Siendo el cambio más notable la velocidad en transmisión de datos; la cual ha permitido a los usuarios disponer de dispositivos móviles cargados de aplicaciones, que hace pocos años atrás, eran impensables tener.

El número de usuarios que integran las redes de comunicaciones a través de dispositivos móviles ha crecido de manera exponencial en las últimas décadas, esto ha implicado que exista cada vez más volumen de tráfico de datos que se deba controlar y direccionar. Complicando tanto, la arquitectura de las redes, como las técnicas de direccionamiento y pasarela de los datos. Lo cual, a la vez, ha venido elevando los costos de mantenimiento y operación de sus equipos, que en la actualidad son de propietarios dedicados. Del mismo modo genera un aumento en el costo de los servicios que se brinda a los usuarios.

El evitar cuellos de botellas y aligerar el tráfico en las comunicaciones, ha sido un problema constante en las comunicaciones, por lo que se han venido planteando distintas alternativas como medidas para solucionar estos inconvenientes, como por ejemplo, técnicas de tunelado, aumento de eficiencia en espectro radioeléctrico para acceso al medio, etc. Pero en el núcleo de las redes no se han realizado cambios sustanciales en su entorno.

Recientemente se ha propuesto una nueva solución, para encarar el futuro de las comunicaciones, se trata de la virtualización de funciones de red, sus siglas en inglés NFV (Network Function Virtualization), la cual va adquiriendo gran aceptación dentro de centros de investigación, universidades y operadoras privadas. Gracias a su versatilidad en funcionamiento y efectividad en costo - beneficio, se estudia su implementación dentro de las redes de comunicaciones actuales y futuras.

La virtualización de la red surge como iniciativa a la cuestión de proporcionar apoyo a la nueva convergencia del tratamiento del flujo de datos, optimizando los recursos físicos tales como hardware y software de gestión. NFV está dirigido a resolver los problemas de infraestructura tecnológica a través de técnicas de virtualización con el fin de concentrar una gran cantidad de equipos de diferentes características dentro de servidores de alto rendimiento y sistemas de almacenamiento, que se encuentran en centros de datos.

Aplicar NFV al núcleo de redes móviles permitirá a los operadores a tener un mejor control sobre su tráfico, dinamizando sus comunicaciones, y en su completa implementación, mejorar el rendimiento de sus servicios, soportando la alta demanda de usuarios y el flujo exponencial de datos. Además, les admitirá no depender de un núcleo estático o rígido, sino de uno manipulable, configurable, de fácil creación y reubicación.

1.1 Motivación

La gestión de redes actualmente se encuentra en constante cambio hacia un modelo de administración más flexible y dinámico, debido a que necesita adaptarse a los requerimientos vigentes de servicios. Ésta tendencia ha sido fomentada en los últimos años con la aparición de los conceptos de Virtualización de Redes y de Redes Programables.

Dichas tecnologías se están desarrollando con mira a incrementar la utilización de recursos de red, pero a su vez, reduciendo los costos de operación. Como tópicos emergentes, se están planteando distintos criterios de aplicación a diferentes entornos de red, por lo que existe la necesidad de realizar estudios de utilización en medios específicos y de ésta forma analizar su efectividad y rendimiento.

La idea sobre flexibilidad operacional de redes, particularmente NFV, genera un abanico de interrogantes concernientes a confiabilidad, rendimiento, seguridad, orquestación, lugares de funcionamiento, desempeño, etc. Del mismo modo sobre el qué se puede virtualizar o no y en donde se deberían localizar las funciones de red para optimizar el rendimiento de las redes.

Todas estas interrogantes son aplicables a los distintos escenarios de casos de uso de la tecnología NFV. En el presente proyecto, aplicando al entorno de redes móviles, se

analizarán los métodos, técnicas y especificaciones que nos permitirán dar solución a todas éstas interrogantes.

1.2 Objetivos

Como objetivo general se plantea *Analizar la virtualización de funciones de red aplicado a las redes móviles y su gestión mediante entorno OSM de MANO*, estudiando todas las posibilidades y métodos, en donde genere mayor eficiencia utilizar NFV, enfocándose de manera especial en el núcleo de red, donde se coordina la pasarela de datos, ingeniería de tráfico y direccionamiento. Así, de esta forma se pueden ponderar sus beneficios dentro de las actuales tecnologías de tratamientos de datos.

Para alcanzar el presente objetivo, se necesitan analizar de forma independiente aspectos específicos como:

- Estudiar el estado del arte sobre la arquitectura de redes móviles de nueva generación, su funcionamiento y despliegue actual. Determinar y ubicar sus funciones de red dentro de sus núcleos que podrían ser propuestas a una posible virtualización para mejorar su rendimiento y generar versatilidad a las operadoras.
- También se proyecta investigar el estado del arte sobre virtualización de funciones de red, sus antecedentes, desarrollo, proyecciones, alcance y aplicaciones actuales dentro de las comunicaciones. De éste modo analizar cómo dicha tecnología se la puede involucrar dentro de las redes móviles como alternativa a la solución de reducción de costos de implementación y así permitirle evolucionar hacia las siguientes generaciones en las telecomunicaciones.
- Para finalizar el proyecto se realizará un análisis de plataformas actuales que se encuentren en desarrollo que permitan crear, configurar y desplegar NFV, así como el método para que puedan ser implementadas por las operadoras móviles dentro de sus arquitecturas. Finalmente se plantea realizar un caso práctico en donde se pueda mostrar cómo es el funcionamiento de la gestión de una red virtualizada, mediante la plataforma OSM.

1.3 Estructura del Documento

El presente documento aporta una descripción total sobre la tecnología NFV, teniendo como eje principal, el análisis de los elementos de direccionamiento de datos que se encuentran en el núcleo de las redes móviles. Para lo cual se ha seccionado en 6 capítulos, que son detallados a continuación:

- **Capítulo 1 – Introducción:** Se describe un contexto actual sobre el uso de la telefonía móvil y como ha ido transformando su tecnología para abastecer la alta demanda de servicios generados por los usuarios. Así como los problemas que esto genera en el tratamiento de datos y posibles soluciones actuales.
- **Capítulo 2 – Estado del Arte LTE:** Se analizan los fundamentos teóricos y técnicos sobre las tecnologías de telefonía móvil LTE, con el fin de comprender su estructura y funcionamiento, proporcionando un panorama amplio de sus componentes potenciales a ser virtualizados en la actualidad, así como en el futuro.
- **Capítulo 3 – Estado del Arte NFV:** Se estudian las bases teóricas sobre las tecnologías de virtualización, sus procedimientos y aplicación en las redes. Además sus enfoques actuales sobre la virtualización de funciones de red reguladas y descritas por la ETSI, donde se detalla su arquitectura y servicios que brinda, como gestión y orquestación de las funciones de red. También se habla de sus beneficios que brindar hacer uso de esta tecnología sobre entornos de alta complejidad como las redes móviles. Se menciona también algunos casos de uso donde se puede aplicar NFV para obtener una mejor relación costo beneficio entorno a los negocios de las telecomunicaciones.
- **Capítulo 4 – NFV de EPC:** En éste capítulo se explican los métodos de virtualizar el núcleo de red móvil. También se analizan las distintas posibilidades que existen para aplicar NFV sobre los componentes de EPC y antes de finalizar el capítulo se describen distintos escenarios de comunicación con NFV sobre EPC. El capítulo termina con recomendaciones de implementación de NFV, sus retos y aplicaciones actuales dentro de su gestión y orquestación.
- **Capítulo 5 – Caso Práctico:** Para realizar un caso demostrativo, primero se analizan los diferentes proyectos desarrollados para gestión, orquestación y despliegue de VNFs. Se enfoca principalmente en entornos de Open Source, con el fin de tener facilidad en la adquisición, modificación e investigación de su funcionamiento así como de la arquitectura que utilizan para aplicar en cualquier entorno de comunicaciones computacionales disponibles.

También se propone el uso de la herramienta OSM, a la cual se la estudia en detalle para aplicar sus servicios a cualquier entorno de redes de comunicación.

- **Capítulo 6 - Conclusiones y Líneas futuras:** Finalmente en el capítulo se describen las conclusiones obtenidas del estudio del proyecto, así como la experiencia de uso de la herramienta OSM de gestión de VNFs. Para después presentar líneas futuras de investigación y desarrollo, tratando de direccionar el enfoque sobre solventar los retos actuales de las tecnologías NFV.
- **Bibliografía**
- **Anexos**

2 Sistemas de Comunicación Móvil

En el presente capítulo se abordará la base teórica de los distintos componentes tecnológicos que involucran la telefonía móvil de última generación, la misma que contiene información profunda sobre su estructura, así como de su funcionamiento. La cual nos servirá como pilar fundamental en el desarrollo del presente documento.

2.1 Redes Móviles de Telecomunicación

2.1.1 Introducción

Un sistema móvil hace referencia a todo enlace de comunicación entre distintos terminales que se encuentren, ya sea, en movimiento o estáticos pero en distintas locaciones. Dicho sistema permite a los usuarios utilizar los servicios que brindan las distintas operadoras de telecomunicaciones, que pueden ser interpersonales, donde los clientes son los que aportan y generan contenido de información dentro la red entre ellos, como llamadas de voz, mensajería de texto (SMS), etc. Mientras que los servicios no interpersonales son aquellos flujos de datos que son otorgados por terceros y el usuario hace uso de ellos, como por ejemplo, acceso a internet, aplicaciones, juegos, video streaming, etc.

En la actualidad las comunicaciones móviles son uno de los servicios de telecomunicaciones más utilizados, siendo los servicios de voz e internet los más activos. Todo gracias a la evolución tecnológica desplegada que extiende su servicio a grandes áreas de cobertura circular (celdas), incrementando en gran medida la reutilización de las mismas frecuencias en diferentes estaciones base y llegando cada vez hacia más usuarios.

Su funcionamiento se basa esencialmente en una estructura de comunicación que permite brindar servicios a los usuarios, que consiste en la combinación de una red de estaciones transmisoras y/o receptoras de radio (BTS), las cuales se interconectan con centrales telefónicas de conmutación (MSC y BSC), que son las responsables de la comunicación entre terminales móviles, portátiles y teléfonos de una red fija tradicional.

Una de las razones por lo que estos sistemas se han convertido en uno de los medios de comunicación más importante, es su capacidad de optimizar el uso del espectro

radioeléctrico, evitando interferencias y permitiendo a los operadores maximizar la capacidad y disponibilidad de sus servicios.

La evolución de las redes móviles se ha visto marcada debido a que las tecnologías inalámbricas han tenido gran acogida y a su vez desarrollo en las últimas décadas. En un principio revolucionó la forma en que se transmitían las señales de voz, por medio de dispositivos inalámbricos que podían estar en movimiento, convirtiéndose en una herramienta primordial para gente común y de negocios.

La primera generación 1G fue concebida únicamente para señales de voz y se caracterizaba por ser estrictamente analógica, con capacidad y calidad de enlaces bajos. Años más tarde llega 2G, con una diferencia notable por ser digital, la cual la caracterizaba, debido a que ya poseía protocolos de codificación más sofisticados. Además soportaba velocidades de información más altas pero aun sin transmisión de datos, en este punto es donde aparece la tecnología GSM (Global System for Mobile Communications) con su método de acceso CDMA (Code Division Multiple Access), ocupada por muchas operadoras hasta la actualidad. Años posteriores, se realizan mejoras a la tecnología y se incorpora GPRS (General Packet Radio System), el cual introduce conmutación de paquetes de las redes móviles GSM (Protocolo IP), apareciendo nuevos servicios de multimedia.

La era moderna dentro de la telefonía celular, se ha establecido desde la aparición de la 3G, caracterizada especialmente por la convergencia de señales de voz y acceso a internet móvil por medio de datos (IP), aumentando sus prestaciones en servicios compatibles con aplicaciones multimedia. Los nuevos protocolos implementados en los sistemas 3G, permiten tener altas velocidades de transmisión de información, yendo más allá del simple hecho de la comunicación analógica de voz. Éste sistema se enfoca ya en aplicaciones que admiten la transferencia de ficheros de audio, video, videoconferencia y acceso web. El estándar que lo soporta es UMTS (Universal Mobile Telecommunications System). [1]

En los últimos años, se han estado estableciendo diversas mejoras de los sistemas 3G, desarrollando estándares y métodos que aumenten sus velocidades de transmisión, así como su ancho de banda, de este modo se pasa a 3.5G, en donde predomina HSPA (High Speed Access Packet), definido en el estándar 3GPP¹ (3rd Generation Partnership Project), el cual establece tipos de modulación y velocidades tanto de bajada como de subida, mucho mayores que su generación predecesora. Pero en la actualidad nos encontramos en los comienzos de las redes de nueva generación, que

¹ <http://www.3gpp.org/technologies/keywords-acronyms/99-hspa>

Telecommunication - 2000 (IMT - 2000)², en donde se mencionan especificaciones como uso de espectros de frecuencia, servicios de conmutación de paquetes como tráfico de protocolo IP, y soporte de muchos servicios multimedia. Entonces 3GPP finalizó la primera versión de su sistema de comunicación móvil, que le precede a GSM, conocido como UMTS.

De este modo los operadores se han visto en la necesidad de dimensionar sus redes acoplando nuevas tecnologías que les permita satisfacer la demanda generada por el conjunto de usuarios del sistema. El cambio ha sido constante, el siguiente paso evolutivo dentro de la telefonía móvil se da en el 2004, cuando 3GPP³ inicia su proyecto en el siguiente sistema de móvil de última generación 4G, el cual lo nombra como tecnología LTE (Long Term Evolution) que permite a los usuarios a hacer uso de los servicios sin cortes y a alta velocidad, principalmente cuando éstos se encuentren en movimiento. En los siguientes puntos del capítulo se adentra con mayor detalle en la estructura de esta tecnología de última generación. LTE, es la de mayor importancia, debido a su evolución futura.

2.1.3 Antecedentes de la Evolución 3G

Para entender los complejos sistemas de comunicación 3G y sus evoluciones implementadas en la actualidad, es importante conocer de dónde vienen y como un sistema móvil ha evolucionado a gran velocidad, de ser una tecnología exclusiva ocupada por pocos miembros debido a su costo, hasta ser en la actualidad un sistema global de comunicación ocupado por más de la mitad de la población del mundo, llegando a ser considerado como un servicio básico más de las personas. Su desarrollo ha cambiado de brindar servicios regionales o nacionales, hasta permitir enlazar complejas conexiones internacionales, donde organizaciones globales como 3GPP⁴, han reunido esfuerzos para desarrollar un estándar global, que envuelva todo los sistemas de comunicaciones de celulares con el fin de interconectar a las millones de personas.

La Unión Internacional de Telecomunicaciones (ITU), inició el trabajo sobre la tercera generación de comunicaciones móviles, aproximadamente en los años 1980. Posteriormente, el nombre de 3G dentro de la ITU fue cambiado a IMT-2000, el cual es un marco de definición para el acceso inalámbrico mundial enlazando los diversos sistemas de redes terrestres y satelitales. Además aquí se especifican las interfaces y

² <https://www.itu.int/osg/spu/imt-2000/technology.html>

³ <http://www.3gpp.org/technologies/keywords-acronyms/98-lte>

⁴ <http://www.3gpp.org/>

espectros radioeléctricos en el que funcionaría 3G, así como el entorno de servicios, arquitectura de red, requerimientos de interfaces de radio y evaluación de la metodología.

Los criterios de evaluación del grupo sobre las tecnologías de IMT-2000, se enfocaban en que toda la arquitectura permita generar más altas velocidades en la transmisión de datos, combinando entre conmutación de circuitos y conmutación de paquetes. Al inicio se especificó que trabajaría en entornos estáticos, velocidades de hasta 2Mbps, pero en la actualidad dichas tasa han ido más allá de su capacidad y que pueden ser desplegadas dentro de sistemas 3G de igual forma. [2]

Las especificaciones dictadas por el organismo europeo 3GPP son de las más ampliamente desplegadas en el mundo, con billones de usuarios. Sus últimos enfoques han sido estudiar y desarrollar una evolución de 3G dentro de un tipo de radio de acceso evolucionado, al cual se lo refieren como LTE (Long-Term Evolution), el mismo que pertenece a una red de acceso igualmente evolucionada, se la llama SAE (*System Architecture Evolution*).

2.1.4 Evolución a UMTS

La ETSI (*European Telecommunications Standards Institute*), en los inicios del año 2000 hizo la selección del método de acceso a CDMA como tecnología para UMTS, con la mejora de ancho de banda (*Wideband - CDMA*), la cual permitió alinear y detener todas las especificaciones que se desarrollaban en paralelo por múltiples regiones.

Para conocer la evolución que ha tenido la telefonía móvil desde sus inicios hasta llegar a UMTS, se debe tomar en cuenta tres aspectos esenciales: técnicos, arquitectura de red y servicios. La parte técnica detalla los métodos de desarrollo de los elementos de red y las tecnologías en las que serían implementados, siempre tomando en cuenta la evolución de la red debido a los requerimientos de sus elementos. Por último los servicios, son los aspectos responsables de la convergencia y avance de las tecnologías móviles, ya que están fuertemente relacionados con la demanda de los usuarios, así como la disponibilidad actual de sus redes.

Dentro de los aspectos técnicos y de ingeniería, en los comienzos de GSM, la idea principal fue crear interfaces abiertas, que permitieron definir el estándar de los componentes del sistema GSM. La apertura de las interfaces brinda muchas ventajas de interconexión, permite que componentes de red de distintas operadoras formen parte de una misma red sin inconvenientes. Debido a la complejidad de las redes móviles por el hecho de la necesidad de cubrir grandes áreas geográficas, su sistema se lo

dividió en subsistemas con el fin de descentralizar su control y operación, de esta manera incrementaron su desempeño y disponibilidad. El sistema GSM define cuatro subsistemas, MS, BSS, NSS, y NMS, los mismos que se muestran en la Fig. 2.2, y un nuevo componente de conmutación de paquetes GPRS.

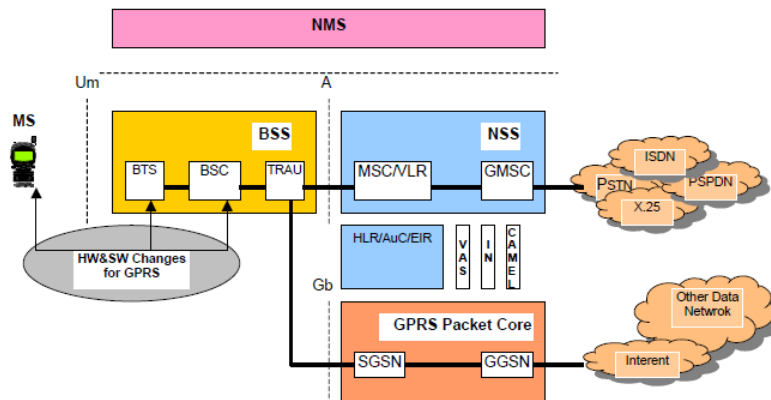


Figura 2.2. Arquitectura de Red GSM e introducción de GPRS.

Solos dos de las tres interfaces que componen la arquitectura GSM, fueron diseñadas como abiertas, la *Um* y *A*, la tercera que estaba comprendida entre NMS y NSS/BSS fue creada por propietarios específicos debido al retraso en sus especificaciones y estandarización.

El componente **MS** (Mobile System) de la arquitectura está formado por **ME** (Mobile Equipment) y **SIM** (Subscriber Identity Module), y es aquel que se registra a la red con la información del suscriptor. **BSS** (Base Station Subsystem), es el siguiente paso para la conexión hacia la red móvil y encargado de controlar todo el acceso inalámbrico de la red, ya sea de enlaces ascendentes o descendentes. Sus componentes son **BTS** (Base Transceiver Station) es aquel que interactúa directamente con el usuario mediante la interfaz *Um* y se encarga de señalización, cifrado y procesamiento de las señales de voz. **BSC** (Base Station Controller), como su nombre lo indica, es el núcleo de la red de acceso, aquel que reparte las frecuencias y controla las potencias de los terminales y estaciones base. Y el último componente es **TRAU** (Transcoder and Rate Adptation Unit), desempeña funciones de *trascodificación*⁵ y de adaptación de tasas de información en los canales de voz, en otras palabras se encarga de comprimir los datos de 64 Kbits/s a 13/12.2/6.5 kbits/s, y repartir dichas señales tratadas a los dos tipos de conmutadores, ya sea de circuitos o de datos **GPRS** (General Packet Radio Service).[3]

⁵ <https://en.wikipedia.org/wiki/Transcoding>

El siguiente paso es la interfaz *A* que interconecta a BSS con **NSS** (Network Switching Subsystem), el cual se encuentra integrado por varios componentes como MSC (Mobile Switching Center), GMSC (Gateway Mobile Switching Centre), AuC (Authentication Center), VLR y HLR (HLR base de datos de usuarios de una red y VLR base de datos temporal de los móviles de una región). Es el encargado de enlazar las comunicaciones entre los equipos móviles y la **PSTN** (Public Switched Telephone Network). En toda esta sección se controlan interconexiones, facturación, estadísticas, señalización, gestión de movilidad, seguridad de información, autenticación y conexiones con redes externas.

La sección de conmutación de paquetes está gobernada por GPRS, que introduce la utilización de IP en la red móvil. Reutiliza la infraestructura de radio de la red GSM, es decir los mismos enlaces de subida y bajada. Se encuentra compuesto por SGSN y GGSN, encargados de transportar los datos hacia el internet o hacia sus usuarios de red. También conocido como la pasarela de datos. Inicialmente GPRS permitía una velocidad de transmisión de 171.2 kb/s.

El último componente de red, **NMS** (Network Management System), es aquel que observa toda la red anterior, realizando funciones de operación y mantenimiento, además de controlar calidad de servicio.

2.2 Redes Móviles LTE

Define la cuarta generación en comunicaciones móviles. En telefonía móvil se aplica LTE, estandarizado por 3GPP⁶ (3G Partnership Project) en la Release 8, 9, 10, 11, y 13 siendo la más actual. A diferencia de UMTS, en LTE se especifican los medios de acceso OFDMA (Orthogonal Frequency Domain Multiple Access) para enlaces descendentes y SC-FDMA (Single Carrier Frequency Domain Multiple Access) para ascendentes, en lugar de WCDMA utilizada por UMTS. Logrando con esto alcanzar velocidades pico de transmisión de 100 Mbps hasta 1 Gbps (LTE - Advanced) en downlink y 50 Mbps en uplink.

Las redes móviles 4G están diseñadas para brindar servicios basados todo en IP, como transmisión de voz, contenido multimedia, etc., a altas tasas de transmisión con soporte de movilidad y calidad de servicio. Telecomunicaciones Móviles Internacionales - Avanzadas (IMT-Avanzadas) es un concepto para sistemas móviles con capacidades más allá de las IMT-2000. En teoría se prevé que las nuevas capacidades de estos sistemas IMT avanzados puedan soportar una amplia gama de

⁶ <http://www.3gpp.org/dynareport/36-series.htm>

velocidades de datos en entornos multiusuario con velocidades de datos pico de destino de hasta aproximadamente 100 Mbps para altos requerimientos de movilidad y hasta 1 Gbps para los requisitos de baja movilidad, como el acceso inalámbrico local. Al contenido y trabajo de IMT-Avanzadas dentro de 3GPP se lo conoce como LTE-Advanced (LTE-A).[4]

A continuación se detalla un resumen de las especificaciones que marcaron la evolución de LTE a LTE-A.

Release 8

En este documento se adopta la modulación OFDMA para el acceso múltiple. Además, la celda puede utilizar un canal con mayor ancho de banda principalmente en el enlace descendente. Mediante el uso conjunto de técnicas de DC y 64QAM se puede lograr una capacidad del enlace descendente de hasta 84 Mbps.

Release 9

Esta versión introduce a la arquitectura de redes las femtoceldas, denotados como Home eNodeB (HeNB). Además, se da soporte a los servicios multicast o broadcast multimedia MBMS (Multimedia Broadcast Multicast Services) y se introducen mejoras en los servicios basados en la localización, LBS (Location Based Services).

Release 10

Esta versión es la primera norma 3GPP compatible con 4G. Se introducen nuevas técnicas como la agregación de portadoras, mejoras en enlace descendente MIMO y en el enlace ascendente MIMO. Asimismo se definen técnicas avanzadas para la coordinación de la interferencia que permiten la implementación densa de femtoceldas.

Release 11

Esta versión está todavía en desarrollo. Esta nueva versión incrementará la agregación de portadoras e introducirá la comunicación multipunto cooperativa Cooperative MultiPoint Communication (CoMP), aumenta la eficiencia espectral, y mejora la eficiencia energética.

Release 12

Uno de los requisitos fuertes era la necesidad de apoyar el rápido aumento en el uso de datos móviles, además de la necesidad de soporte eficiente de diversas aplicaciones que garanticen al mismo tiempo una experiencia de usuario de alta calidad. Algunos de las mejoras a la versión 12 incluyen:

- Células pequeñas mejoradas para LTE, la introducción de una serie de características para mejorar el apoyo de HetNets⁷
- Entre sitios se realiza agregación de portadoras, para mezclar y combinar las capacidades y backhaul⁸ de las células adyacentes
- Nuevas técnicas de antenas y receptores avanzados para maximizar el potencial de las células más grandes
- Interfuncionamiento entre LTE y Wi-Fi o HSPDA

2.3 Arquitectura de Red LTE

En esta sección se detalla la arquitectura de la red de comunicaciones móviles basada en las especificaciones del sistema LTE mencionadas en los puntos anteriores. Para ello, se realizó un análisis inicial de la arquitectura genérica adoptada en los sistemas celulares 2G/3G actuales, se identifican las piezas fundamentales que componen la arquitectura de red de toda la familia de sistemas especificada por 3GPP (GSM, UMTS, LTE).

A diferencia de las redes anteriores GSM y UMTS, la arquitectura LTE está diseñada solo para soportar conmutación de paquetes. Está compuesta esencialmente por 3 secciones, equipo de usuario, red de acceso E-UTRAN (Evolved - UTRAN), y el núcleo de red (Red Troncal) EPC (Evolved Packet Core). El conjunto de estos dominios se lo conoce como EPS (Evolved Packet System), y por otro, la evolución del subsistema IMS concebido inicialmente en el contexto de los sistemas UMTS. Los diferentes componentes han sido diseñados para soportar todo tipo de servicios de telecomunicación mediante mecanismos de conmutación de paquetes (All - IP), por lo que no resulta indispensable disponer de un componente adicional para la provisión de servicios en estilo de conmutación de circuitos, como el caso de las anteriores generaciones de 3GPP. Por lo tanto el nuevo núcleo de red de LTE, EPC constituye una versión evolucionada del núcleo GPRS.

En la siguiente figura también se muestra la división de la arquitectura en sus cuatro principales dominios de alto nivel: Equipo de Usuario (UE), Evolved UTRAN (E-UTRAN), Evolved Packet Core Network (EPC), y el campo de los servicios (IMS). La estructura base de los dominios de arquitectura de alto nivel son funcionalmente equivalentes a las de los sistemas 3GPP existentes. Las nuevas mejoras desarrolladas dentro de su arquitectura se centran únicamente a la red de Acceso por Radio y a las

⁷ <http://www.3gpp.org/hetnet>

⁸ <https://es.wikipedia.org/wiki/Backhaul>

Redes Centrales, E-UTRAN y EPC, respectivamente. Mientras que UE y IMS permanecen intactos arquitectónicamente, como la anterior generación de comunicaciones.[4]

En la Figura 2.3 se muestran las principales interfaces de E-UTRAN y EPC. La interfaz entre E-UTRAN y EPC se denomina S1 y proporciona a la EPC los mecanismos necesarios para gestionar el acceso de los terminales móviles a través de E-UTRAN. La interfaz radio entre los equipos de usuario y E-UTRAN se denomina E-UTRAN Uu. Por otro lado, las plataformas de servicios como IMS y la conexión a redes de paquetes externas IP se llevan a cabo mediante la interfaz SGI de la EPC. La interfaz SGI es equivalente a la interfaz Gi definida en las redes GPRS/UMTS y constituye el punto de entrada/salida al servicio de conectividad IP proporcionado por la red LTE (los terminales conectados a la red LTE son “visibles” a las redes externas a través de esta interfaz mediante su dirección IP).

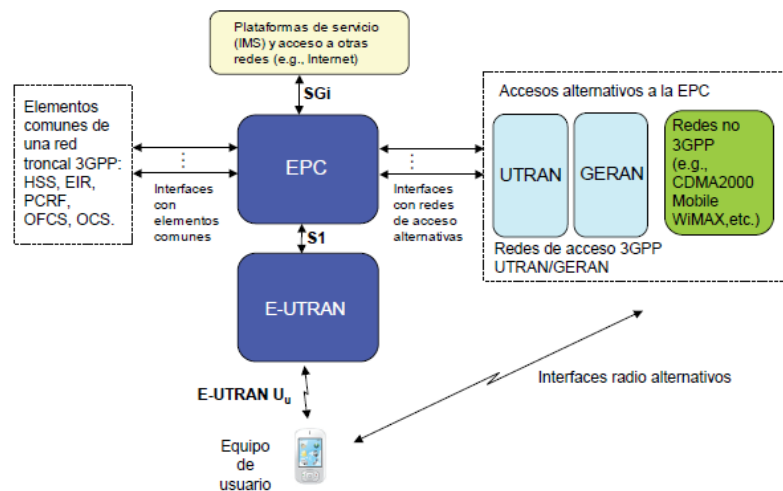


Figura 2.3. Arquitectura de Red LTE. [5]

2.3.1 Arquitectura E-UTRAN

El único nodo en la E-UTRAN es el eNodo B. Al igual que en los sistemas anteriores GSM/UMTS, el eNodoB es una estación base de radio que tiene el control de todas las funciones relacionadas con el acceso por radio del sistema. Las estaciones base, como el eNodoB se distribuyen normalmente a lo largo del área de cobertura de redes, cada una de ellas se las coloca cerca de las antenas de radio reales. Además a diferencia de las redes de acceso de GSM y UMTS compuestas por estaciones base (BTS, Nodob) y equipos controladores (BSC y RNC), E-UTRAN integra toda la funcionalidad de la red de acceso. [5]

Funcionalmente eNodoB actúa como un puente de capa 2 entre el UE y el EPC, por ser el punto de terminación de todos los protocolos de radio hacia el UE, y de transmisión de datos entre la conexión de radio y la conectividad basada en IP correspondiente hacia al núcleo EPC. En este punto, el eNodoB realiza procesos de cifrado/descifrado de los datos, y también la compresión/descompresión de cabeceras IP, lo cual significa evitar en repetidas ocasiones el envío de las mismas secuencias de cabeceras IP. Los eNodeB están interconectados entre sí por medio de la interfaz X2, y del mismo modo se comunican con el núcleo de red por la interfaz S1, la cual se subdivide en dos interfaces diferentes: S1-MME para sustentar el plano de control y S1-U como soporte del plano de usuario.

Como se explicó en los puntos anteriores, la separación entre plano de control y plano de usuario es una característica importante en la organización de las torres de protocolos asociadas a las interfaces de la red LTE. La descripción de la arquitectura de E-UTRAN se detalla en las especificaciones del 3GPP TS 36.300 [6]

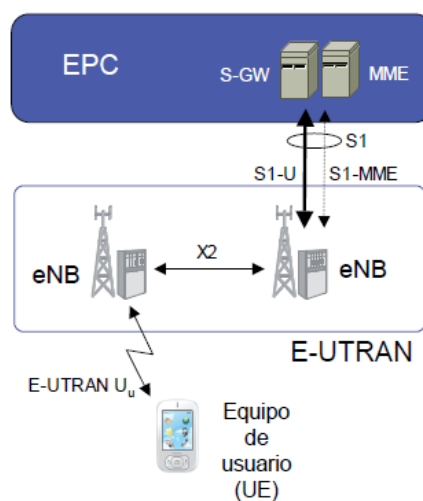


Figura 2.4. Red de Acceso E-UTRAN [5]

2.3.2 Arquitectura de EPC

El nuevo núcleo de red ha sido concebido para proporcionar un servicio más integral y avanzado, pensando sobre todo en la convergencia de las comunicaciones, por lo tanto está creado para que soporte funcionalidades "all-IP", es decir conectividad IP. La descripción completa de la red troncal EPC se recoge en los documentos 3GPP TS 23.401 [7]. En la especificación mencionada se cubre la arquitectura de la red troncal EPC cuando la red de acceso es E-UTRAN, así como la utilización de redes de acceso 3GPP alternativas o complementarias como UTRAN y

GERAN. Por otro lado, extiende la arquitectura de la red troncal EPC para soportar el acceso a través de otras redes no 3GPP.

De cara a introducir los diferentes componentes de la arquitectura completa de EPC de forma progresiva, la arquitectura mostrada en la Figura 2.5 comprende únicamente las entidades de red que forman el núcleo de la red troncal EPC para la provisión de servicios de conectividad IP a través de una red de acceso E-UTRAN, junto con las entidades de red e interfaces que soportan las funciones relacionadas con el control de del servicio de conectividad (e.g., control de QoS) y de los mecanismos de tarificación.

El núcleo de la red troncal EPC está formado por tres entidades de red principales, MME (Mobility Management Entity), Serving Gateway (S-GW) y Packet Data Network Gateway (P-GW), que, junto a la base de datos principal del sistema denominada HSS (Home Subscriber Server), constituyen los elementos principales para la prestación del servicio de conectividad IP entre los equipos de usuario conectados al sistema a través de la red de acceso E-UTRAN y redes externas a las que se conecta la red troncal EPC.

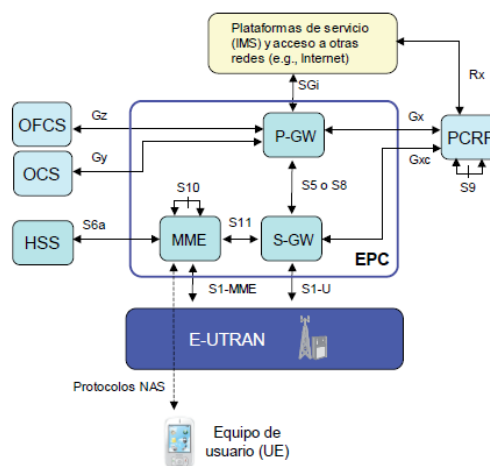


Figura 2.5. Arquitectura de Núcleo de Red EPC[5]

2.3.3 Serving Gateway (S-GW)

El componente S-GW direcciona y envía los paquetes de datos de usuario, mientras que también actúa como el eje de movilidad para el plano de usuario durante las transferencias de información entre eNodeB y así como también para la movilidad entre LTE y otras tecnologías 3GPP. En la configuración de la arquitectura básica del sistema, la función de alto nivel de S-GW es de gestión de túneles y de conmutación en el plano de usuario. También es parte de la infraestructura de red mantenida en las centrales de operación.

Las interfaces S5/S8 se basan en la técnica de túnel GTP. S-GW mapea entre los flujos de servicios IP y túneles de GTP que se realizan en P-GW, por lo tanto S-GW está conectado a PCRF, pero no lo necesita. Todo el control está relacionado con los túneles del protocolo GTP, y viene ya sea de MME o P-GW. Cuando la interfaz S5/S8 utiliza PMIP (Proxy Mobile IP)[8], S-GW lleva a cabo el mapeo entre el servicio IP que fluye en S5/S8 y los túneles GTP en la interfaz S1-U, de este modo se conectará a PCRF para recibir la información de asignación.

El S-GW tiene un papel muy secundario en funciones de control. Sólo es responsable de sus propios recursos, y les asigna en base a las peticiones de MME, P-GW o PCRF, que a su vez tiene en cuenta la necesidad de establecer, modificar portadora limpias para el UE. En la figura 2.6 se muestra la estructura funcional y de comunicación del nodo S-GW.

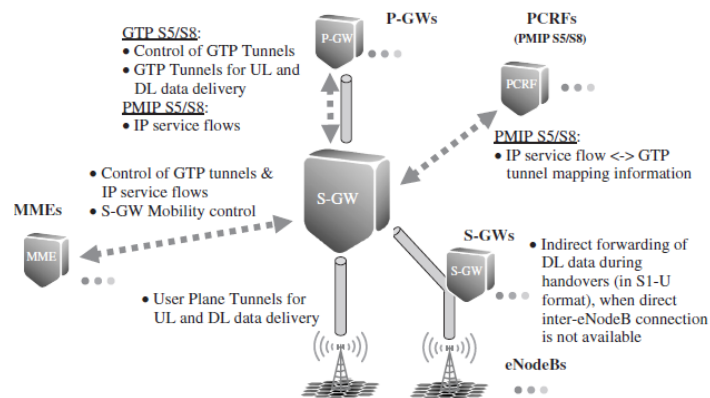


Figura 2.6. Funciones Principales y Conexiones Lógicas de S-GW [4]

2.3.4 PDN Gateway (P-GW)

La PDN-GW proporciona conectividad al UE a las redes de paquetes de datos externos al ser el punto de salida y entrada de tráfico para el usuario. Un UE puede tener conectividad simultánea con más de una PDN-GW para acceder a múltiples redes de datos de paquetes. El PDN GW-lleva a cabo la aplicación de políticas, filtrado de paquetes para cada usuario, soporte de la carga, interceptación legal, y la detección de paquetes. Otra función principal del PDN-GW es actuar como el ancla para la movilidad entre 3GPP y tecnologías no-3GPP tales como WiMAX y 3GPP2 (CDMA 1x y EV-DO)[5]. Al igual que S-GW es localizado en las centrales de las operadoras.

El P-GW aplica DHCP cuya funcionalidad es requerida para consultas de un servidor DHCP externo, y este poder entregar la dirección de un UE. En función de la necesidad puede asignar direcciones IPv4 o IPv6, y es el UE quien puede indicar si

desea recibir la dirección (es) en el proceso de señalización, o después de que se haya conectado a la capa de enlace.[9]

El tráfico entre P-GW y redes externas es en forma de paquetes IP que pertenecen a diferentes flujos de servicios IP. Si la interfaz S5/S8 con dirección a S-GW se basa en GTP, por tanto P-GW lleva a cabo la asignación de los datos IP a través de túneles GTP. Por otro lado P-GW establece y asigna portadoras, según la petición, ya sea a través de PCRF o desde la S-GW, la transmite la información de la MME. La figura 2.7 muestra las conexiones lógicas de P-GW, y las funciones principales de estas interfaces.

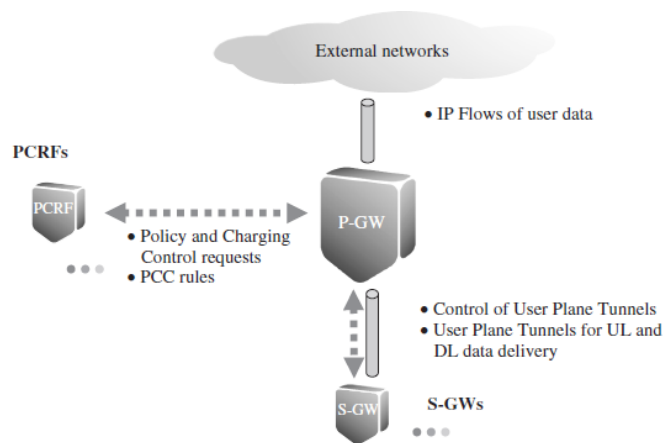


Figura 2.7. Funciones Principales y Conexiones Lógicas de P-GW [4]

2.3.5 Arquitectura de Protocolos de Comunicación a través de LTE

En esta sección se detallan las torres de protocolos utilizadas en las interfaces descritas en los puntos anteriores. Debido a que muchas de las interfaces están basadas en una misma torre de protocolos, la descripción se realiza en base a los diferentes protocolos utilizados.

La comunicación entre los distintos niveles de infraestructura de red, se realiza en base a los protocolos que se muestran en la figuras 2.8 y 2.9. Los protocolos en las capas de aplicación, transporte y red son los mismos que en UMTS, se maneja TCP/UDP e IP respectivamente. Para que un terminal se comuniquen con un eNB lo hace por medio de los siguientes protocolos, los cuales le permiten ingresar al núcleo de la red LTE.

Protocolo PDCP (Packet Data Convergence Protocol), es el responsable de comprimir la cabecera IP, disminuyendo la sobrecarga y mejorando la eficiencia en las interfaces de radio, además aplica funciones de cifrado y protección. A continuación se encuentra el protocolo RLC (Radio Link Control), es el responsable de la segmentación y concatenación de los paquetes PDCP. Garantiza retransmisiones y entrega de

paquetes a capas más altas, además desempeña funciones de corrección de errores mediante el método ARQ (Automatic Repeat Request). [6]

En la siguiente capa de comunicación se encuentra MAC (Medium Access Channel), encargado de programar los recursos de uplink y downlink. Garantiza calidad de servicio sobre la interface de aire. Finalmente la capa física (PHY) es la responsable de todos los temas sobre señales de radio, modulación/demodulación, codificación y decodificación, aparte de otras funciones como MIMO (Multi Input Multi Output).

Al igual que en las tecnologías anteriores GSM/UMTS, el núcleo de red de LTE maneja dos pilas de protocolos, tanto para el plano de usuario, como para el plano de control. Dentro de los componentes de EPC, como se explicó anteriormente, se comunican mediante distintos protocolos, siendo los más importantes GTP, PMIP y NAS. [9]

- **GTP-U (Plano Usuario):** Proporciona un mecanismo de encapsulado para el envío de paquetes de usuario (paquetes IP del usuario) entre nodos de una red IP. Los paquetes que corresponden a un mismo servicio portador EPS se transportan con un identificador de túnel único denominado TEID (Tunnel Endpoint Identifier). El establecimiento de un túnel GTP-U consiste básicamente en la elección del identificador TEID asociado a un determinado servicio portador EPS en ambos extremos del túnel. La señalización necesaria para establecer el túnel se realiza mediante otros protocolos como GTP-C o S1-MME.
- **GTP - C (Plano Control):** Gestiona sesiones a través de los mensajes y procedimientos de señalización especificados para GTP-C. La red se encarga de la creación de túneles GTP-U entre las entidades de la red por donde transcurre el plano de usuario. Dichos túneles forman parte de la propia gestión de sesiones en la red, mediante el establecimiento, mantenimiento/actualización y liberación de conexiones PDN. El protocolo GTP-C se utiliza en las interfaces S3, S4, S5/S8, S10, S11 y S16.
- **PMIP:** Es el protocolo alternativo de GTP para la interfaz S5 / S8. Se ocupa de la gestión de la movilidad a nivel de capa de red IP, pero no incluye funciones de gestión. Todo el tráfico que pertenece a la conexión de un equipo de usuario a un PDN particular, se maneja en conjunto. PMIP se ejecuta directamente en la parte superior de IP, y se lo utiliza como el transporte IP estándar.

- NAS (Non Access Stratum):** Es un protocolo desarrollado por el 3GPP, el cual lleva a cabo la gestión de movilidad de los equipos de usuario (EPS Mobility Management, EPM) y la gestión de las sesiones para el establecimiento de la conectividad entre el equipo de usuario y la pasarela P-GW (EPS Session Management, ESM) . Además se soporta entre el UE y un nodo MME por ser desarrollado específicamente para E-UTRAN.

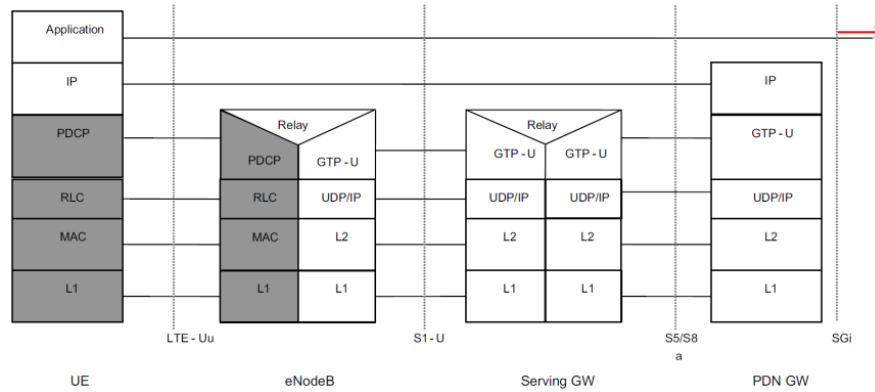


Figura 2.8. Pila de Protocolos de Plano de Usuario en LTE. [4]

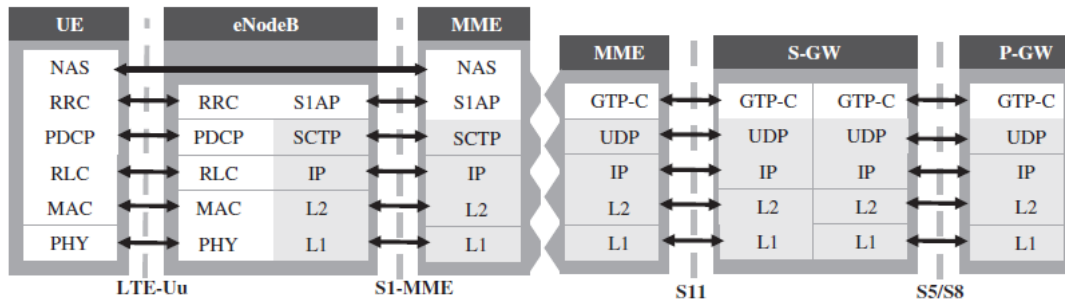


Figura 2.9. Pila de Protocolos de Plano de Control en LTE.[4]

3 Estudio de la Tecnología NFV

3.1 Virtualización

Al proceso de crear versiones virtuales (software) de recursos físicos específicos que emulan las mismas características físicas del hardware, se lo conoce como virtualización. Es comúnmente utilizada en el ámbito de la Tecnología de la Información (IT), con el fin de dividir un recurso físico en varios virtuales, por ejemplo, la memoria virtual, partición del disco duro y la más práctica máquina virtual. El concepto de virtualización [10] fue descrito por primera vez por Christopher Strachey, en el documento titulado "Time Sharing in Large Fast Computers". Más adelante IBM introdujo el término de la máquina virtual (VM) por primera vez, a mediados de los años 60's. La idea inicial fue crear varias máquinas virtuales de una computadora central para permitir correr varios procesos y aplicaciones simultáneas en un ordenador, pensando en evitar gastos en adquisición de nuevos ordenadores costosos.

Dentro de todas las áreas donde se puede aplicar el concepto de virtualización, resaltan tres casos, que son los más utilizados:

- *Virtualización de Almacenamiento:* Consiste en crear centro de almacenamiento que posee un conjunto de dispositivos comunes, los cuales son vistos por el sistema operativo servidor como un único dispositivo de almacenamiento, que puede ser gestionado por una vista única.
- *Virtualización de Servidor:* Es el proceso relacionado a esconder los recursos físicos de un servidor y dividir su software en varios números de nuevos servidores que comparten un recurso físico en común. Los mismos que son vistos por el Sistema Operativo Central como un hardware individual.
- *Virtualización de Redes:* Éste concepto es el de los más actuales, en donde se crean un conjunto de recursos de red formando una red virtual. Permite funcionar y coexistir distintas redes virtuales en una forma aislada sobre una infraestructura común.

En el proceso de virtualización de servidores, el administrador del servidor utiliza un software especial para crear estas instancias virtuales aisladas. Hay tres maneras diferentes que se utilizan para virtualizar un servidor: virtualización completa, para-virtualización y la virtualización del sistema operativo (OS virtualización).

Virtualización Completa: Se requiere una capa de software especial que se encuentra en la parte superior del hardware físico e interactúa con los recursos físicos

de la máquina. A esta capa se la conoce como "hipervisor", o también se la llama como "gestor de máquina virtual (VMM)". El hipervisor proporciona la plataforma necesaria para las máquinas virtuales huésped para poder operar en la parte superior.

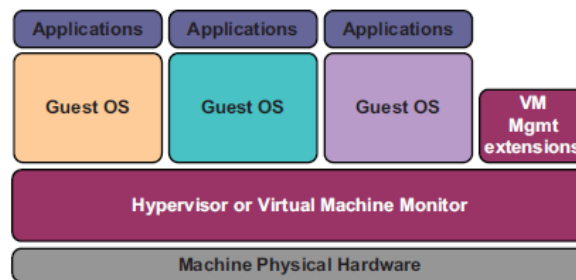


Figura 3.10. Pila de Protocolos de Plano de Control en LTE.[11]

Para Virtualización: Un hipervisor también se utiliza, pero, además, modifica el sistema operativo huésped para soportar hiper llamadas sobre funciones nativas. Esto significa que las máquinas virtuales se ejecutan como sistema operativo huésped modificado. En esta técnica las máquinas virtuales son conscientes el uno del otro.

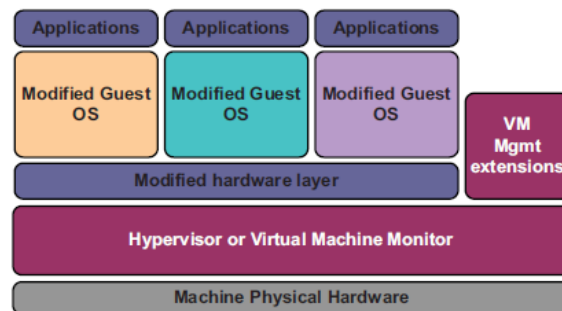


Figura 3.11. Pila de Protocolos de Plano de Control en LTE.[11]

Virtualización de Sistema Operativo: No se utiliza hipervisor en este caso, las funcionalidades de virtualización están soportados dentro del sistema operativo anfitrión. Los servidores invitados deben utilizar el mismo sistema operativo que el host, lo cual puede ser visto como un inconveniente de este enfoque.

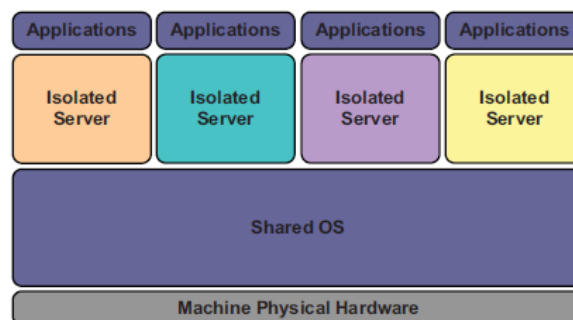


Figura 3.12. Pila de Protocolos de Plano de Control en LTE. [11]

3.2 Virtualización de Redes

Los operadores de red constantemente se encuentran acumulando una amplia y creciente variedad de dispositivos de hardware propietario. Si se desea aplicar un nuevo servicio de red a menudo requiere dos aspectos básicos, variedad de equipos nuevos y encontrar el espacio suficiente para poder darlos cabida, lo cual se está convirtiendo en algo cada vez más difícil; agravada por los crecientes costos de la energía, capacidad de inversión de capital y la complejidad de diseñar, integrar y operar aparatos más sofisticados, basados en hardware propietario.

Por otra parte, los ciclos de vida de hardware son cada vez más cortos ya que la tecnología y los servicios de innovación se aceleran con más rapidez, inhibiendo el despliegue de nuevos servicios de red y la innovación, en un mundo cada vez más conectado. El enfoque actual de la virtualización de redes es facilitar la transformación de la industria, desarrollando un ecosistema abierto e interoperable por medio de especificaciones, implementaciones y experiencias de uso.

Frente al rápido crecimiento sobre la demanda de banda ancha en redes móviles, los operadores móviles para mantener la eficiencia y capacidad, constantemente se encuentran adaptando sus infraestructuras de red a nuevas alternativas que les representen buena adaptabilidad, rendimiento y disminución de la complejidad de la red, así como sus costos. Por tal motivo se introduce el concepto, no tan actual, de virtualización de funciones de red. El cual se lo enfocará en este caso particular, a las redes móviles LTE.

La virtualización de funciones red (NFV), dentro de los operadores móviles, busca esencialmente evitar el despliegue de nuevos equipos dedicados y de hardware propietario, cada vez que necesita escalar. La centralización de varias de sus funciones bajo una máquina virtual optimizará su despliegue y servicios.

3.3 Virtualización de Funciones de Red (NFV)

NFV fue diseñado con el propósito de desacoplar las funciones de red, de los equipamientos hardware de red. Dicho de otra forma, el desacoplamiento permite colocar software, que realiza funciones específicas de un dispositivo, en una máquina diferente que el dispositivo original[12]. Por ejemplo, para evitar instalar un complejo equipo de red, que haga la función de controlador de sesiones de comunicación, se podría desplegar su virtualización en un centro de conjunto de servidores de múltiple propósito.

Aplicando el mismo concepto de la virtualización de un servidor clásico que ejecuta diferentes sistemas operativos, software y procesos, independientemente. Implementar funciones de red virtualizadas en centros de datos, posibilita la migración transparente entre máquinas virtuales o máquinas reales, permitiendo una mayor flexibilidad en lo que se refiere a gestión, asignación y escalado de recursos. Esto ayudaría a evitar el exceso de aprovisionamiento de equipos dentro del ecosistema de las redes, aumentando la eficiencia energética, y gasto en mantenimiento y operación de los equipos.

La ETSI (European Telecom Standards Institute), en su White Paper sobre NFV, titulado Network Functions Virtualization – Introductory White Paper[13], describe la arquitectura que debe tener una plataforma NFV para ser desplegada en entornos de multi-proveedores. Una NFV es una implementación de software que realiza una función de red capaz de ser ejecutado en un entorno de Infraestructura Virtual. En dicha infraestructura se describen diferentes capas de abstracción, permitiendo desacoplar a NFVs del hardware donde se lo implementa. A continuación se muestra la figura de la visión general de la virtualización de funciones de red.

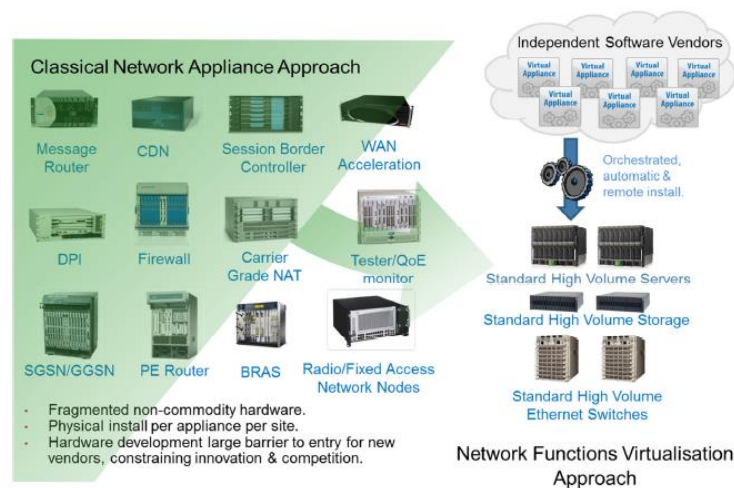


Figura 3.13. ETSI visión de NFV. [14]

Dentro del contexto de NFV, la virtualización de funciones es utilizada por los operadores de telefonía móvil para entender que nodos de red y aplicaciones son necesarias para la construcción de servicios de extremo a extremo con tecnología NFV. Las funciones de red virtualizadas están desplegadas y operadas por equipos físicos o más recursos informáticos estándar, redes y almacenamiento. Al conjunto de funciones de red virtualizadas desplegadas y operadas se los conoce como recursos de infraestructura descritos por el ETSI como Infraestructura NFV (NFVI). Es decir una NFV es capaz de ejecutarse en la NFVI. La infraestructura NFVI incluye una capa de virtualización que abstrae el hardware para soportar el desacoplamiento de las NFVs

del hardware. En la capa de virtualización se gestiona el entorno de ejecución (cómputo virtualizado, almacenamiento y redes) esto quiere decir que les permite crear a las NFVs instancias dinámicas de los recursos físicos. En la práctica es común aplicar la capa de virtualización con un hipervisor⁹ y un sistema operativo invitado (OS) para gestionar a las NFVs. [15]

En la arquitectura existe una sección muy importante al momento de desplegar las NFVs, se trata de la orquestación de los recursos NFVI en conjunto con las NFVs. La orquestación de estos recursos puede combinar y optimizar las funciones de red debido a los usuarios, flujo de tráfico, u otra política definida dentro de la red. Servicios muy a menudo como traducción de direcciones de red (NAT), cortafuegos e inspección profunda de paquetes (DPI) se aplican a una sesión de datos en secuencia.

3.3.1 NFV ISG (Grupo de Especificación de la Industria)

La estandarización de la tecnología NFV está liderada por ETSI en conjunto con varias operadoras de telecomunicaciones (AT&T, BT, Deutsche Telekom, Orange, Telecom Italia, Telefónica, Verizon, entre otras), proveedores de tecnologías de la información y proveedores de infraestructura de telecomunicaciones. A este grupo se lo conoce como ISG (Industry Specification Group), cuyo objetivo no es la producción de estándares, sino conseguir el consenso de la industria en los requerimientos técnicos y de negocio de NFV, y en acordar enfoques comunes para alcanzar estos requerimientos bajo una arquitectura común. Los resultados son publicados de forma abierta y compartidos con los principales grupos de estandarización, foros industriales y consorcios, como IETF¹⁰.

Dentro de las especificaciones logradas, se lanzaron varios documentos en donde se explican con más detalla sus distintos componentes y estructura. Entre los que se destacan por su importancia son: Entorno de arquitectura, Infraestructura (NFVI), Gestión, Orquestación (MANO) y Casos de Uso. [14]

3.3.2 Marco Arquitectónico NFV

La especificación de NFV ISG sobre la arquitectura de NFV es una de las más importantes, ya que proporciona la base para el futuro del ecosistema NFV de toda la industria de la comunidad de operadores. Define los diferentes bloques funcionales de su estructura, además delinea aspectos y puntos de referencia entre ellos. Además se

⁹ <https://es.wikipedia.org/wiki/Hipervisor>

¹⁰ <http://www.etsi.org/nfv>

especifican explícitamente ciertos componentes básicos funcionales que pueden ser proporcionados por los diferentes actores de la industria, preparando así el camino para una solución totalmente interoperable multipartidista y multiplataforma NFV.

La arquitectura general de NFV se muestra en la Figura 3.14. Ilustra los distintos bloques donde los proveedores pueden optar por aplicar software con el fin de ofrecer "productos compatibles NFV". Este marco arquitectónico está siendo utilizado para guiar el trabajo futuro en los diferentes grupos de trabajo NFV ISG, donde se analizaron adicionalmente los detalles de cada uno de los bloques de construcción con más profundidad. También se informa la base de un análisis del alcance de los componentes para priorizar el trabajo en curso dentro del ISG NFV. De igual forma describe cuál debería ser la estrategia de interacción con otros organismos que tengan interés en los bloques funcionales y las interfaces entre ellos.

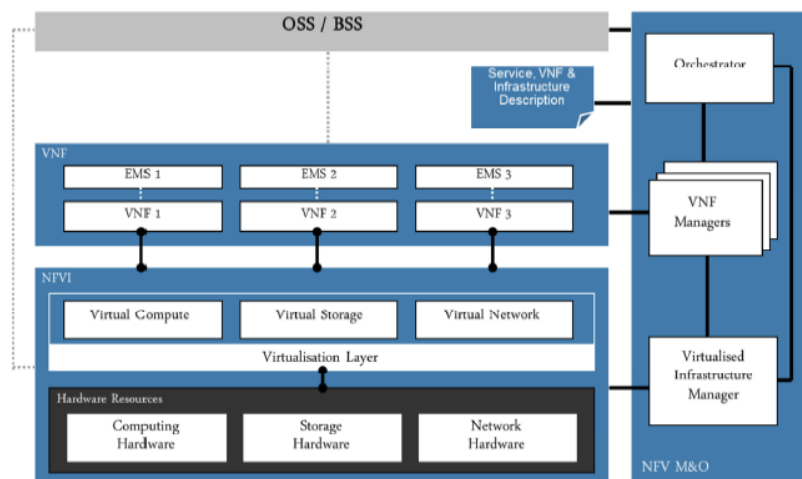


Figura 3.14. Marco Arquitectónico de tecnología NFV. [14]

3.3.3 Infraestructura de NFV

La infraestructura de NFV (NFVI) es necesaria para apoyar la gama de casos de uso y campos de aplicación ya identificadas por el NFV ISG, a la vez que proporciona una plataforma estable para la evolución del ecosistema NFV, con el fin de soportar los requerimientos de las comunicaciones futuras. El NFVI es el agregado total de los componentes de hardware y software que se acumulan en el entorno en el que son desplegadas las NFVs. La NFVI proporciona una infraestructura de múltiples usuarios aprovechando la tecnología de virtualización de TI estándar que pueden soportar múltiples casos de uso y campos de aplicación de forma simultánea, como se muestra en la figura 3.15. [16]

La NFVI se implementa como un conjunto distribuido de nodos desplegados en varios puntos de presencia (NFVI) que son requeridos para apoyar los objetivos de localidad y latencia de los diferentes casos de uso, así como de campos de aplicación. NFVs pueden desplegarse de forma dinámica en el NFVI en función de demanda dentro de los límites de capacidad de los nodos.

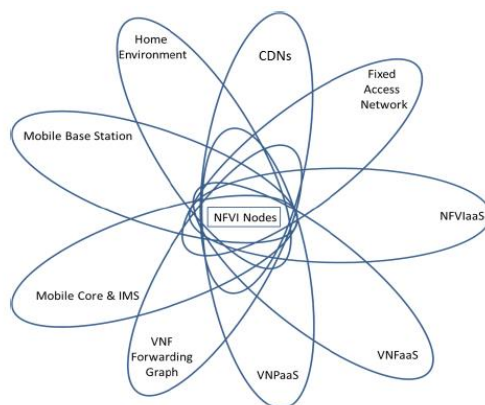


Figura 3.15. NFVI Multiplicidad y Campos de Aplicación NFV [16]

3.3.4 NFV MANO (Gestión y Orquestación)

La característica principal de NFV es desacoplar implementaciones de software de las funciones de cómputo, almacenamiento y recursos de red a través de una capa de virtualización. Por lo tanto este desacoplamiento requiere un nuevo conjunto de funciones de gestión y orquestación, crea nuevas dependencias entre ellas, por lo que requiere de interfaces estandarizadas, interoperables y de modelos comunes para la asignación de información.

MANO abarca la gestión del ciclo de vida y la orquestación de los recursos tanto físicos como software que soportan la virtualización de la infraestructura. Por otro lado la gestión y orquestación de NFV se enfoca en las tareas de la administración de la virtualización específica en el marco NFV. MANO también interactúa con (NFV externa) OSS/BSS¹¹, la cual permite a la NFV ser integrada en un entorno de gestión más amplio de toda la red ya existente.

El gestor al ser una sección importante posee algunas capas de VNF:

- La gestión de la infraestructura NFV es proporcionado por cualquier infraestructura como servicio (IaaS), es utilizado por cualquier operador o empresa, los cuales pueden aprovechar las APIs proporcionadas por

¹¹ <http://www.ossline.com/2010/12/definition-oss-bss.html>

aplicaciones como Amazon AWS, OpenStack, o entornos propios de IaaS de alguna operadora.

- El servicio de orquestación realiza el encadenamiento lógico de los elementos VNF a través de la entrada de las plataformas BSS/OSS con el Administrador de VNF para proporcionar un servicio de extremo a extremo.
- La aplicación de gestión y la interacción con los módulos de orquestación son proporcionados por el gestor de VNF. Para gestionar las aplicaciones NFV, con capacidades de EMS (Element Managers System), el gestor de VNF proporcionaría la funcionalidad para cada VNF según corresponda:
 - Implementación de servidores
 - Gestión de los recursos del servidor
 - La instalación y gestión de VNFs
 - Conexiones de facturación y acceso con IaaS/PaaS para que las aplicaciones se cargue y autentiquen correctamente.

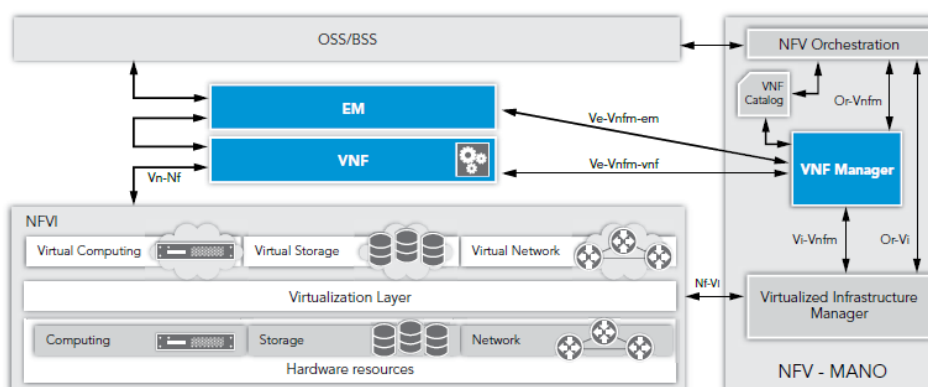


Figura 3.16. Arquitectura de Gestión y Orquestación desarrollada por NFV. [17]

3.3.5 NFV Casos de Uso

En el documento de casos de uso NFV [18], la agrupación NFV ISG describe los campos de aplicación que abarcan todo el ámbito de trabajo del ISG. Analiza los posibles casos potenciales en donde NFV podría ser un gran avance para las nuevas tecnologías de telecomunicaciones como 5G.

Dentro de los campos aplicables que se describen son por ejemplo, virtualización del núcleo de las redes móviles e IMS, virtualización de estaciones bases celulares, virtualización de entorno de hogar, virtualización de CDN¹² y virtualización de redes

¹² <https://www.akamai.com/us/en/cdn.jsp>

de acceso fijo. De las cuales habla de los componentes que las integran y de cómo podrían ser aplicadas tecnologías NFV para optimizar sus redes.

La Virtualización de Funciones de Red es aplicable, en definición, a cualquier plano de control y procesamiento de paquetes de datos dentro de redes de comunicación. Los modelos que se detallan a continuación pueden ser aplicados tanto como para análisis técnico y comercial. Si se realiza un estudio más profundo en éstos modelos de aplicación se podrían considerar como casos de estudio y opciones de implementación futura.

- Elementos de conmutación como routers, switches, y equipos NAT
- Del mismo modo funciones contenidas en casa como routers de acceso a internet.
- Elementos de Red que presten servicios de Gateway y tunnelling (VPN, SSL)
- Analizadores de tráfico y de calidad de servicio (QoE)
- Entidades de servicios y señalización: IMS, SBC (Session Border Control)
- Plataformas de cloud computing, como por ejemplo virtualizar funciones de red como entornos IaaS y PaaS.
- Virtualización de Ambientes de hogar, como por ejemplo Gateways Residenciales y STB (Setup Boxes) utilizados para servicios de Media (TV).

Existen muchas más alternativas de elementos que tienen la posibilidad de ser virtualizados, pero antes de pensar en aplicar NFV a un determinado equipo, es necesario revisar las recomendaciones de ETSI y del grupo ISG, donde describen los casos que pueden ser más rentable y que presenten menos riesgos para las operadoras utilizar NFV. El despliegue de NFV depende del ahorro en costos y reducción de infraestructura que genere. A continuación en la figura 3.17 se muestran los valores y riesgos de virtualizar un componente específico de red.[19]

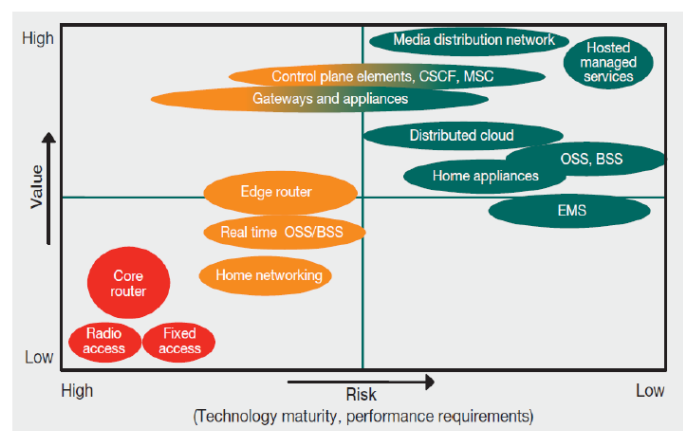


Figura 3.17. Pila de Protocolos de Plano de Control en LTE. [19]

3.4 Beneficios de NFV

La virtualización se ha venido aplicando a distintas áreas, en la actualidad los operadores de red van a transformar la forma en la que diseñan y dimensionan sus redes. Del mismo modo en el campo de la telefonía móvil, la virtualización sobre redes 4G y 5G ha planteado un paradigma, debido a sus posibles nuevas fronteras de aplicación y cambios en sus modelos de negocios. A continuación se detallan algunos de los beneficios más destacados que proporciona utilizar NFV, dentro de las redes de comunicaciones.

Reducción de Costos: El hecho de concentrar varios elementos de red virtualizados dentro de un conjunto de servidores, supone un aprovechamiento económico, como en la adquisición de equipos hardware de propietarios. Esto conlleva a una reducción de uso energético y espacial, por tanto los operadores no tendrán que invertir en entrenamiento y capacitación de personal para su operación y mantenimiento. Además la concentración de funciones de red genera un uso eficiente de los recursos hardware de los operadores, disminuyendo de forma notoria el tiempo y recursos empleados para el diseño y dimensionamiento de una red.

Algunos de los operadores consideran implementar Infraestructura NFV, como un servicio de plataforma interna que ayude al rápido despliegue de nuevo hardware personalizado, tratándolo como PaaS (Platform as a Service) o IaaS (Infrastructure as a Service). Esto ayuda operar funciones de red con hardware homogéneo, permitiendo la fácil interoperabilidad con enlaces virtuales, y además alta capacidad de requerimientos. [20]

Eficiencia Operacional: Las redes móviles y arquitecturas de servicios son significativamente complejas, sofisticadas y diversas debido a sus entornos y medios de despliegue. Tratar de automatizar procesos es un reto complejo pero, si se lo logra genera mayor versatilidad dentro de las redes.

La plataforma NFV permite la automatización de autocorrección de problemas en servicios “end to end” como los destacados, conmutación de tráfico dinámicamente de caminos congestionados, matar proceso de máquinas virtuales de VNF que están detenidos y transfiere la carga de trabajo de las VNFs a una nueva máquina virtual inicializada.

Operadores móviles pueden colocar VNFs en un centro de datos con plataforma en común, las cuales pueden ser compartidas por múltiples entidades del mercado y servicios, por ejemplo operadoras móviles de diferentes países, permitiendo la interconexión entre ellas y evitar saltos de comunicación más largos que implican

mayor coste tanto al operador como a los usuarios. Esta implementación necesitaría estudios sobre varios aspectos importantes, como el costo de transportar las redes, latencias y formación de tráfico generado.

Las operadoras móviles si desean implementar nuevos servicios, con NFV pueden tener mayor y mejor control sobre sus procesos, de esta forma reduciendo riesgos de incidencias que perjudiquen el normal funcionamiento de red y que no afecte a los usuarios. En caso de que el nuevo servicio no tenga el éxito esperado, fácilmente se lo podrá retirar y eliminar de la red, sin tener que asumir pérdidas por la compra de hardware propietario.

Innovación: La ausencia de hardware propietario, permite dar paso al avance en innovación de nuevos servicios generados por las operadoras móviles en conjunto con productores de software, teniendo la facilidad de desplegar sus aplicaciones dentro de entornos con plataformas NFVs.

Los modelos de arquitectura de 4G-LTE brindan altas capacidades de comunicación en sus servicios, más su convergencia a todo sobre IP, le permite tener un ambiente óptimo para implementar sistemas con plataformas NFVs dentro de su red central. Así podrá tener mayor escalabilidad y dispondrá de recursos aptos para soportar las nuevas tendencias de comunicación que generarán nuevas formas de servicios.

4 Desarrollo

4.1 Aplicación de NFV a Redes Móviles LTE

Los operadores de redes móviles están evolucionando sus redes de voz y datos hacia la virtualización de funciones de red (NFV) y también a las redes definidas por software (SDN), con el fin de ser más competitivos a través de la reducción de costos y mejorar las eficiencias operacionales de servicios, alcanzando cada vez más agilidad y rapidez en la entrega de nuevos servicios.

Como se detalló en las secciones anteriores los modelos de red 4G, poseen tres secciones principales: el acceso a la red por medio de E-UTRAN, la red central, EPC, y la nube de servicios IMS. Para cada una de estas secciones, existe la posibilidad de aplicar NFV, dependiendo a qué tipo de servicio se lo quiere incrementar su eficiencia, por ejemplo, si aplicamos en los nodos de acceso eNodeB, VNF permitiría introducir una expansión para redes de servicios M2M (Machine to Machine), convirtiéndose las operadoras móviles en redes convergentes hacia el IoT (Internet of Things). Del mismo modo si aplicamos VNF a la red central EPC, se incrementaría la eficiencia y fluidez del tráfico de datos, soportando cada vez más usuarios.

Y finalmente si aplicamos a la nube de servicios IMS, se podrán introducir nuevos y mejores servicios de negocios como VoLTE (Voz sobre IP en LTE). Se debe tomar en cuenta que los mismos criterios de aplicación de VNF para redes LTE se los puede aplicar a redes móviles 3G-UMTS, ya que en la actualidad, dentro de una misma arquitectura de una operadora coexisten ambas tecnologías, hasta que se termine de evolucionar completamente a conmutación de todo sobre IP.

Por tal motivo, antes de aplicar NFV a cualquier red de comunicaciones, primero se deben separar sus redes o secciones e identificar sus elementos importantes, para luego analizar qué tan confiable es virtualizarlos y que esto no supondrá un riesgo para la operadora. Tomando en cuenta que no todos los equipos de una red pueden ser virtualizados, debido a su compleja y sofisticada composición, así como su funcionamiento, podrían ser sensibles a pérdidas de rendimiento y capacidad, afectando de forma directa tanto a los usuarios como a la operadora móvil.

¿Por qué NFV para el Núcleo de Red?

Con el aumento en la demanda de servicios multimedia, la necesidad de mejorar la calidad de experiencia de cliente y la aparición de nuevos servicios tales como voz

sobre LTE (VoLTE) y M2M, los operadores de redes móviles están buscando nuevas soluciones para hacer frente a los objetivos de negocio.

Estos objetivos llevaron a la examinación de las nuevas tecnologías que podrían abordar mejor la naturaleza dinámica e impredecible de la prestación de servicios de banda ancha móvil para consumidores y empresas. El primer enfoque se realizó hacia la industria de la computación en la nube y en base a sus servicios se estudió si resultaría una elección lógica para las operadoras aplicar NFV sobre los componentes del núcleo de red (EPC), tomando en cuenta sus objetivos de costo/beneficio en los servicios prestados.

4.2 VNF de EPC

La clave para conseguir una virtualización exitosa de cada componente de EPC es, desplegar un buen desempeño de red y principalmente mantener las máquinas virtuales completamente aisladas. La virtualización de la EPC permite a los proveedores de servicios que se den cuenta rápidamente sobre el importante ahorro de costos y el aumento de la velocidad del mercado en el despliegue de nuevos servicios. Además, los proveedores de servicios pueden reducir los gastos operativos ejecutando sus productos de forma dinámica y de ésta manera dando de baja varios recursos de red basado en la demanda y la capacidad de la red.

Donde se realiza y gestiona el flujo crítico de datos en LTE, es dentro de EPC. Es preciso identificar el funcionamiento de cada elemento interno de EPC, así como su importancia dentro del procesamiento de los paquetes, debido a que cada uno de ellos tiene aspectos específicos que considerar al momento de decidir si se los despliega con NFV, por tal motivo sus aspectos y características fueron descritas en los capítulos anteriores.

Los Niveles de Acuerdo de Servicios (SLA), juegan un papel importante dentro de la virtualización dentro del núcleo de la red móvil. Debido a que la virtualización, por concepto se desempeña y ejecuta fuera del área de comercio de equipos propietarios. Por tanto los proveedores se ven obligados a optimizar el procesamiento en el plano de datos dentro de sus equipos para satisfacer y asegurar buena calidad de servicio en servicios regulares brindados a los usuarios.

Aplicar VNF a EPC (vEPC), significa que se puede desplegar múltiples VNFs y ser gestionados por NFVI, que es capaz de abastecer el rendimiento de la escalabilidad dentro de los planos de control y de datos de usuario, los cuales requieren distintos tipos de recursos.

Los elementos directos que se puede aplicar VNF dentro de EPC de 4G, son S-GW, P-GW, MME, HSS y PCRF, a los cuales se los puede agrupar en varios centros de servidores, que vendrían a ser nuevos núcleos de red (EPC). Esto significa que pueden escalar independientemente en función de las necesidades de requerimientos de la red, y de igual forma que un centro de datos, podrían ser ubicados en distintos sitios donde sean más necesarios, para incrementar los recursos del plano de datos y de control, sin afectarse el uno al otro.[7][18]

La arquitectura de EPC brinda mayor flexibilidad a los operadores, permitiéndoles agrupar y combinar nodos EPC que involucren un número de NFVs. Por ejemplo como se muestra en la Fig. 4.18, se pueden crear distintas NFV con varios nodos EPC independientes y que se pueden comunicar hacia la red de acceso con la interfaz externa S1. Dentro de estos nodos EPC, se encuentran virtualizadas las entidades S-GW, P-GW, etc. Las cuales gestionan de forma separada las funciones de plano de control de las de datos, al igual que de las funciones de plano de Media (datos de información). Con esto les permite a los operadores optimizar la gestión y control de las comunicaciones entre EPC e IMS para brindar servicios actuales como VoLTE.

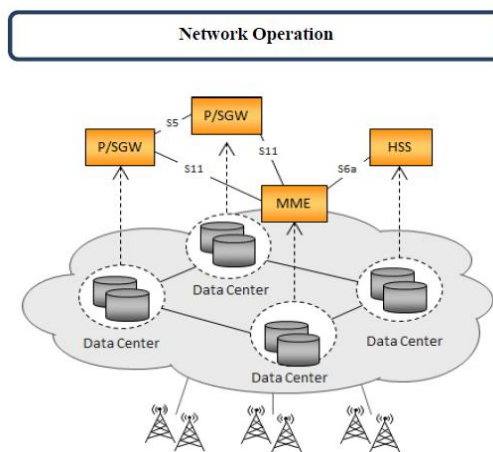


Figura 4.18. Virtualización General de EPC. [18]

EPC está diseñado para permitir que los servicios de banda ancha móvil mediante la combinación de infraestructura IP y movilidad puedan fluir de mejor manera a más grandes velocidades. Por otra parte, el EPC está diseñado para soportar una variedad de tecnologías de acceso, por lo que visto desde ese punto ahora, se hace imprescindible disponer de una red flexible, robusta y de fácil manejo; una red que pudiera escalar bajo demanda en tiempo real y ser fácilmente manejable.

La virtualización de EPC puede aplicar directamente sobre los siguientes componentes que soportan conectividad IP:

1. La entidad de gestión de movilidad (MME) es la principal entidad de plano de control en la red LTE.
2. El servicio de pasarela (SGW) es responsable de enrutamiento y reenvío de paquetes de datos de usuario desde y hacia la estación base.
3. El paquete de pasarela de red de datos (PDN-GW) (PGW) asegura la conectividad entre el plano de datos de usuario y redes externas.
4. El servidor local de abonado (HSS) es la base de datos central de información del usuario.
5. La función de reglas de carga y la política (PCRF) es responsable de la aprobación y la decisión de la política y de carga en tiempo real para cada servicio y el usuario.

Aplicar NFV por ejemplo a S-GW, P-GW, MME, pueden ser escalados independientemente acorde a sus requerimientos específicos, por ejemplo donde puede ser necesario incrementar recursos de plano de usuario sin afectar los recursos o simplemente independientemente del plano de control y viceversa. La relación de las VNF con el plano de datos puede requerir mayor cantidad de recursos de infraestructura de NFV a diferencia de la relación con el plano de señalización. [18]

Para alcanzar la continuidad y disponibilidad de servicios deseados por el operador, el rendimiento en el plano de control y de usuario debe ser asegurado. Como por principio, la virtualización de funciones desacopla el hardware de las funciones de red, diseñar nuevos esquemas más tolerantes y resistentes a fallos puede llegar a ser posible utilizando la portabilidad de instancias VNF, aprovechando la re-locación de las máquinas virtuales. En la figura 4.19 se muestra la interconexión entre EPC e IMS, donde sus componentes internos son instancias VNF, que están trabajando por medio de sesiones de IMS. Aquí se puede aplicar el concepto de reubicación, debido que si una de las instancias virtualizadas VNF sufre una sobrecarga o desbordamiento de datos, puede fallar, por tal motivo su gestión, control, manejo de instancias y sesiones debe ser realizada de manera apropiada, con el fin de evitar pérdida de servicios y mantener la continuidad del flujo de información.

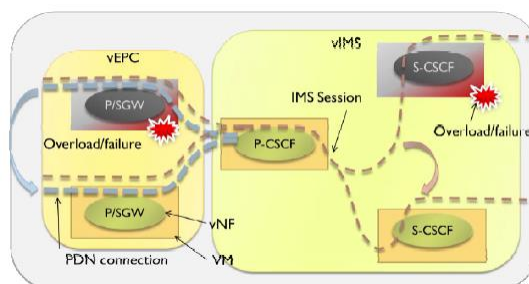


Figura 4.19. Creación y Reubicación de Instancias VNF.[18]

4.2.1 Coexistencia entre redes Virtualizadas y No-Virtualizadas de EPC

Se pueden diseñar distintos escenarios de VNF, según las necesidades y recursos de los operadores. Existe la posibilidad de realizar un único núcleo de red basado en NFV, o simplemente varios VNF, que reúnan algunas entidades virtualizadas. Pero con la capacidad de coexistencia entre ambas arquitecturas

Un operador móvil como fase inicial, necesitará desplegar una arquitectura híbrida, donde puedan coexistir nodos EPC no virtualizados con nodos VNFs, También deberá tener la capacidad y libertad de poder escoger, coordinar y controlar el flujo de datos entre ellos, acorde a los requerimientos de recursos de red. A continuación en la Fig. 4.20, se muestra un ejemplo de redes con diferentes características que coexisten.

El hecho de implementar estos métodos de arquitectura, genera nuevos retos a los operadores móviles. Deben ser capaces de coordinar el funcionamiento entre ambas arquitecturas, empezando desde el punto de acceso en E-UTRAN. Tendrán que controlar y dirigir el tráfico por ambos núcleos de red, dependiendo de la locación de cada uno y según la cantidad de tráfico que se lo esté generando. Además los servicios que brindan deben ser transparentes a los usuarios que los solicitan: No deben conocer si están ocupando una entidad virtualizada o no. Para lograr esta transparencia, se los debe construir con replicación, detección de fallos y recuperación frente a fallos, como el diseño de una aplicación distribuida, aplicando mecanismo de monitorización y diagnóstico.

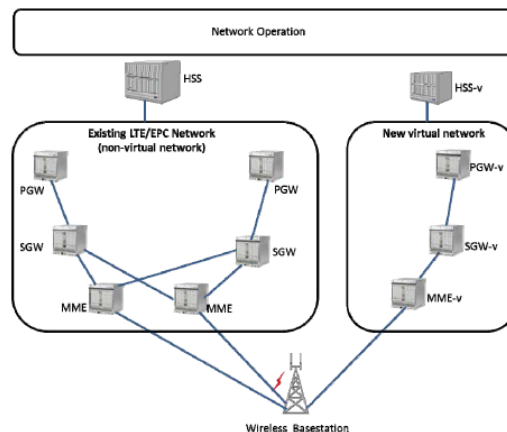


Figura 4.20. Coexistencia entre Redes Virtualizadas y No Virtualizadas.[18]

4.2.2 Virtualización Parcial de EPC

En función de las necesidades del operador móvil, la virtualización le brinda mayor flexibilidad en la configuración de su arquitectura, así como se explicó en el caso anterior, pueden existir simultáneamente varios núcleos EPC. En esta sección se

describe como se definiría la arquitectura de una virtualización parcial de los componentes EPC. La ETSI presenta dos posibles escenarios, los cuales se describen a continuación:

- Virtualización de algunos de los componentes del núcleo de la red móvil. Pueden ser componentes de control o nodos de servicios. En la siguiente figura se muestra la virtualización solo de los nodos MME, que se conectan a los componentes reales P-GW y S-GW, por medio de la interfaz S11.

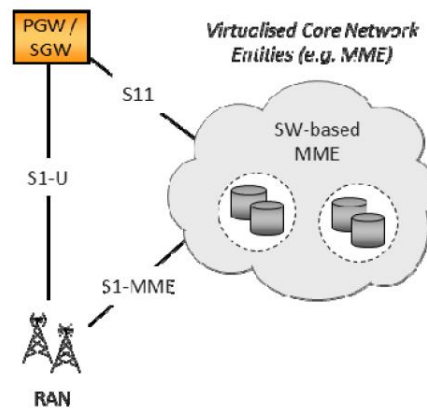


Figura 4.21. Virtualización Parcial de Componentes EPC.[18]

- La virtualización puede ser usada para servicios, usuarios y dispositivos específicos o para no sobrepasar la capacidad de tráfico de una instancia de red no virtualizada. En la figura a continuación se observa que tanto para redes habituales y virtualizadas se encuentra la entidad MME, el operador será capaz de coordinar y gestionar el tráfico que desee por cada una de ellas según sea su necesidad. Del mismo modo lo puede aplicar al resto de nodos que componen su núcleo.

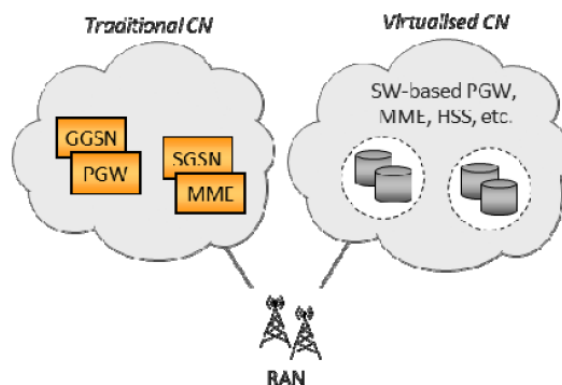


Figura 4.22. Virtualización según Servicio Específico. [18]

4.2.3 Virtualización de la Entidad de Gestión de Movilidad (vMME)

Dentro del plano de control, MME es el principal componente. Es responsable de coordinar algunos aspectos importantes dentro de las comunicaciones celulares como por ejemplo, mantener el estado de movilidad de UE, gestionar portadoras y provisionar QoS a los flujos de datos de los usuarios. Dentro de la arquitectura de virtualización, vMME se implementa como instancia NVF en los centros de virtualización. Esto permite desplegar varios componentes MME a escala en función del crecimiento de la red, compartiendo el procesamiento entre todas las instancias activas MME creadas.

Para poder realizar su buen funcionamiento, el protocolo de comunicación y el estado de los usuarios son almacenados en una base de datos compartida como se muestra en la figura 4.23. Para facilitar la compartición de procesamiento, el switch de la entrada del centro de datos actúa como un balanceador de carga y programa cada proceso de solicitud para la instancia MME.

Para garantizar la escalabilidad, resistencia y rendimiento del procesamiento de una instancia NFV, cada procedimiento se divide en transacciones de petición y respuesta. Por ejemplo, cuando una NFV termina su proceso, guarda el estado de la transacción en la base de datos compartida. Entonces si una solicitud posterior llega a una instancia MME, ella en primer lugar consulta el estado de la transacción en la base de datos para de esta forma, pueda continuar con el resto de procesos. Por lo tanto, se evita el bloqueo de la función NFV si posee procedimientos con varias etapas que dependen de la finalización de fases anteriores [21].

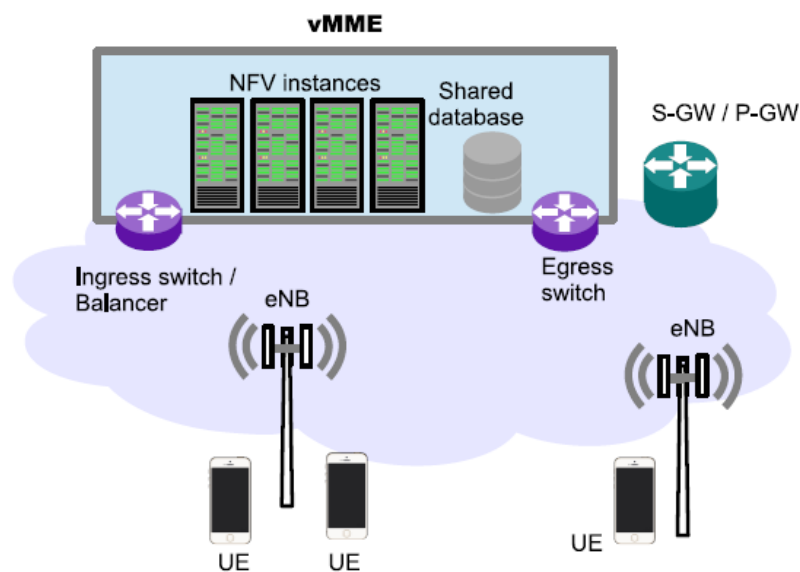


Figura 4.23. Sistema General vMME [21]

4.3 Agrupamiento de Entidades EPC en entorno NFV

Para los proveedores de equipos de telecomunicaciones la implementación de un EPC virtualizado (vEPC) es su principal objetivo. EPC como se ha descrito anteriormente, engloba múltiples funcionalidades, por lo tanto la creación de instancias de VNF en la nube tiene un tremendo efecto en el rendimiento; debido a esto, VNFs se agrupan en función de sus interacciones y la carga de trabajo. En general, es beneficioso crear instancias de cada grupo definido en un servidor físico que puede estar en la nube o simplemente en una red local en función de la carga de trabajo y de los servicios que brindan.

En base a los estudios de la arquitectura de EPC de LTE, se debe tomar en cuenta el análisis de las interconexiones y funcionalidades de las entidades vEPC con el fin de lograr un menor control del tráfico y de la congestión de señalización en el plano de datos. El enfoque propuesto mantiene los dos principios de la arquitectura de EPC, así como el desacople entre los planos de control y de datos.

Para entender de mejor manera, se recomienda realizar un análisis exhaustivo de la arquitectura de red LTE. Esta agrupación divide a las entidades en cuatro segmentos. A continuación en la Figura 4.24., se detallan sus beneficios. [22]

GRUPOS	ENTIDADES	BENEFICIOS
Segmento Uno	<ul style="list-style-type: none"> • HSS • MME 	<ul style="list-style-type: none"> • Interconexión entre HSS y MME ocurre localmente. • Menos transacciones de red entre Vswitches. • Transacciones de red usa protocolo LDAP, el cual es eficiente para consultas de información de base de datos
Segmento Dos	<ul style="list-style-type: none"> • HLR • SGSN 	<ul style="list-style-type: none"> • Soporta la combinación existente entre SGSN y el sistema de EPC por medio de la interfaz Gn. • Interconexión entre HLR y SGSN ocurre localmente. • Menos transacciones de red entre Vswitches. • Transacciones de red usa protocolo LDAP, el cual es eficiente para consultas de información de base de datos
Segmento Tres	<ul style="list-style-type: none"> • PGW • PCEF • SGW 	<ul style="list-style-type: none"> • Minimiza el número de procesos en el plano de datos de los nodos • Ayuda a superar el reenvío de datos y los cuellos de botella de red • Mejora el monitoreo y carga de los datos
Segmento Cuatro	<ul style="list-style-type: none"> • UDR (User Data Repository) • OCS (Online Charging System) • OFCS (Offline Charging System) • PCRF 	<ul style="list-style-type: none"> • Unifica la Base de Datos de los usuarios; menos fragmentación. • PCRF interactúa localmente con UDR para generar políticas. • OCS y PCRF interactúan localmente. • Punto central de Interacción por OSS/BSS. • Menos transacción de red a través de Vswitches

Figura 4.24. Agrupamiento de entidades EPC en entorno NFV. [22]

4.3.1 Segmento Uno

En esta agrupación se migra hacia la nube las entidades MME y HSS. Con la implementación de HSS junto a MME, los procesos de autenticación y autorización se realizan internamente, sin ningún tipo de tráfico de datos a través de la red. El HSS asegura que todas las comunicaciones con MME suceden como si MME realmente se encuentra accediendo a la base de datos completa de HSS. El HSS envía consultas de información de usuario desde el repositorio de datos de usuario (UDR), el cual es la base de datos central de información del usuario. HSS desempeña todos los procesos de autenticación y autorización con la entidad MME.

Además esta agrupación minimiza el número de transacciones de red que deberían ser efectuadas para autenticar al usuario debido que HSS obtiene toda la información requerida con una única consulta (query)[22]. Todo esto gracias a que se ocupa el protocolo LDAP, que permite el intercambio y gestión de la información de servicios de directorios sobre redes IP. Para el intercambio de información segura ocupa TLS y SSL, las mismas que ocupan menos recursos a diferencia de IPsec, por lo tanto proveen mayor flexibilidad a una plataforma virtualizada.

4.3.2 Segmento Dos

En el presente grupo, SGSN se migra en conjunto con HLR. En forma general SGSN es una entidad de servicio que tiene una funcionalidad muy parecida a la combinación de MME y SGW. En forma general SGSN es una entidad que existe en el núcleo de las redes GPRS, el cual permite a los sistemas 2G y 3G transmitir paquetes IP hacia redes externas.

SGSN realiza tareas esenciales como la gestión de enrutamiento y transferencia de paquetes. Además, realiza funciones en el plano de control como, gestión de movilidad, de enlace lógico, y de autenticación. HLR posee la base de datos que conserva la información de usuario dentro del sistema móvil GSM, por lo tanto se combina con SGSN por las mismas razones que lo hacen MME y HSS. [22]

El resultado de esta combinación permite tener una base de datos unificada y soporta la coexistencia de SGSN por medio de la interfaz Gn con el sistema de EPC. La interfaz Gn está basada en el protocolo de túnel GPRS (GTP).

4.3.3 Segmento Tres

Aquí se propone a que los componentes PGW y SGW se migren hacia la nube. Se realiza la combinación de dos entidades que trabajan en el plano de datos, por lo que se minimiza el número de nodos de procesamiento del plano de datos. La aplicación de estas dos entidades en una máquina virtual o VNF genera un gran beneficio dentro del procesamiento centralizado en el plano de datos, ayudando a superar cuellos de botella en la red.

En este agrupamiento, los datos del usuario no se direccionan a PGW después de haber sido atendidos por SGW, en su lugar, el paquete posee acceso directo a PGW, el cual direcciona hacia las redes externas. Dicha combinación tiene el beneficio de evitar enrutamiento innecesario a través de los Vswitches, que son una de las principales causas de cuello de botella en entornos virtuales. [22]

Para obtener un mejor rendimiento de VNF también se puede utilizar el acceso directo de interfaz de red, con el fin de satisfacer latencias y calidad de servicio requeridos por parte de los componentes PGW y SGW. Esta migración elimina el tráfico de señalización entre SGW y PGW, ya que son dichas transacciones son realizadas internamente.

4.3.4 Segmento Cuatro

Aquí se propone que los componentes UDR, PCRF, OCS y OFCS sean agrupados y migrados. Tener agrupado a UDR en conjunto con PCRF permite gestión eficiente de la función de política de información de los usuarios, debido a que la PCRF solicita información de usuario por cada portadora. Esto previene saturación en el intercambio de información en el nodo de red. Del mismo modo como consecuencia reduce latencias y acelera la aplicación de políticas a la PGW.

En este caso OCS se utiliza para facturación a los usuarios que hagan uso de la red en tiempo real, como por ejemplo; prepago, mientras que SCFO se utiliza para facturar a los usuarios después de haber terminado la sesión. OCS y SCFO se interconectan con PCRF y PCEF con el fin de recopilar información de sesiones y poder cumplir las políticas de tarificación, como por ejemplo finalizar la sesión de comunicación una vez haya consumido el crédito contratado.

Este grupo se caracteriza debido a que todas las entidades se encuentran conectados a través del protocolo GTP. Los protocolos SCTP y TCP, son conocidos por consumir ancho de banda aun cuando se intercambian pequeñas cantidades de datos, debido a que los paquetes de control poseen mecanismos de acuse de recibo para establecer

conexión. Mientras que GTP está basado en UDP en su capa de transporte. Mantener el enfoque de interfaces con GTP para el control de señalización permite tener un mejor rendimiento computacional y de recursos de red. Aunque la agrupación propuesta introduce beneficios en el hecho de minimizar tráfico de señalización para evitar congestión en la red de control, es necesario tener mucha potencia de cálculo porque la mayoría de las transacciones se llevan a cabo internamente.[22]

Una vez detallado cada segmento de agrupación y su forma de despliegue y conexiones, a continuación se muestra en la figura 4.25, cómo estarían formados los segmentos y por medio de qué interfaces se comunican. De esta forma se puede visualizar como quedaría formado un vEPC, que puede ser desplegado en la nube o posiciones geográficas estratégicas según la necesidad del operador. Además éste núcleo puede ser replicado y distribuido en distintos centros de datos, y al mismo tiene la capacidad para comunicarse con núcleo EPC tradicional.

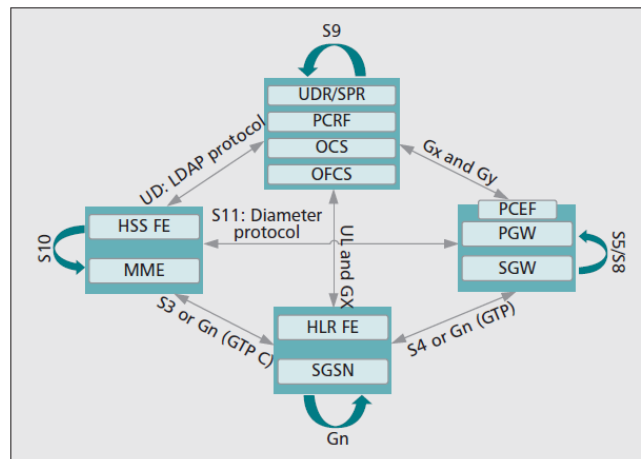


Figura 4.25. Agrupación de componentes vEPC. [22]

4.4 Tipos de Escenarios de Comunicación con vEPC

4.4.1 Escenario de Comunicación End to End

En la sección anterior se explicó cómo se implementaría una vEPC agrupándolos por segmentos a sus componentes donde cada uno de ellos puede ejecutarse en diferentes VNFs. Cada una de las diferentes funciones MME, P-GW, etc., corren en sus propias máquinas virtuales en la nube. El eNodeB estará funcionando ya sea como en su forma tradicional interconectada al núcleo vEPC o a su vez puede ser probado como un emulador[23]. Se aplica un túnel GTP entre eNB y S/PGW en eNodeB.

Se mantiene QoS, utilizando etiquetado MPLS para la identificación de los flujos en los switches OpenFlow (OFS). Como se muestra en la figura 4.26, los paquetes de datos de UE son direccionados entre eNB y el S/P-GW siguiendo las especificaciones actuales de 3GPP basado en túneles de GTP.

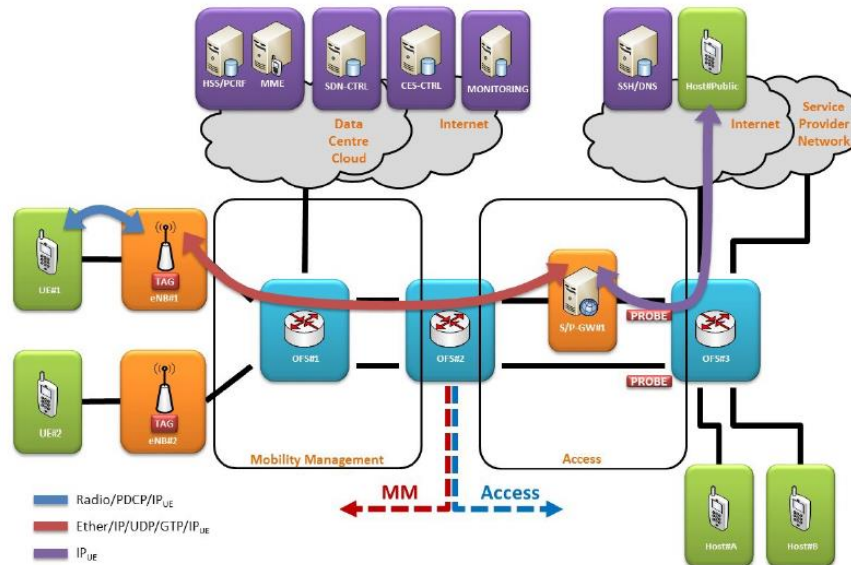


Figura 4.26. Ejemplo de escenario vEPC con túnel tradicional GTP[23].

4.4.2 Escenario con Arquitectura de Plano de Usuario sin Estado

El principal objetivo de vEPC es eliminar todos los estados de movilidad de los nodos en el plano de usuario de vEPC. En esta propuesta se pretende que enrutadores y conmutadores estándar se pueden utilizar para reenviar el tráfico de nodos móviles en el plano de usuario de vEPC. Se plantea una solución que es básicamente una combinación de los protocolos existentes definidos en IETF [24].

En lo que respecta a la arquitectura, para el plano de usuario, se propone utilizar dos routers frontera tradicionales antes de que los paquetes sean direccionados hacia vEPC. Estos routers son EPC Edge Router (EPC - E) y Router (RTR), los cuales se describen a continuación:

- **EPC Edge Router (EPC - E):** Se encuentra localizado en el mismo sitio de SGW y termina con el túnel GTP que es establecido por eNodoB (RAN). EPC-E es compatible con las funciones del plano de usuario de la SGW y PGW, pero adicionalmente se lo configura una dirección *anycast* para la interfaz de red con dirección a eNodoB. Debido a la dirección *anycast*, el tráfico del UE siempre se encamina a la EPC-E más cercana del equipo de

usuario. EPC-E también mantiene la información de enrutamiento de cada equipo de usuario que se notifica por el plano de control.

- **Router (RTR):** Es un router IP normal cuya función es anunciar la información de enrutamiento del UE hacia la red externa. El plano de control de vEPC distribuye la información de enrutamiento de cada equipo de usuario mediante un protocolo de enrutamiento como BGP. Por lo tanto, no se necesitan protocolos adicionales distintos de los protocolos de enrutamiento para RTR. [24]

En la arquitectura funcional de ésta propuesta se omite el túnel que se establece originalmente entre SGW y PGW para encaminar paquetes en el plano de usuario. Mediante la eliminación de este túnel, los paquetes de la UE se direccionan con tablas de enrutamiento en los routers de la red central. En lugar de una configuración del túnel, los estados creados por un Proxy Mobile IP se distribuyen a todas las entidades de enrutamiento (EPC-E y RTR) por un protocolo de enrutamiento. EPC-E y RTR no forman parte de ningún proceso de señalización del plano de control.

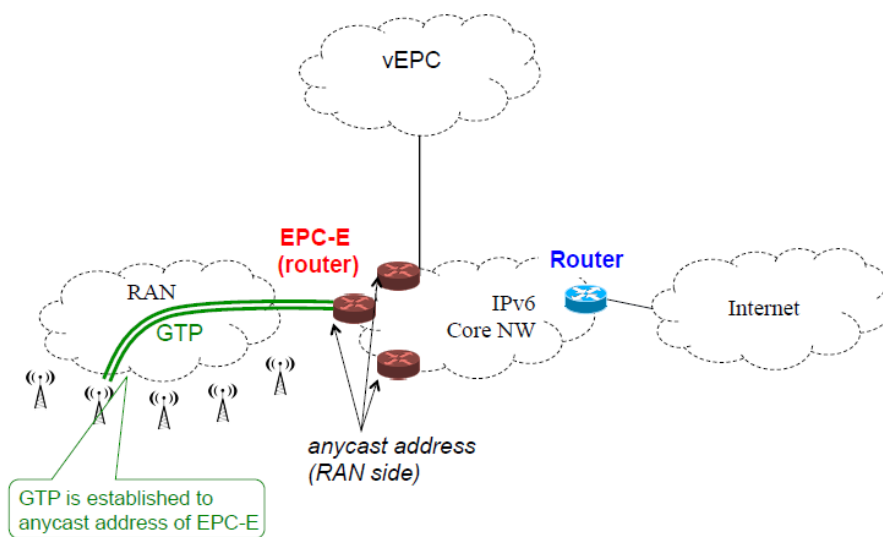


Figura 4.27. Arquitectura Propuesta escenario plano de usuario sin estado. [24]

4.4.3 Uso del Protocolo OpenFlow en Arquitectura LTE-EPC

Para habilitar la flexibilidad, capacidad de virtualización y programación de la red móvil, se ha realizado una propuesta sobre el plano de control para la arquitectura EPC. En dicha propuesta, todos los protocolos de control que se ejecutan en las interfaces S1-MME (entre MME y eNB) y S11 (entre MME y SGW) son reemplazados

por el protocolo OpenFlow, como se muestra en la figura 4.28. Por lo tanto, lo que se logra es la separación de todas las funciones de control de las de transmisión de datos en SGWs.

Entonces, toda la inteligencia de procesamiento de SGW del plano de control, se la centraliza y ejecuta en la capa superior del controlador OpenFlow como una aplicación. Por tanto se podría gestionar y orquestar de mejor manera todas las instancias VNF creadas como vMME, vSGW y vPGW. La función de transmisión de datos se realiza por el plano de usuario SGW (SGW-Data), que de igual forma puede dicha instancia estar virtualizada.

El controlador OpenFlow es el componente principal dentro de dicha propuesta, debido que gestiona el plano de reenvío de datos de eNB y SGW-D. Además es responsable de los procesos de establecimiento de sesión de usuario y monitorización de la carga en el plano de datos [25].

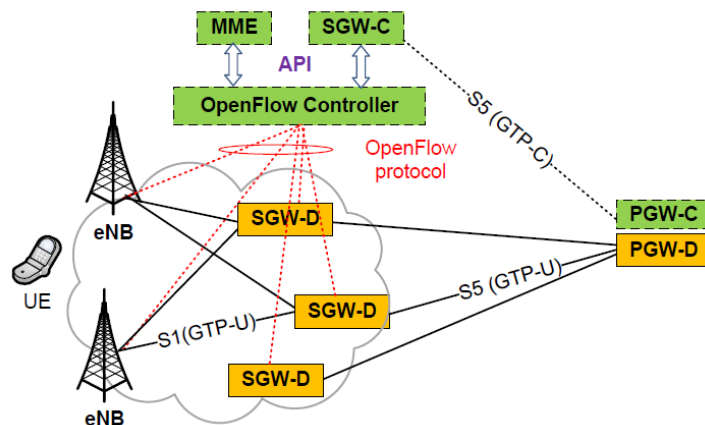


Figura 4.28. Arquitectura vEPC Basada en OpenFlow [25]

Dentro de la arquitectura EPC basada en el protocolo OpenFlow, la gestión del plano de datos incluye principalmente dos procedimientos, OpenFlow unión inicial y OpenFlow configuración de portadora de acceso.

- 1) **OF Unión Inicial:** Consiste cuando el usuario UE envía un mensaje de petición de unión al eNB. Este dirige el mensaje de solicitud de acople dentro de OFPT_PACKET_IN como mensaje uno y lo reenvía al controlador OF. Éste a su vez, reenvía la petición de unión hacia MME, para realizar los procesos de autenticación y autorización. El intercambio de autenticación entre UE y MME se lo hace a través del controlador OF, y una vez realizado con éxito éste proceso, el controlador Of dirige la información hacia SGW de control (SGW-C) para seleccionar a SGW de datos (SGW-D) y a PGW.

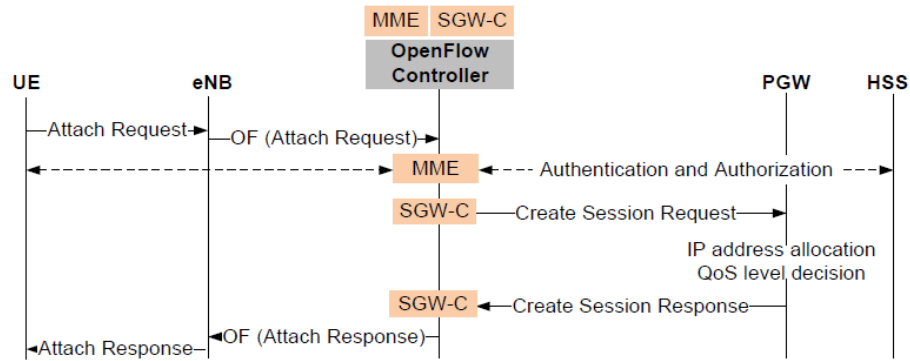


Figura 4.29. Proceso de unión inicial OF.[25]

- 2) **OF Configuración de Portadora de Acceso:** Este procedimiento se aplica a cada nueva sesión que se establece. Cuando eNB recibe un paquete inicial, primero comprueba y revisa sus tablas de flujo. Como no existe ninguna entrada para este flujo de paquete inicial, eNB lo reenvía al controlador OF como cabecera del paquete por medio del mensaje de OFPT_PACKET_IN. El controlador OF analiza la cabecera del paquete con el fin de identificar el tipo de sesión y las direcciones IP de origen y destino. Una vez realizado este proceso envía esta información a SGW-C, la cual a su vez envía de nuevo al controlador OP como respuesta la dirección IP de SGW-D. Por tanto el controlador OF crea una entrada de flujo para los paquetes subsiguientes relacionados con dicha sesión y lo envía a eNB a través del mensaje OFPT_PACKET_OUT.

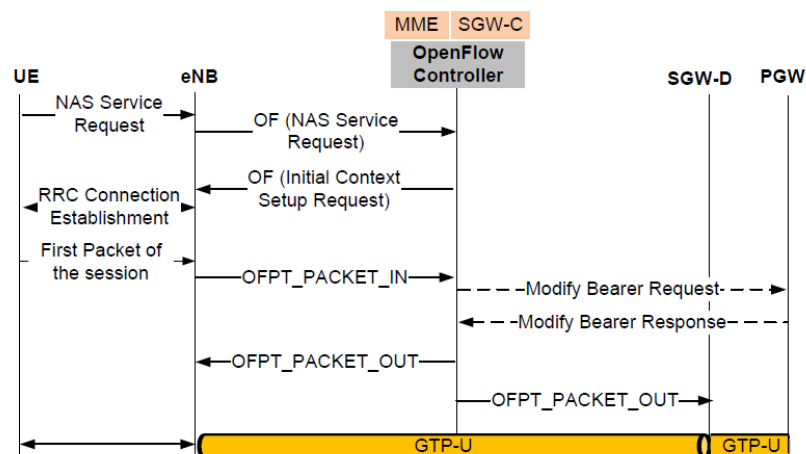


Figura 4.30. Proceso de configuración de portadora OF.[25]

4.5 Recomendaciones en Implementación de Virtualización de EPC

Cuando se piensa en aplicar el proceso de virtualización sobre redes móviles, se deben tomar en cuenta ciertos aspectos de alto nivel, para brindar las mejores soluciones a los usuarios y evitar posteriores problemas en el uso de la tecnología NFV. El grupo de la ETSI hace la recomendación de tomar en cuenta, lo siguiente:

1. *Escalado de Recursos*: Ampliación y disminución de recursos de red de una EPC virtualizada.
2. *Conciencia de Servicio*: Servicio de asignación de recursos conscientes a las funciones de red.
3. *Virtualización transparente para los servicios*: Los servicios que usan una función de red no necesitan saber si es una función de red es virtualizada.
4. *Virtualización transparente para el control y gestión de la red*: Los planos de control y gestión de red no necesitan ser conscientes si una función es virtualizada o no.
5. *Mantenimiento de Estado*: Estado sobre gestión de redes y de funciones de redes durante la reubicación, replicación escalamiento de recursos de funciones de red.
6. *Monitoreo, Detección de fallos, Diagnóstico, Recuperación*: Se recomienda implementar mecanismos apropiados para monitorizar, detectar fallos, diagnosticar y recuperar todos los componentes y sus estados después de haber sido virtualizados.
7. *Disponibilidad de Servicio*: Alcanzar la misma calidad y nivel de servicio de una comunicación extremo a extremo en un núcleo de red virtualizado, tanto como en una red central móvil tradicional.
8. *Mecanismo de Separación Plano de Control del Plano de Usuario*: Identificar y separar los datos de control de los de usuario en redes centrales virtualizadas.
9. *Impacto sobre Funciones Relevantes*: Minimizar el impacto sobre funciones de redes no virtualizadas y soportar Sistemas Operativos de Redes.

Con estas recomendaciones se podrá mantener y mejorar los servicios prestados por parte de las redes centrales móviles virtualizadas (vEPC) y tradicionales.

4.6 Retos y Aplicaciones de la Gestión y Orquestación de NFV

Como se ha detallado en las secciones anteriores, aunque se han hecho muchos progresos y avances relacionados a la virtualización de funciones de red, todavía existen muchos desafíos técnicos en donde distintos operadores y proveedores de servicios han desarrollado distintos proyectos relacionados a MANO de NFVs. Creados con el fin de obtener los beneficios esperados de NFV y optimizar su gestión. El reto detrás de este concepto, es que MANO representa un aspecto crítico en lo que se refiere a garantizar el correcto funcionamiento de la infraestructura de NFV, así como de las funciones de red virtuales. MANO proporciona las funcionalidades y herramientas necesarias para el aprovisionamiento de VNFs, así como las operaciones relacionadas a la infraestructura de configuración y ejecución de estas funciones. Incluye la gestión del ciclo de vida y la orquestación de los recursos físicos y virtuales que apoyan el correcto funcionamiento de VNFs.

En esta sección, se analizarán los retos y aspectos actuales que están relacionados con MANO de NFV. Empezando por la explicación del entorno MANO que se ha propuesto por ETSI. Posteriormente se muestra un panorama de proyectos representativos y productos de proveedores enfocados en MANO de NFV. Detallando los más representativos y especialmente que son de acceso libre. Los cuales permiten tener mayor integración por parte de desarrolladores e investigadores dentro del campo de NFV.

4.6.1 Entorno MANO desarrollado por la ETSI

El marco arquitectónico NFV-MANO está basado en principios que soportan los conceptos de estructuración y gestión por capas, además de la funcionalidad de la orquestación de cada uno de esos dominios.

El marco promovido por ETSI MANO se muestra en la figura 4.31. Para su mejor entendimiento, a los bloques funcionales del entorno se los pueden agrupar en tres entidades principales: (1) capas arquitectónicas de NFV (Color Naranja); (2) gestión y orquestación de NFV (Color Azul) y (3) Sistemas de gestión de red (Color Verde). De esta forma, sus bloques funcionales que los conforman, están conectados entre sí mediante un conjunto de puntos de referencia [26]. Las capas arquitectónicas del entorno NFV incluyen NFVI y VNFs. Donde NFVI constituye la base principal para el despliegue de virtualización, es la combinación de ambos recursos hardware y software, que forman el entorno en el que se despliegan VNFs.

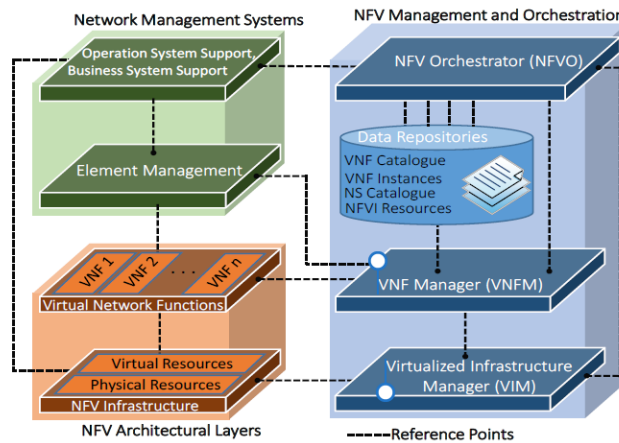


Figura 4.31. Arquitectura MANO propuesta por ETSI NFV. [27]

MANO de NFV como se mencionó, consiste en tres bloques funcionales, VIM (Manager), NFVO (Orchestrator) y VNFM (Manager), más repositorios de datos. Los mismos que se detallan a continuación.

- 1) **Gestor de Infraestructura Virtualizada (VIM):** Un VIM gestiona y controla los recursos físicos de NFVI como computación, almacenamiento y redes, que usualmente se encuentran dentro del dominio de infraestructura de un operador. Esto resulta que una arquitectura NFV puede contener más de un VIM, ya que con cada uno de ellos se puede gestionar y controlar los recursos NFVI de un proveedor de infraestructura determinado. En principio, un VIM puede estar especializado en gestionar varios tipos de recursos (por ejemplo nodos en el NFVI).

La siguiente lista expresa las funciones realizadas por el VIM. Estas funcionalidades pueden estar expuestas por entidades externas autorizadas [26]:

- Orquesta la locación, actualización, despliegue y recuperación de recursos NFVI (incluyendo la optimización del uso de tales recursos), y la gestión de las entidades virtualizadas asociadas a los recursos físicos de computación, almacenamiento y red. Por lo tanto, el VIM mantiene un inventario de la asignación de recursos virtuales a los recursos físicos.
- Soporta la gestión de reenvío de gráficos VNF, por ejemplo, mediante la creación y mantenimiento de enlaces virtuales, redes virtuales,

sub-redes, y puertos. Del mismo modo la gestión de las políticas para garantizar la seguridad y control de acceso del tráfico de red.

- Recolección de Información sobre rendimiento y fallos (a través de notificaciones) de los recursos de hardware, software (por ejemplo hipervisores), y de recursos virtualizados (Virtual Machine).

2) Gestor de VNF (VNFM): Cada instancia VNF se supone que tiene un VNFM asociado. El VNFM es responsable de la gestión del ciclo de vida de VNFs. Esto quiere decir que a cada VNFM se le puede asignar la gestión de una o varias instancias VNF del mismo o de diferentes tipos.

La mayor parte de las funciones del Gestor de VNF se supone que son funciones comunes genéricas, aplicables a cualquier tipo de VNF. Sin embargo, el marco arquitectónico NFV-MANO debe apoyar también los casos en que VNFs necesitan funcionalidades específicas para la gestión del ciclo de vida. Los siguientes puntos detallan algunas funciones desempeñadas por VNFM[26]:

- Instancia VNF, incluye la configuración inicial VNF con direcciones IP antes de la finalización de la operación de instanciación VNF.
- Actualización de Software de Instancia VNF
- Modificación de Instancia VNF
- Recolección de resultados de NFVI sobre medición de rendimiento, fallos e información de eventos.
- Terminación de Instancia VNF
- Notificación del cambio de gestión del ciclo de vida de VNF

3) Orquestador NFV (NFVO): El NFVO está dirigido a la combinación de varias funciones con el fin de crear servicios de extremo a extremo. Quiere decir que la funcionalidad NFVO se puede dividir en dos categorías: orquestación de recursos, y orquestación de servicios. El primero es utilizado para proporcionar servicios de apoyo con el fin de acceder a recursos NFVI de manera independiente de cualquier VIM, y también la administración de instancias VNF compartiendo los recursos NFVI. Y el segundo se refiere a la creación de servicios de extremo a extremo mediante la composición de diferentes VNFs. A continuación se expresa una lista de funciones específicas que realiza NFVO[26]:

a) **Orquestación de Servicios**

- Gestión de plantillas para la implementación de servicios de red e instancias VNF
- Gestión del ciclo de vida de los servicios de red como por ejemplo, recolectar información sobre desempeño, eventos, iniciación y terminación de servicios.
- Validación y autorización de solicitudes de recursos NFV de Gestores de VNF.
- Gestión y evaluación de políticas para las instancias VNF y de servicios de red.

b) **Orquestación de Recursos**

- Validación y autorización de solicitud de recursos de NFVI de los gestores VNF.
- Gestión de recursos NFVI a través del dominio de infraestructura de los operadores, incluyendo la distribución, reservación y ubicación de los recursos NFVI.
- Soporta la gestión de la relación entre las instancias VNF y los recursos NFVI.
- Gestión de políticas y la ejecución de los servicios de red e instancias VNF.

4) **Repositorios de Datos:** Son bases de datos que mantienen diferentes tipos de información sobre NFV MANO. Cuatro tipos de repositorios se pueden considerar[27]:

- El catálogo de servicios de red es un conjunto de plantillas predefinidas, que definen cómo los servicios pueden ser creados y desplegados, así como las funciones necesarias para el servicio y su conectividad.
- El catálogo VNF es un conjunto de plantillas que describen el despliegue y las características operativas disponibles de VNFs.
- El repositorio de recursos NFVI contiene información acerca de los recursos disponibles NFVI / asignados.
- El repositorio de instancias NFV contiene información sobre todas las funciones y servicios durante su existencia.

4.6.2 Retos y Puntos de Investigación

- A. **Gestión de Recursos:** Los servidores que se utilizan para albergar VNFs tienen una cantidad limitada de memoria, cómputo y capacidad de almacenamiento. Y puesto que en la práctica estos servidores pueden ser distribuidos a través de múltiples dominios, donde la capacidad del enlace será también finita. Por lo tanto, para lograr un crecimiento escalar de NFV, los recursos físicos deben ser manejados de manera eficiente. Dinamismo, escalabilidad y automatización son características importantes que se deben aplicar a este tipo de gestión de recursos.

- B. **Gestión Distribuida:** Los enfoques actuales de MANO se enfocan principalmente en soluciones centralizadas, que pueden producir limitaciones de escalabilidad, especialmente en escenarios donde los servicios se extienden a través de múltiples dominios administrativos. Esto generaría la sobrecarga de comunicación y retardos en la recolección de datos asociados a un gran número de fuentes heterogéneas, para su análisis, lo que impide que estos procesos sean ejecutados con frecuencia. Como resultado, el retraso en el aprendizaje y conocimiento del estado de los servicios y los recursos no permite realizar operaciones de reconfiguración en línea.

- C. **Programabilidad e Inteligencia:** La idea principal de NFV es desplegar y mantener servicios complejos a través de recursos físicos heterogéneos. Para soportar su funcionamiento adecuado se debe desarrollar un conjunto de interfaces programables, las cuales se extenderán hasta la funcionalidad actual de SDN, fuera del ámbito de control de los recursos de conectividad simples. En la actualidad las soluciones SDN controlan la distribución del tráfico en la red de acuerdo con reglas de reenvío. Se requieren abstracciones adicionales que se aplican a la computación y almacenamiento de recursos para que las funciones de red, pueden crear instancias a través de múltiples tecnologías de los proveedores. Del mismo modo interfaces que permitirán al programa de forma dinámica la configuración de las funciones de control y, como por ejemplo, su colocación.

- D. **Interfaces e Interoperabilidad:** Uno de los principales objetivos de NFV es terminar con el enlace entre los proveedores de equipos y proveedores de servicios técnicos. Uno de los requisitos clave para esto es el apoyo a la

interoperabilidad. Si bien se han hecho progresos notables por el ETSI en la definición del marco NFV MANO y las interfaces, todavía hay mucho por hacer en cuanto a la definición de interfaces destinadas a apoyar la interoperabilidad entre diferentes proveedores con diferentes funciones. Por ejemplo, todavía existen proyectos y soluciones que utiliza un modelo personalizado de funciones y servicios. Esto significaría que, hasta que se definan las interfaces, es complicado realizar interoperabilidad entre diferentes operadores relacionados a un único servicio.

4.6.3 Proyectos Desarrollados Sobre NFV MANO

Cuando los clientes o personas interesadas en hacer uso de tecnologías que les permitan desplegar VNFs, necesitan tener una visión amplia en relación de las asociaciones existentes entre los distintos vendedores, y analizar las mejores opciones. Siempre revisando que dichas empresas brinden constantes mejoras, actualizaciones y soportes a sus plataformas, ya que en este entorno tan cambiante y evolutivo, muchos de los servicios prestados pueden quedar obsoletos y en desuso.

Además todo cliente está en la libertad de solicitar al proveedor, algunos aspectos importantes como:

- Listas de interoperabilidad verificadas y de soluciones certificadas
- Documentación validada sobre pruebas realizadas y desempeño de características.

En base a lo descrito, a continuación se presenta una lista de proyectos de código libre desarrollados que permiten el despliegue y orquestación de VNFs. Son proyectos que se encuentran en constante actualización y evolución, además que algunos de ellos realizan aleaciones estratégicas, con el fin de mantener interoperabilidad y compatibilidad entre las distintas tecnologías.

A. OPNFV

La Fundación Linux ha desarrollado un proyecto de código abierto denominado OPNFV¹³, el mismo que ha sido integrado por proveedores de servicios técnicos y proveedores. OPNFV fue pensado para validar y aplicar las especificaciones estándar existentes proporcionadas por ETSI, aportando mejoras a los proyectos de código abierto y desarrollo a nuevas

¹³ OPNFV es un proyecto de código abierto fundado y organizado por

funcionalidades necesarias tanto dentro OPNFV y a proyectos externos de exploración y producción. La primera plataforma que resulta de esta filosofía se lo denomina OPNFV Arno, la cual fue lanzada en junio de 2015.

OPNFV se centra principalmente en la construcción de infraestructura de NFV (NFVI) y Gestión de infraestructura virtualizada (VIM) mediante la integración de los componentes de los proyectos como OpenDaylight, OpenStack, almacenamiento Ceph, KVM, Open vSwitch y Linux. Estos componentes, junto con interfaces programables de aplicaciones (API) y otros elementos NFV forman la infraestructura básica necesaria para las funciones de red virtualizada (VNF), componentes de Gestión de Red y de orquestación (mano).

Actualmente OPNFV ha realizado una mejora a su software y lanza su nuevo proyecto ambicioso denominado **Brahmaputra**¹⁴, el cual trae una plataforma de funcionalidad NFV. Brahmaputra tiene mucha mayor estabilidad, sistematización e integración. Además implementa un riguroso proceso de integración de comunidades de socios y más de 30 proyectos aprobados que han contribuido nuevas capacidades, especificaciones y recursos de la comunidad a la plataforma.

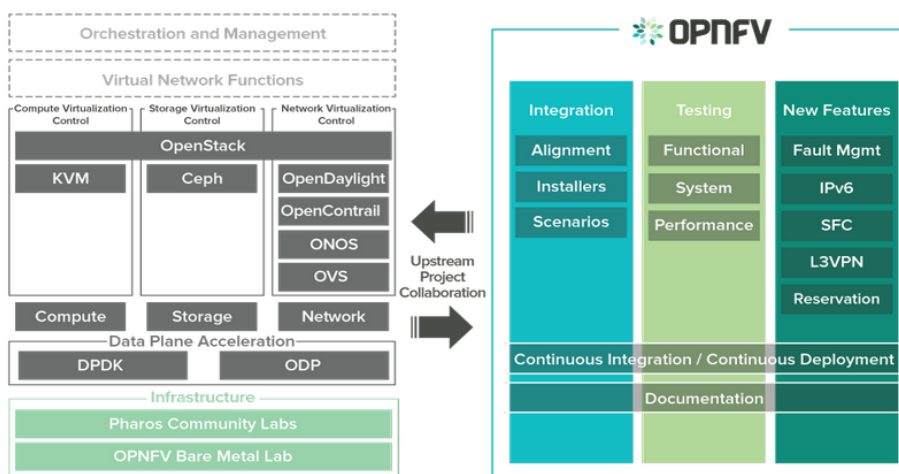


Figura 4.32. Arquitectura funcional de Brahmaputra.

B. OpenMANO

¹⁴ <https://www.opnfv.org/software>

OpenMANO¹⁵ es un proyecto de código abierto liderado por Telefónica, el cual tiene como objetivo la aplicación del marco ETSI NFV MANO, además de abordar aspectos relacionados con el rendimiento y la portabilidad mediante la aplicación de los principios de funcionamiento de Enhanced Platform Awareness (EPA)¹⁶. Como se muestra en la Fig. 4.33, la arquitectura OpenMANO está constituida por tres componentes principales: openmano, openvim, y una interfaz gráfica de usuario (GUI). Además, hay dos interfaces de línea de comandos (CLI) que se utilizan para interactuar con openmano y openvim.

Openvim es una implementación específica de VIM-NFV, la cual está directamente conectado con los nodos de cómputo y de almacenamiento de NFVI. Trabaja en conjunto con un controlador OpenFlow con el fin de crear la infraestructura de la topología de red, y hacer cumplir los principios de la EPA. También ofrece una interfaz basada en REST¹⁷ en dirección norte (API openvim), donde se ofrecen servicios en la nube incluyendo la gestión del ciclo de vida de las imágenes, instancias y redes. OpenMANO tiene una interfaz norte (openmano API), basado en REST, donde se ofrecen servicios MANO incluyendo la creación y eliminación de plantillas de VNF, instancias VNF, plantillas de servicio de red e instancias de servicios de red, para al final ser presentados de forma visual en su entorno gráfico GUI, donde se puede comprobar el funcionamiento de todas las redes e instancias creadas.

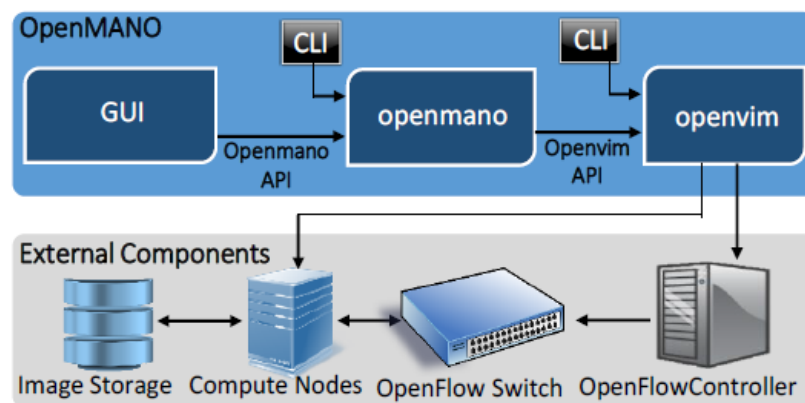


Figura 4.33. Arquitectura OpenMANO de Telefónica [27].

C. Open Source MANO (OSM)

¹⁵<http://www.tid.es/es/innovacion-de-largo-plazo/network-innovation/telefonica-nfv-reference-lab/openmano>

¹⁶ https://01.org/sites/default/files/page/openstack-epa_wp_fin.pdf

¹⁷ https://es.wikipedia.org/wiki/Representational_State_Transfer

Lanzada en Febrero de 2016, con el objetivo de favorecer el aumento del ecosistema de software abierto, que permita el despliegue rápido de servicios relacionados con las funciones de red de virtualización (NFV), los operadores como Telefónica, British Telecom, Telenor, y Telecom Austria se han unido con proveedores de soluciones de la nube NFV como Intel, Canonical, RIFT.io, y Mirantis, todos ellos como miembros fundadores de un nuevo proyecto de código abierto alojado por el ETSI llamado Open Source MANO (OSM¹⁸).

Este proyecto se enfoca en evolucionar un modelo conjunto ETSI NFV-ISG a un modelo de datos implementable, el cual se convertirá en la base de código de OSM para la dirección de multi usos de NFV. Esta realidad se ha venido trabajando con la mira de alcanzar un modelo de implementación uniforme que puedan ser desplegados por la comunidad de operadores al más bajo costo de integración.

OSM está compuesto por cuatro módulos específicos, los cuales son necesarios para poder brindar los servicios de comunicación extremo a extremo, dentro de las comunicaciones móviles. En la figura 4.34 se muestra la estructura arquitectónica que lo define[28].

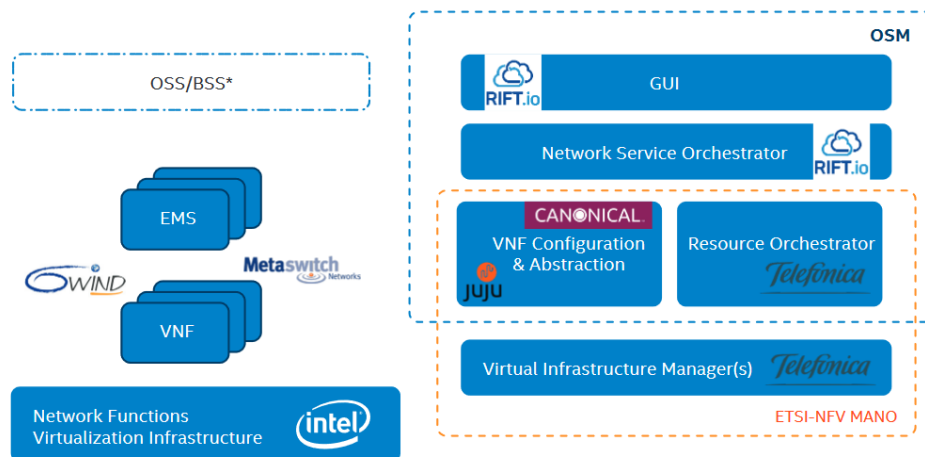


Figura 4.34. Vista simplificada de la arquitectura OSM[28].

¹⁸ <https://osm.etsi.org/>

- **Recurso Orquestador (RO):** Este servicio es proveído por Telefónica (OpenMANO), ésta capa es la responsable de procesar los requerimientos de locación de las instancias VNF y direccionar a VIM hacia una ubicación apropiada de computación, red y almacenamiento, para desplegar VNFs con sus interconexiones.
- **Configuración y Abstracción de VNF (CM):** Este servicio lo provee Canonical¹⁹ (Juju). Esta capa ofrece la gestión de la funcionalidad de VNF y es la responsable de los atributos de configuración de procesamiento. Juju brinda un modelo de aplicación genérica de código abierto, el cual permite a OSM concentrarse en retos específicos de orquestación.
- **Orquestador de Servicios de Red (SO):** El servicio de orquestación de redes (NSO) es brindado por RIFT.io²⁰, el cual es el responsable de la entrega de servicios de red extremo a extremo y además de configurar componentes VNF. NSO provee una interfaz simple para la gestión del ciclo de vida de los servicios de red (NS).
- **Interfaz de Usuario Gráfica (GUI):** Del mismo modo RIFT.io brinda la capa del sistema de visualización, en donde se puede observar el estado del entorno de orquestación. Este provee una simple e intuitiva forma para interactuar con NSO, mediante herramientas de fácil despliegue, creado y borrado de entornos VNFs.
- **Gestor de Infraestructura Virtual:** Esta capa también la provee Telefónica (OpenVIM)²¹. VIM es la capa en el sistema que gestiona los recursos de cómputo, redes, y almacenamiento del dominio infraestructural. La diferencia con las otras capas es que VIM está fuera del enfoque del proyecto OSM, pero es un servicio de gran importancia para un despliegue real. Aquí se pueden ocupar:

¹⁹ <http://www.canonical.com/>

²⁰ <https://www.riftio.com/>

²¹ https://osm.etsi.org/wikipub/index.php/OpenVIM_installation_to_be_used_with_OSM_release_0

- **OpenVIM:** Recurso de OpenMANO, el cual incluye controlador OpenDaylight SDN.
- **OpenStack Kilo**²²: Gestiona la infraestructura para despliegues de Metaswitch con una capa modular Neutron (ML2) la cual permite soportar configuración automática en el plano de datos de un switch.

D. **OpenStack:** OpenStack²³ Permite a los usuarios desplegar máquinas virtuales y otras instancias que manejan diferentes tareas con el fin de agilizar la gestión de un entorno de nube. Además permite realizar un escalado horizontal fácil gracias a los componentes autónomos que utiliza, quiere decir que las tareas que corren al mismo tiempo pueden servir fácilmente a más o menos usuarios sobre la marcha. También gestiona recursos computacionales, de almacenamiento y de red a través de un datacenter, administrado por medio de una API OpenStack.

Y lo más importante, OpenStack es un software de código abierto, lo que significa que todo aquel que desee puede acceder al código fuente, y será capaz de realizar cualquier cambio o modificación que crea necesaria, y de esta forma compartir libremente estos cambios de nuevo a la comunidad en general.

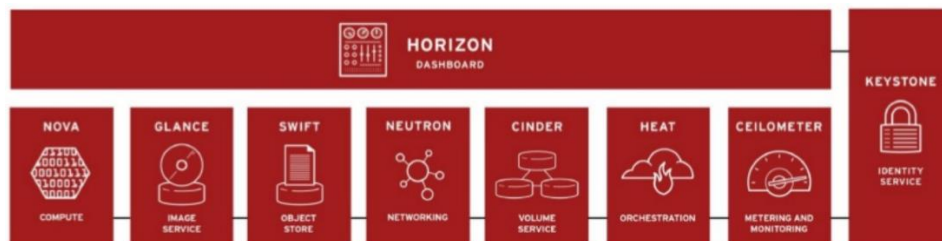


Figura 4.35. Arquitectura de OpenStack.

Existen en la actualidad varios proyectos que se están desarrollando por parte de empresas privadas como por ejemplo, Dell, Brocade e Intel, iniciaron una colaboración en conjunto con el fin de proveer servicios de telecomunicaciones basadas en

²² <https://www.openstack.org/software/kilo/>

²³ <https://www.openstack.org/>

soluciones NFV, teniendo como resultado el despliegue de vRouter Brocade²⁴ dentro de la plataforma NFV de Dell.

Algunos casos colaboran en el desarrollo de plataformas de código abierto, así como otras que lo realizan en forma de software propietario. Por tal motivo se ha generado un ecosistema con gran variedad de soluciones, pero sin objetivos claros de estandarización. En donde el proyecto de la ETSI – OSM, está iniciando un nuevo camino debido a que, como se lo mencionó, integra distintos componentes que son el resultado de varios proyectos de empresas relacionadas al enfoque de virtualización y orquestación de redes.

En el siguiente capítulo se desarrolla un caso práctico haciendo uso de algunas de las herramientas de código abierto que permiten desplegar y gestionar un escenario virtual.

²⁴ <https://www.sdxcentral.com/articles/featured/nfv-performance-test-validates-80-gbps-brocade-vyatta/2014/10/>

5 Caso Demostrativo

Para llegar a entender la filosofía de la teoría planteada sobre los procesos de virtualización de las funciones de red, aplicados a componentes de las redes centrales de las operadoras móviles, y como llegar a orquestarlos de una forma eficiente y dinámica, se plantea un caso práctico de demostración haciendo uso de una de las herramientas de MANO de código abierto, detallada en el capítulo anterior. Además para que el entorno sea más realista y didáctico se utiliza la herramienta de virtualización de redes sobre Linux, VNX²⁵. La misma que nos permitirá crear y desplegar el escenario que pretendemos virtualizar y gestionar de forma automática.

Antes de pasar a explicar el desarrollo del caso práctico, se explicará en qué consiste cada herramienta y cómo se ejecuta su funcionalidad interna.

5.1 Componentes Básicos de Escenario

La rápida evolución en la convergencia de las comunicaciones, ha venido desarrollando una gran variedad de proyectos que ofrecen mejorar los entornos de servicios, arquitecturas y costos para las distintas operadoras. La virtualización de funciones de red está progresando en propuestas, diseños, entornos de despliegue y gestión de VNFs. Como se detalló en la sección anterior, existen una gran variedad de propuestas, pero dentro de ellas la más reciente que se encuentra en fase de prueba y que promete mayor versatilidad y soporte es la liberada por ETSI, llamada OSM (Open Source MANO).

¿Por qué OSM?: OSM es una iniciativa que ingresa al mercado con soluciones a todos los retos que se han venido planteando en torno a la virtualización de funciones de red, por este motivo se lo puede resumir en tres razones por las que hace la diferencia, frente a las alternativas generadas por vendedores y operadoras.

1. Es un desarrollo de código abierto albergado y soportado por ETSI, lo cual genera varios beneficios a los clientes, ya que la plataforma se encuentra en constante desarrollo y evolución. Dando soluciones a todas las problemáticas que se generan en las comunicaciones actuales, pensando siempre en la flexibilidad, dinamismo y automatización en control y gestión de las redes.

²⁵ http://web.dit.upm.es/vnxwiki/index.php/Main_Page

2. Comprende e integra toda la complejidad de las actuales arquitecturas de red que se despliega en campo, como soportes de multi sitios, compatibilidad entre gestores de cloud (VIM) y separación de gestiones de recursos (RO), de los gestores de servicios (SO).
3. Es una iniciativa que parte de cero, está comprendida por la integración de varios proyectos como OpenMANO, Juju Charms, y Rift.ware, los mismos que sirven como “seed code”, para la implementación total de OSM. Lo cual ayuda a evitar en gran medida la sobre ingeniería debido a la abstracción que representa su implementación.

De igual forma OSM brinda varias facilidades dentro del entorno de orquestación como:

- **Estratificación:** Provee delimitación clara entre las distintas capas, módulos, y modelos de componentes de información, con el fin de diferenciar los procesos de operación y consumo.
- **Abstracción:** Presenta una clara diferenciación entre los distintos niveles de arquitectura, basada en la recomendación de ETSI-NFV.
- **Modularidad:** Debido a su división por capas, provee una clara modularidad con agentes APIs y plug-ins, permitiendo el enlace entre las distintas capas.
- **Simplicidad:** Intenta tener la más baja complejidad dentro de la gestión de la información y servicios. Por lo que OSM se encontrará en constante evolución de una forma coherente para mantener un control apropiado dentro de la arquitectura

Por estas razones, en el camino del desarrollo del presente proyecto se decidió poner en práctica OSM y crear un entorno sencillo en donde se pueda comprender el funcionamiento de los procesos de gestión y orquestación de instancias de red virtualizadas así como sus funciones. Esto nos permite mantenernos al día en la evolución de las propuestas de MANO sobre las NFVs. A continuación se detallan los componentes que conforman OSM y el escenario en general.

5.1.1 Herramienta VNX

VNX es una herramienta de código abierto desarrollada por la Universidad Politécnica de Madrid, la cual nos brinda la facilidad de desplegar grandes escenarios

de redes virtuales dentro de un entorno virtualizado que se encuentran interconectados por medio de túneles de capas 2 y 3, donde a su vez se pueden conectar hacia redes externas. Como se menciona en [29], VNX es una arquitectura modular basada sobre una API de control de máquinas virtuales, lo cual presenta grandes ventajas, como por ejemplo la introducción de libvirt²⁶ y Dynamips²⁷. La primera es una API de Linux para virtualización que soporta distintos hipervisores como KVM, QEMU, Virtualbox, et. Por lo que puede ser desplegado en varias plataformas de virtualización. Y la segunda es un emulador de routers Cisco

Su configuración y despliegue se lo puede hacer de forma sencilla mediante la ejecución de comandos, gracias a la inclusión de un demonio denominado ACED, el cual lee el fichero de configuración XML y configura los parámetros dictaminados.

En la figura 5.36, se muestra la arquitectura modular de VNX controlada por una de las APIs que involucra.

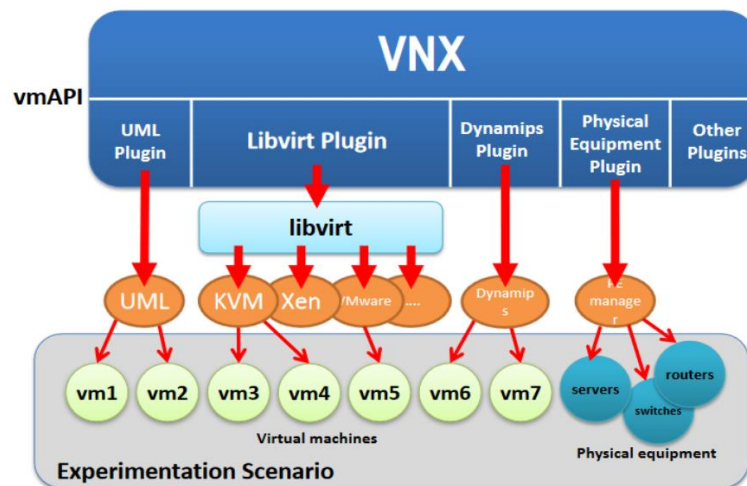


Figura 5.36. Arquitectura funcional de VNX [29]

Se plantean 4 sencillos pasos para la creación y despliegue de escenarios en VNX:

1. Planteamiento y diseño del escenario a desplegar
2. Especificación del escenario con editor XML y editor gráfico VNUML
3. Creación y ejecución del escenario virtual detallado en el fichero XML

²⁶ <https://libvirt.org/>

²⁷ <https://github.com/GNS3/dynamips>

4. Prueba e interacción con el escenario mediante la ejecución de comandos dentro de la máquinas virtuales desplegadas.

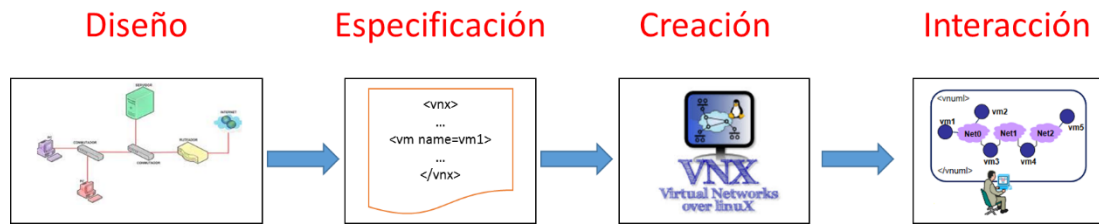


Figura 5.37. Pasos para creación de escenarios en VNX.

Una ventaja que se destaca de VNX, es permitir el despliegue y creación de varias máquinas virtuales a partir de un solo sistema de archivos (filesystem), mediante la utilización de la técnica “cow”, copy on write. Su función principal es admitir a varias máquinas virtuales compartir un mismo fichero “filesystem”. Como resultado genera ahorro en la capacidad de almacenamiento y a su vez escalabilidad en el despliegue de escenarios virtuales grandes.

5.1.2 Herramienta OPENMANO

Como se mencionó, en el capítulo anterior OPENMANO es una herramienta que permite una práctica implementación de cualquier arquitectura de red, para su gestión y orquestación, bajo las estandarizaciones de NFV ISG ETSI [30]. Como se muestra en la figura 5.38, Openmano esencialmente está constituido por 3 capas:

- **Openvim:** Hace referencia a la implementación de ETSI - VIM. No es más que el encargado de abstraer los procesos de virtualización dentro de los nodos de cómputo físicos. Además posee un controlador openflow, el cual le permite desplegar máquinas y redes virtuales dentro de servidores, así como la interconexión con equipos físicos de red. Su API openvim, ofrece servicios como la creación, borrado y gestión de imágenes, nodos, instancias y redes. Openvim es la primera capa sobre orquestación general de NFV, por lo que dentro de la arquitectura de NFVI, existen otros proveedores que prestan servicios de VIM, como por ejemplo Openstack.
- **Openmano:** Es el orquestador que está por encima de la capa de virtualización que puede ser openvim u openstack. Es una capa de abstracción que entiende de descriptores de VNF y de escenarios de red. Su

API openmano, ofrece interfaces basadas en REST²⁸ (Representational State Transfer), donde brinda servicios de creación, borrado y despliegue de plantillas e instancias VNF. Estas instancias están interconectadas por redes, que normalmente se las llama como de plano de datos. Además cuando se despliega un escenario de red se requiere conectar las interfaces de control de las VNF a una red “externa”, por ejemplo internet.

- **Openmano-GUI:** Es el componente superior de la estructura de gestión y orquestación de entornos NFV. Interactúa con el servidor openmano, a través de la API northbound, permitiendo la visualización de un entorno gráfico amigable para el usuario. Aquí se muestra todo lo desplegado en las capas inferiores, como equipos, redes, VNFs y escenarios.

Para un mejor acercamiento y detalle sobre la estructura y funcionamiento de openmano, se recomienda ingresar al sitio web del proveedor.

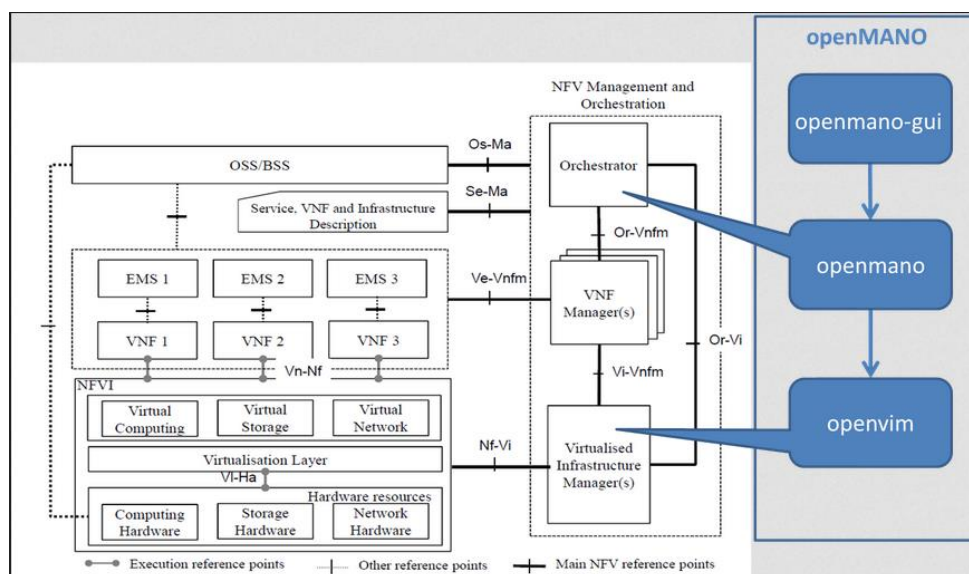


Figura 5.38. Arquitectura NFVI y aplicación de Openmano²⁹.

Modos de Operación de OpenMANO: Dentro de su estructura de configuración y despliegue OpenMANO, provee cinco modos de ejecución de su gestor de infraestructura virtualizada (vim):

²⁸ https://resources.docs.salesforce.com/sfdc/pdf/api_rest.pdf

²⁹ <https://github.com/nfvlabs/openmano>

1. **Normal:** Por defecto, OpenMANO necesita que se configure un controlador Openflow (OFC) y que se lo despliegue sobre equipos reales como switches y hosts, para un propósito general.
2. **Host Only:** Se lo utiliza cuando no se dispone de switch OpenFlow tampoco de su controlador (OFC), y se lo ejecuta sobre un nodo computacional. Aquí las conexiones de red del Plano de Datos se las debe hacer manualmente.
3. **Development:** Este modo es muy versátil ya que permite desplegar OpenMANO forzosamente sobre un cloud, ya que si no se dispone de ningún equipo físico y hardware de red especializado se lo debe hacer en la nube. Utiliza una red puente (bridge network) en lugar de una red de plano de datos OFC. Éste mismo bridge de desarrollo permite ser usado por todas las redes de plano de datos. El inconveniente de este modo es que se tiene un bajo desempeño del nodo de cómputo.
4. **Test:** El modo de prueba permite ejecutar sobre cualquier host con el fin de demostrar su funcionamiento y comprender las abstracciones que nos presenta sobre el despliegue de redes y VNFs dentro de máquinas virtuales. Se debe tomar en cuenta que en este modo no existe un despliegue real de máquinas virtuales y está diseñado para probar la API de OpenMANO.
5. **OF Only:** En este modo simplemente permite probar la integración del protocolo openflow, pero para ello se necesita de su equipo físico OpenFlow switch.

Para poder cambiar su configurar o escoger el modo de uso entre los distintos modos, se debe acceder al fichero de configuración de **openvim.cfg**, de OpenMANO.

5.1.3 Herramienta OPENSATACK

En la presente práctica se plantea ocupar openstack como vim, el cual se encargará de la gestión de la infraestructura virtualizada. Ahí se desplegará un escenario sencillo que deberá ser gestionado por Openmano.

Openstack se desarrolló para proveer una plataforma encargada de orquestar nubes computacionales a gran escala. Su tecnología tiene la capacidad de desplegar máquinas virtuales (VMs), así como redes para interconectar dichas máquinas, todo esto sobre un equipo estándar. Para lograr toda ésta orquestación, involucra un conjunto de

componentes que tienen funciones específicas que se llevan a cabo en el despliegue. A continuación se detallan:

- a. **Openstack Compute (Nova):** Controla la unión de la computación en la nube, es el componente principal de un servicio de infraestructura. Crea una capa de abstracción para la virtualización de recursos del servidor estándar, como la CPU, la memoria RAM, adaptadores de red y unidades de disco duro, con funciones para mejorar la utilización y la automatización.

La gestión de VMs como desplegar, cambiar el tamaño, suspender, detener y reiniciar el sistema, lo hace mediante la integración de un conjunto de hipervisores compatibles. También hay un mecanismo para almacenar en caché las imágenes de máquinas virtuales en los nodos de cómputo para el aprovisionamiento más rápido.

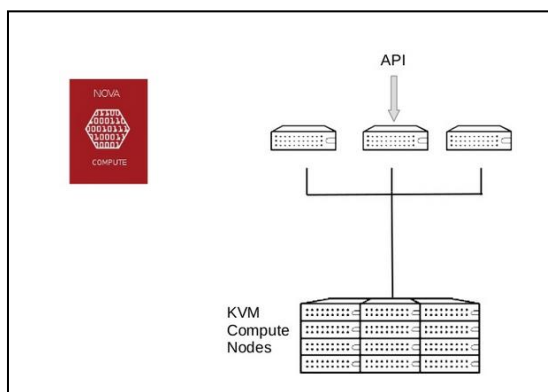


Figura 5.39. Componente Nova con hipervisor KVM.

- b. **Openstack Networking (Neutron)**³⁰: Neutrón, antes conocido como Quantum, provee la capacidad para gestionar redes locales con las características de LAN virtual (VLAN), DHCP, e IPv6. Los usuarios pueden definir las redes, subredes y routers para configurar su topología interna, para luego asignar direcciones IP y VLANs a dichas redes. Además se incorpora la característica de direcciones IP flotantes, las cuales permiten a los usuarios asignar direcciones IP externas fijadas a las máquinas virtuales. En el diagrama a continuación se muestra una configuración simple de red Neutrón, la cual usa un proveedor de red para conectividad hacia el internet, además tiene dos redes aisladas basadas en redes residentes (tenants) para proveer comunicación entre las instancias creadas.

³⁰<https://access.redhat.com/documentation/en/red-hat-enterprise-linux-openstack-platform/version-7/architecture-guide/#sec-networking>

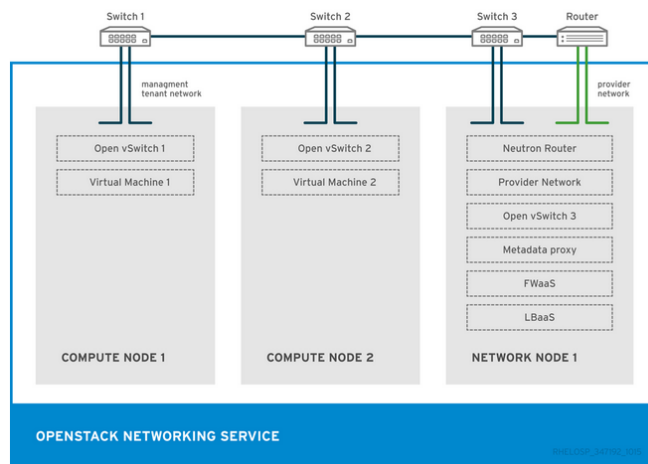


Figura 5.40. Diagrama de red Neutrón.

- c. **OpenStack Horizon:** Provee un portal propio basado en web, que permite interactuar con los servicios de openstack, como el despliegue de una instancia, asignación de direcciones IP y la configuración de controles de acceso. El diseño modular permite al tablero (dashboard) interactuar con otros productos tales como facturación, seguimiento y herramientas de gestión adicionales.

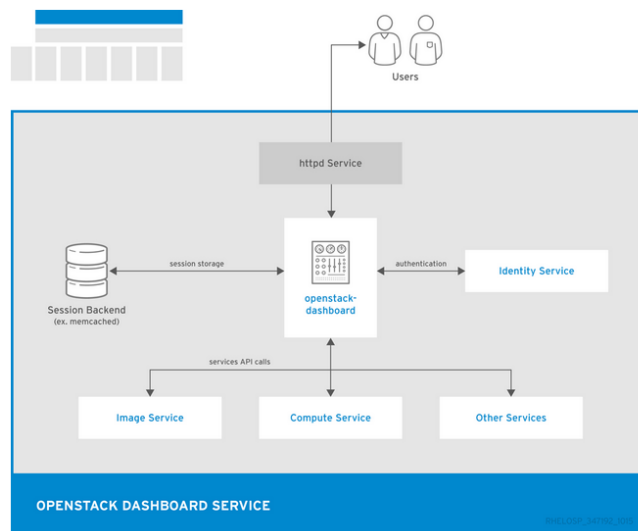


Figura 5.41. Arquitectura de Horizon³¹.

³¹<https://access.redhat.com/documentation/en/red-hat-enterprise-linux-openstack-platform/version-7/architecture-guide/#sec-networking>

Como se puede observar Openstack necesita de distintos componentes para su gestión de la capa de infraestructura, por lo tanto posee una arquitectura mínima para su despliegue y gestión, la misma que está compuesta por 3 nodos: Nodo Controlador, Nodo de Red y Nodo Computacional. En cada uno de dichos nodos se ejecutan distintos servicios como:

- En el nodo controlador se ejecuta el servicio de identidad, Servicio de imágenes, gestión de cómputo y redes, plug-in de red, y el tablero (dashboard) de visualización. También incluye los servicios de apoyo como base de datos, y el Protocolo de tiempo de red (NTP). Opcionalmente, el controlador también ejecuta porciones de almacenamiento, almacenamiento de objetos, Servicio de Base de Datos, orquestación, y telemetría. Estos componentes proporcionan funciones adicionales para su entorno. [31]
- En el nodo de red se ejecutan servicios como el plug-in de Networking, agentes de capa 2 y capa 3, así como varios agentes que proveen y operan redes residentes (tenants). Los servicios de capa 2 incluyen aprovisionamiento de redes virtuales y túneles. En cambio, servicios de capa 3 incluyen enrutamiento, NAT y DHCP. Este nodo también se ocupa de la conectividad externa (Internet) para instancias o máquinas virtuales residentes (tenants).
- El nodo de cómputo es el encargado de ejecutar la parte del hipervisor, que permite operar las máquinas virtuales residentes (tenants) o las instancias. Por defecto éste nodo utiliza KVM como hipervisor. También ejecuta el plug-in de red y el agente de capa 2, los cuales operan redes residentes (tenants) e implementar grupos de seguridad. El usuario puede ejecutar más de un nodo de cómputo.

En el siguiente diagrama se muestra la arquitectura básica de un escenario openstack, el cual si se despliega debe involucrar todos sus servicios básicos descritos anteriormente, así como sus redes internas y externas para su interoperabilidad y orquestación. [31]

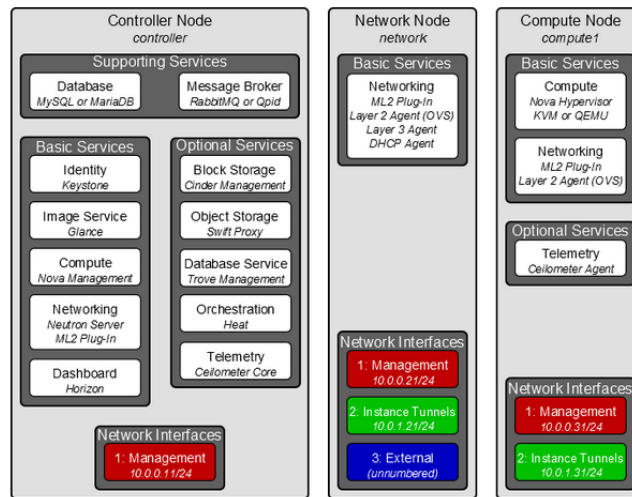


Figura 5.42. Arquitectura básica de OpenStack con Neutrón.[31]

5.1.4 Herramienta JUJU

Juju³² pretende ser una herramienta de despliegue de servicios y orquestación que permite la misma clase de colaboración y facilidad de uso que hoy se ve alrededor de la gestión de componentes y paquetes en torno a los servicios. Con juju, diferentes desarrolladores son capaces de crear servicios de manera independiente, y hacer que estos servicios se comunican a través de un protocolo de configuración simple. Entonces, los usuarios pueden llevar el resultado de la combinación de ambos autores y muy cómodamente implementar estos servicios en un entorno, asemejándose a la forma cómo las personas son capaces de instalar una red de paquetes con un único comando a través de APT.

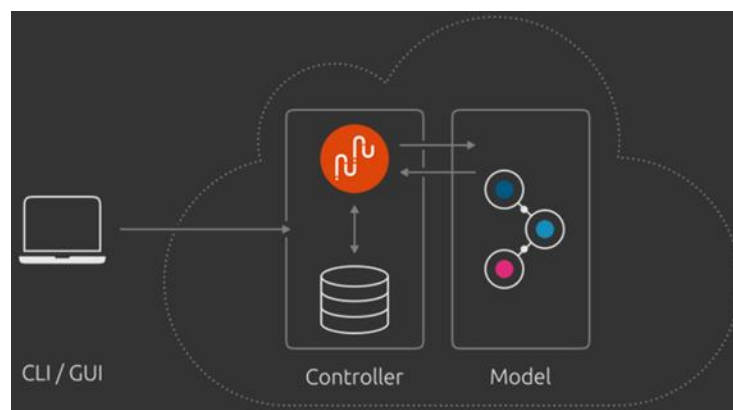


Figura 5.43. Estructura básica de juju.

³² <https://jujucharms.com/docs/stable/about-juju>

Juju es un software del tipo código libre. Es un modelo universal para la arquitectura orientada al despliegue de servicios. Juju permite desplegar, configurar, gestionar y mantener servicios de gran escala con rapidez, tanto en nubes públicas, como en servidores físicos, OpenStack, y contenedores. Además se puede utilizar Juju desde la línea de comandos o a través de su interfaz gráfica de usuario.

Con Juju, diferentes desarrolladores son capaces de crear fórmulas de servicios, llamados "charms", de forma independiente, y hacer que estos servicios coordinen su comunicación y configuración a través de un protocolo sencillo, de esta forma un "charm" dice a juju cómo gestionar un servicio. Por ejemplo, después de servicios como "la base de datos MySQL" o "wordpress", que se hayan implementado, juju puede definir las relaciones entre ellos ("Wordpress necesita mysql") y exponer algunos servicios para el mundo exterior.

Charms están compuestos de metadatos, datos de configuración, y los ganchos (hooks) con algunos archivos de soporte adicionales. Charms utilizan "hooks", normalmente escritas como scripts, los cuales juju invoca para gestionar el ciclo de vida del servicio. Hay dos tipos de hooks: Hooks de unidad y Hooks de relación. Los primeros incluyen instalar, configurar, iniciar, actualizar-charm, y detener. Los segundos, hooks de relación, incluyen juntar, cambiar, desplegar, y romper relaciones. Del mismo modo, el hecho de definir relaciones en un charm simplifica la implementación mediante la lógica de lo que un charm puede y no, conectarse. Además charm contiene todas las instrucciones necesarias para implementar y configurar los servicios basados en la nube. Los charms a disposición del público se encuentran en la tienda en línea que representa el conocimiento de expertos conocida como DevOps³³. Si no puede encontrar lo que se está buscando o si lo que está disponible no satisface sus necesidades a continuación, toda persona es libre de escribir el suyo propio.

De la misma forma que toda plataforma moderna de gestión, juju también puede ser utilizado junto con algún orquestador para implementaciones físicas. Así, por ejemplo, si se tiene un charm para Hadoop, puede usarlo para instalar Hadoop a través de miles de servidores con el orquestador.

Modelado de Servicio: en entornos actuales, los servicios son raramente desplegados aisladamente, incluso aplicaciones menos complejas como servidores web, requieren de los servicios de una base de datos. De manera obligatoria en entornos más complejos, se necesita integrar, configurar y conectar mayor cantidad de

³³ <https://es.wikipedia.org/wiki/DevOps>

servicios. Por tal motivo el servicio de Juju provee herramientas para gestionar dichos servicios.

Aplicaciones específicas tales como dependencias, escalado, y operación de eventos como copias de seguridad, actualizaciones y opciones de integración son encapsuladas o integradas en los charms. Por lo tanto esta estructura y conocimiento puede ser compartido entre los miembros del equipo, reutilizados en todas partes desde portátiles hasta máquinas virtuales y la nube, para ser compartida con otras organizaciones.

Charm define todo lo que se conoce acerca de la implementación colaborativa de un servicio en particular. Todo lo que se tiene que hacer es usar cualquier charm disponible (o creado propio), y el correspondiente servicio será desplegado en poco tiempo, en cualquier nube, servidor o máquina virtual. En la figura a continuación se observa de forma general la estructura de la arquitectura de funcionamiento de juju.

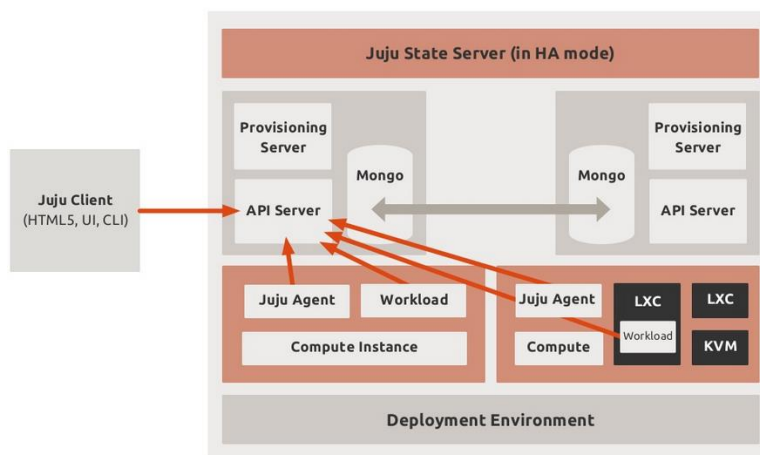


Figura 5.44. Arquitectura de funcionamiento de Juju.

5.1.5 Herramienta RIFT.ware

RIFT.io desarrolló RIFT.ware, que según [32], menciona que fue la primera plataforma de NFV de la industria de código libre, que simplifica el desarrollo, despliegue y gestión de servicios de red virtuales a escala Web. Como se ha mencionado en los capítulos anteriores, las funciones de red necesitan incorporar las prácticas y métodos de funcionamiento de los centros de datos de hiperescala, con el fin de cumplir el objetivo principal de las operadoras sobre la reducción de manera significativa el costo de operación y despliegue, además de proveer una entrega ágil de servicios. Como muchas de estas prácticas son comunes a todas las funciones de la red, se deduce que la plataforma de centro de datos tiene que seguir la tendencia evolutiva

de las últimas décadas mediante la incorporación de nuevas técnicas, de escala Web. Dentro de los beneficios de Rift.ware se pueden mencionar los siguientes:

- **Simple:** RIFT.ware abstrae la complejidad de la red subyacente y la infraestructura de computación de la nube para reducir el desarrollo, la integración, costos y el esfuerzo de integración. Presenta la capacidad de que constructores VNF puedan migrar de los equipos dedicados hacia servicios virtualizados en poco tiempo.
- **Seguro:** Funciones de red pueden aprovechar las ventajas de colocar sobre plataformas confiables la carga de trabajo segura y así generar soporte en cifrado para obtener un nivel de seguridad dentro de las empresas.
- **Hiperescala:** RIFT.ware permite la posibilidad de iniciar con una única VNF en una máquina virtual y a continuación escalar a miles de máquinas virtuales en múltiples sitios y nubes híbridas con capacidad de recuperación de nubes a gran escala.
- **Código Abierto:** Rift.ware es una plataforma de código libre que evita la dependencia de un proveedor, facilita interoperabilidad, así como proporcionar opciones y flexibilidad.

Plataforma Rift.ware de Virtualización de Servicios de Red:

La Plataforma NSV se diseñó bajo las normas establecidas por ETSI y está construida con las más modernas tecnologías web, con el fin de ofrecer automatización, capacidades de hiperescalabilidad y servicios de red virtualizados. RIFT.ware engloba virtualización de cómputo y NFVs para implementar dentro de cualquier organización y establecer comunicaciones en cualquier infraestructura de nube a toda escala. RIFT.ware tiene cuatro componentes principales ilustrados en la figura 5.45 y se describen a continuación.

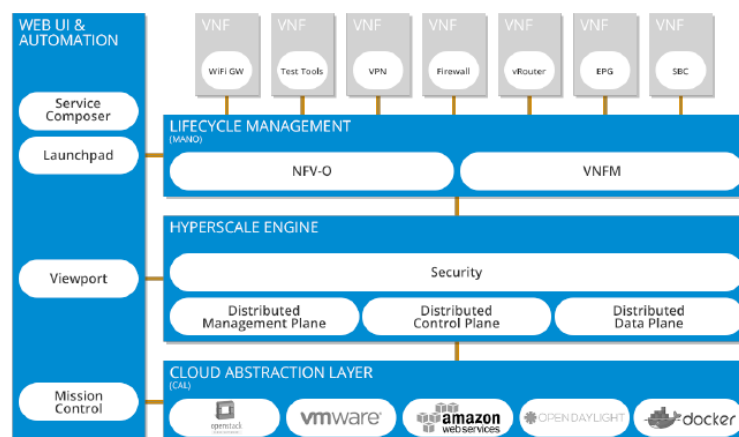


Figura 5.45. Estructura funcional de software Rift.ware. [32]

Tres Pre-requisitos de la Plataforma VNF: En el despliegue de una plataforma ampliamente adoptada en la industria, siempre necesitará cumplir estas tres condiciones de importancia:

1. La plataforma debería ser construida usando software de código abierto, esencialmente desarrollada y mantenida por una amplia comunidad abierta. El problema surge cuando se ocupa una plataforma de propietario, debido a que cuando existe alguna actualización o innovación en la misma, como por ejemplo la publicación de APIs, nadie más puede tener acceso para modificarla según necesidades locales. Por lo tanto una plataforma de código abierto puede ser implementada sin costos de licencia y ser modificada por terceras partes que se encuentren participando en el desarrollo de la comunidad.
2. La plataforma debe ser de tipo modelo impulsado (model driven), asegurando su programabilidad y extensibilidad. Las capacidades de la plataforma deben ser modeladas en función de un lenguaje de modelado de datos estándar, por tanto pueden ser accedidos y reusados por desarrolladores de VNF, sin la necesidad de escribir código de integración. Las instancias de modelo pueden ser configuradas en capas de alto nivel de abstracción. Todo esto genera que la plataforma sea extensible, así cuando, cada vez que se libere una nueva capacidad, ella es capaz de implementarla inmediatamente.
3. Para propósitos de desarrollo de VNF, la plataforma debería ser ejecutada en la nube como un PaaS (Platform as a Service), alojada por una organización de vendedor neutral de VNF. Además los operadores podrían implementar una instancia de la plataforma como un entorno de tiempo de ejecución en su propio NFVI. De esta forma VNFs se montarán sobre la plataforma y podrán ser seleccionadas de un catálogo PaaS.

Rift.ware Gestión del Ciclo de Vida: La capa de gestión del ciclo de vida forma los recursos informáticos y redes virtuales dentro de una topología estructurada, que puede soportar de uno a cientos VNFs y servicios de red. La gestión del ciclo de vida es totalmente compatible con las especificaciones de gestión y orquestación de NFV ETSI.

Rift.ware fue diseñada para escalar tanto en servicios de redes, como en sus gestiones. Por tal razón contiene la administración de los componentes de VNF (VNFM) y la orquestación de NFV (NFVO), referentes a la arquitectura propuesta por ETSI.

- **Gestor VNF (VNFM):** Se encarga de supervisar el ciclo de vida de las instancias VNF, como:
 - Inicializado de VNF desde su descriptor y su gestión
 - Escalado y disminución de VNFs
 - Monitoreo y recolección de parámetros que determinan el estado de una VNF

- **Orquestador NFV (NFVO):** Administra el ciclo de vida de los servicios de red y orquesta los recursos a través de múltiples gestores de infraestructuras virtualizadas:
 - Incorpora nuevos servicios de red y paquetes de funciones de red virtuales
 - Gestión global de los recursos, tanto topologías de red físicas como lógicas
 - Controla la gestión de políticas relacionadas con la escalabilidad, fiabilidad y alta disponibilidad para las instancias de servicio de red
 - Autoriza peticiones de recursos de infraestructura de funciones de red virtualizadas

Rift.ware Características MANO: Además Rift.io actualmente es uno de los miembros fundadores del proyecto de la comunidad Open Source MANO (OSM³⁴), integrado por los proveedores de servicios y vendedores de soluciones centrados en proveer un proceso abierto de administración y orquestación alineados con la gestión de NFV-ETSI y las especificaciones de orquestación (mano). RIFT.ware contribuyó con código base que incluye la capa de red de servicio orquestación (NSO), que es responsable de la prestación de servicios de red extremo a extremo y dirige la prestación de dichos servicios a través de los componentes de orquestación de recursos y configuración de VNF. Además RIFT.io también proporcionó la capa de interfaz gráfica de usuario y el sistema de visibilidad (GUI), lo que genera una visión única de cómo configurar y revisar el estado del entorno. [33]

RIFT.ware genera soportes mejorados en su plataforma agregando atributos para asignar de forma inteligente la carga de trabajo para proveer un entorno virtual mejorado. Los atributos que posee se detallan a continuación:

³⁴ <https://osm.etsi.org/>

- **Multi-Cloud:** Valida, despliega VNFs y cadenas de servicios de red en cualquier sistema de gestión de la nube. Además aprovecha las capacidades únicas del entorno de nube.
- **Open APIs y Arquitectura Modular:** Arquitectura de plugins multi lenguajes con interfaces estándar bien definidos de la industria. Ofrece integración con cualquier plataforma de orquestación de terceros, herramienta de análisis, software de gestión de las nubes, plataforma de SDN, y función de red. RIFT.ware proporciona un entorno de múltiples proveedores totalmente automatizado de soporte extremo a extremo.
- **Automatización Extremo a Extremo:** Proporciona la capacidad de creación flexible de políticas para diseñar servicios de red automáticos.
- **Interfaz de Usuario Gráfica:** La interfaz de usuario provee una simple e intuitiva forma de interactuar con la plataforma MANO de Rift.ware. Crea VNFs seleccionados, e inicia instancias de servicios de red del tablero de una forma fácil como navegar por una tienda de aplicaciones en línea. Además la interfaz de usuario también proporciona estadísticas en tiempo real sobre el uso de VNFs, servicios de red, y una vista detallada de las topologías de red de cómputo.

5.2 Propuesta de Escenario Práctico

El escenario de experimento que se plantea involucra la utilización de la plataforma de orquestación Open Source MANO (OSM), la misma que se encargará de gestionar un escenario desplegado en la plataforma OpenStack, la misma que emula un sencillo entorno de cloud-computing. La idea del experimento consiste en acoplar estas dos plataformas y que puedan comunicarse dinámicamente, tratando de plasmar la idea general sobre la virtualización de funciones de red, entender el despliegue sobre la nube y su gestión en un entorno amigable y versátil.

Simultáneamente el despliegue del escenario de experimento se lo desea hacer de forma automática, para ello utilizamos la herramienta VNX que nos permitirá integrar a todos los módulos dentro de su entorno. Con esto se pretende facilitar el uso de OSM, debido que al ser una plataforma modular, tiene un grado de dificultad, al momento de su ejecución y posterior entendimiento.

En la figura a continuación se muestra la estructura arquitectónica del actual escenario, estructurado y detallado por capas de despliegue de los distintos componentes que involucran. Ésta arquitectura de capas está basada en las normas de la ETSI-NFV.

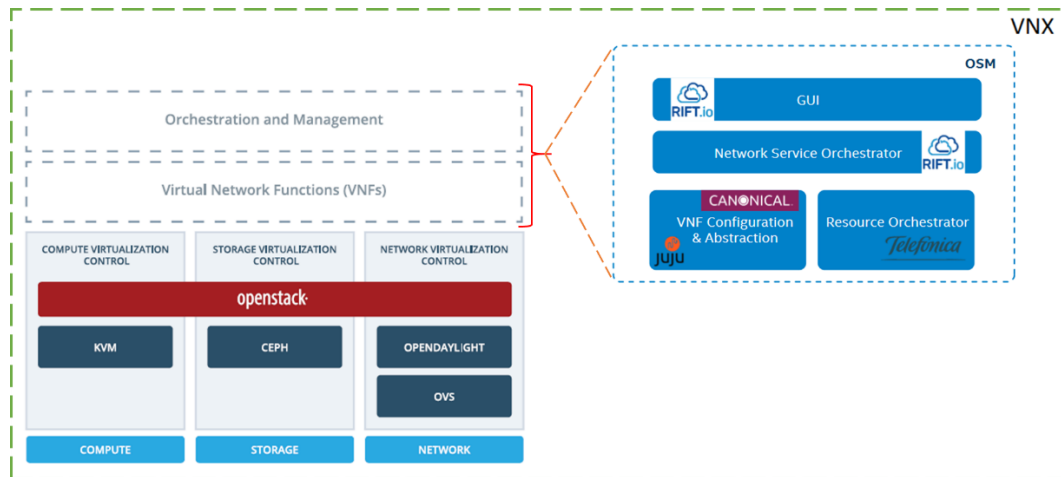


Figura 5.46. Arquitectura por capas de escenario experimental.

Como se puede observar la estructura total del escenario está dividida por dos grupos marcados, el primero que hace referencia a NFVI, el cual está implementado bajo la plataforma OpenStack, en donde se gestionará los recursos físicos para implementar máquinas virtuales. Y el segundo grupo está conformado por OSM, encargada de gestionar las capas superiores para orquestación.

5.2.1 Escenario Openstack

El escenario está compuesto por cuatro máquinas virtuales, que son: Un controlador (Controller), el mismo que está basado en LXC³⁵, el cual es una tecnología que permite virtualizar a nivel de Sistema Operativo. Un nodo de Red (Network), para gestión de los nodos de cómputo, y finalmente posee dos nodos de cómputo (Compute), todos ellos basados en KVM³⁶. Este escenario de red se lo despliega sobre Ubuntu 14.04.04 LTS y OpenStack³⁷. La información más detallada sobre su instalación y manejo independiente sobre el escenario, se encuentra pública en el sitio web de VNX.

La arquitectura del escenario base se muestra en la figura 5.47. Se detallan sus interfaces de conexión así como sus direcciones IP establecidas. En cada nodo se implementan sus funciones especiales para el funcionamiento del escenario.

³⁵ <https://linuxcontainers.org/>

³⁶ http://www.linux-kvm.org/page/Main_Page

³⁷ <https://www.openstack.org/software/kilo/>

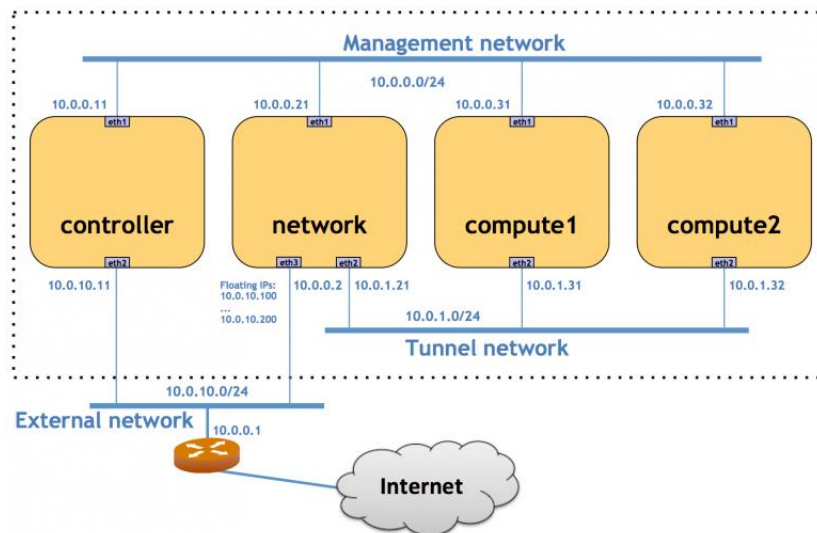


Figura 5.47. Escenario OpenStack Kilo.

Nodo Controller: Es el nodo principal del escenario, al cual se le deben agregar funciones que gestionen tanto la computación como la red de la nube.

- **Openstack Compute:** Estos componentes permiten interactuar con identidades Openstack como imágenes de discos y servidores, además de proveer al usuario el panel “dashboard” como una interfaz de gestión. Para lograr esto se necesita de algunos componentes como:
 - **Nova-api-service:** Acepta y responde al usuario final las llamadas a la API de cómputo. También inicia actividades de orquestación como ejecutar instancias.
 - **Nova-api-metadata service:** Gestiona peticiones de metadatos de las instancias y se lo ocupa cuando se ejecuta en modo multi-host³⁸.
 - **Nova-compute service:** Un demonio que crea y termina instancias de máquinas virtuales a través de APIs del hipervisor.
 - **Nova-network worker daemon:** Acepta tareas de red, como establecer y configurar interfaces de enlaces (bridging) o cambia reglas de las tablas IP.

Existen muchos más componentes como nova-consolauth daemon, nova-scheduler, nova-conductor, etc. Los mismos que cumplen funciones específicas como autorización, programación y envío de certificados al momento de su ejecución.

³⁸ http://docs.openstack.org/admin-guide/networking_arch.html

- **Openstack Network:** El servidor de redes utiliza el daemon neutrón-server para exponer la API de red y habilitar la administración del plug-in de red configurado. Por lo general, el plug-in requiere el acceso a una base de datos para el almacenamiento persistente. Se puede desplegar el componente Network en el mismo host que almacena al nodo Controller, sin embargo es un componente totalmente autónomo y puede ser ejecutado en un nodo aparte. Incluye algunos de los siguientes agentes:
 - **Openvswitch:** Componente que genera un switch virtual
 - **Neutron-openvswitch-agent:** Agente que permite gestionar al openvswitch
 - **Neutron-server:** Acepta y direcciona las peticiones API al plug-in adecuado de red.
 - **Neutron-agent:** Se ejecuta en cada hipervisor para desempeñar la configuración del vSwitch local.
 - **Neutron-dhcp-agent:** Provee servicios dhcp para redes tenant, requeridos por ciertos plug-ins.
 - **Neutron-l3-agent:** Provee reenvío l3/NAT con el fin de facilitar el acceso a redes externas para VMs sobre redes tenant.
 - **Neutron-metering-agent:** Brinda medida de tráfico l3 para redes tenant.

Existen más agentes como neutrón-haproxy y neutrón-lbaas-agent que permiten balancear y gestionar cargas.

Nodo Compute: Es un componente importante de la infraestructura como servicio (IaaS). Al igual que los otros nodos, requiere de agentes y componentes internos para su funcionamiento, como:

- **Openvswitch:** Componente que genera un switch virtual
- **Componente KVM:** Es una solución completa de virtualización basada en Linux.
- **Nova-compute:** Permite gestionar al hipervisor KVM
- **Neutron-openvswitch-agent:** Es un agente que permite la gestión de openvswitch.

A continuación en la figura 5.48 se muestra la arquitectura mínima de un cloud Openstack, en donde se describen los componentes y agentes que cada nodo debe poseer para su funcionamiento integrado.

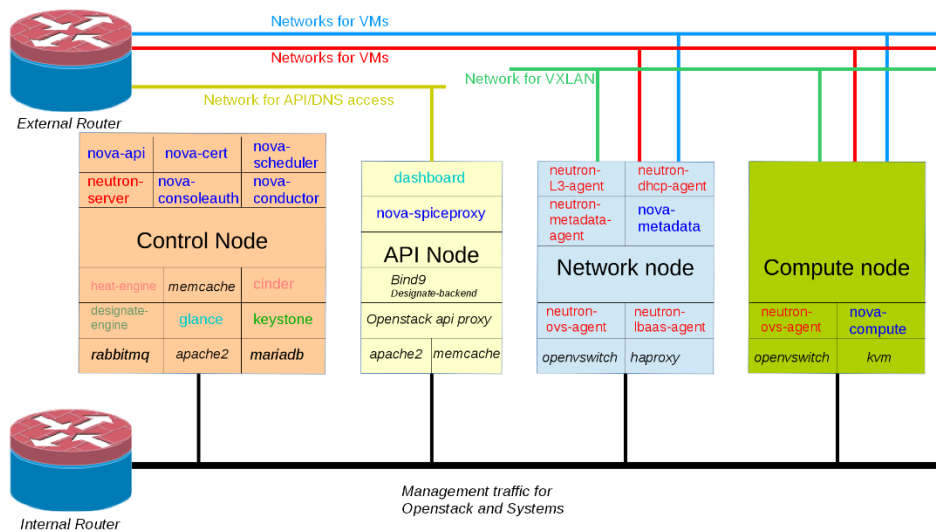


Figura 5.48. Arquitectura mínima de Openstack con agentes.

5.2.2 Escenario OSM

En teoría OSM puede soportar cualquier tipo de escenario real donde exista comunicación extremo a extremo, y sobre ella se lo puede cargar servicios como los brindados por IMS. Pero como toda plataforma de gestión y orquestación de recursos virtuales, necesita una infraestructura mínima, la misma que se muestra en la figura 5.58.

Para una implementación sencilla de OSM, la plataforma necesita de tres máquinas virtuales para el despliegue de cada uno de los componentes de las capas como VIM, CM, SO, y RO. Los cuales se interconectarán por medio de redes de gestión y redes lógicas. También se los puede enlazar por medio de equipos físicos como por ejemplo, switch. El cual serviría como el gestor de las comunicaciones entre la plataforma OSM y la VIM. Las redes de comunicación se las divide de la siguiente manera [28]:

- Red de Infraestructura de Datacenter, la misma que es utilizada por VIM para comunicarse con los nodos de cómputo
- Red de Gestión Telco/VNF, la misma que es equivalente a una red de administración de telecomunicaciones tradicional, y es usada por CM para configurar VNFs
- Redes de Plano de Datos, son las que se encuentran entre VMs, y usan interfaces basadas en SR-IOV o Puertos Físicos de comunicación.

- Otras Redes Virtuales, son aquellas que se colocan entre VMs y usan puentes Linux (bridges), switches virtuales como OVS, etc., cuando las VMs se encuentran localizadas en diferentes nodos de cómputo.

A continuación en la figura 5.49, se muestra la arquitectura mínima que debe poseer OSM para su despliegue. Esta arquitectura es la presentada por OSM, en donde hace uso de un switch físico, el mismo que se encarga de enlazar las comunicaciones entre los distintos componentes de la plataforma. Está destinado a manejar simultáneamente las distintas redes, viniendo a ser como un nodo de gestión de red.

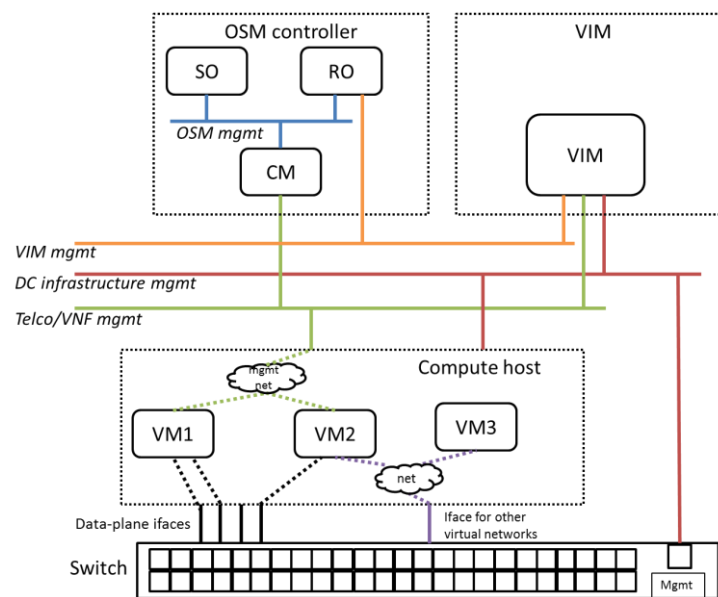


Figura 5.49. Arquitectura mínima de plataforma OSM con Switch físico³⁹.

La idea principal de la práctica es implementar todo el escenario en conjunto virtualizado, es decir, no hacer uso de equipos de red físicos, como por ejemplo switch, y en su lugar utilizar agentes y módulos virtuales que nos permitan reemplazarlos dentro del escenario virtual con el fin de obtener una escenario didáctico, manejable y de fácil entendimiento para los usuarios. Junto con la ayuda de VNX se intenta aligerar el proceso complicado y tedioso de crear e instalar todos los componentes sobre máquinas virtuales, más la eliminación del uso de un equipo de red físico, como switch. A continuación en la figura 5.50 se muestra la idea general de implementación del escenario sobre un entorno totalmente virtualizado. Posteriormente se irán detallando todos sus puntos de conexión y comunicación dentro del mismo.

³⁹ https://osm.etsi.org/wikipub/index.php/Release_0_minimal_infrastructure

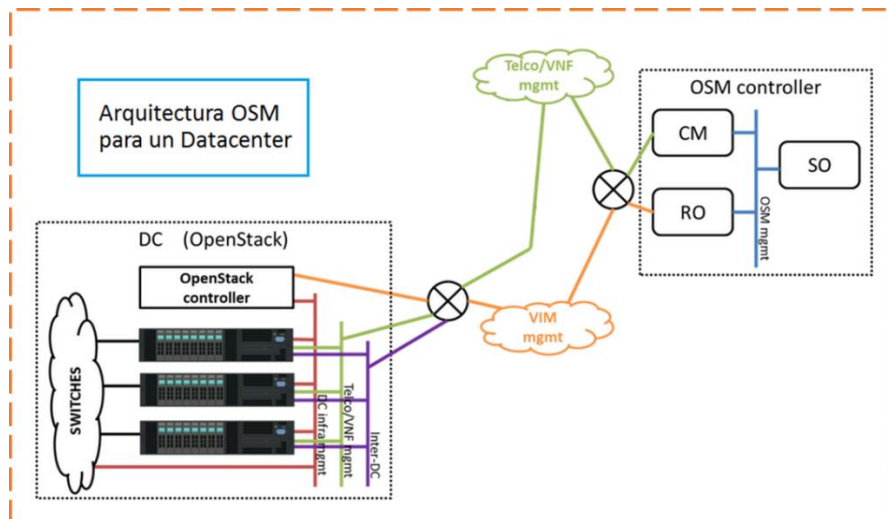


Figura 5.50. Escenario virtualizado de OSM junto con Openstack.

5.2.3 Detalles del Modelo de Datos

El concepto básico de OSM es gestionar y orquestar funciones de redes virtualizadas, pero para ello, como se ha explicado, cada uno de sus componentes cumplen funciones específicas para lograr dicho objetivo. Dentro del proceso de orquestación de las distintas capas, se debe tomar en cuenta como OSM gestiona a las VNFS.

OSM soporta descriptores de virtualización de funciones de red (VNF) y de servicios de red (NS). Los modelos de datos de estos descriptores están especificados ampliamente en la norma de la ETSI MANO. El componente encargado de definir y gestionar los descriptores es RO (OpenMANO).

- **Descriptor de Virtualización de Funciones de Red [VNFD]:** En primer lugar VNF es una función de red basada en software que puede ser desplegada sobre un datacenter NFV. Mientras que VNFD define los recursos requeridos para desplegar una VNF. Además de incluir varios componentes que son parte del VNF, define la información necesaria para la configuración de las VNFS.
- **Escenarios de Red [NS]:** Definen las topologías e interconexiones de las VNFS. De forma general sus interconexiones se realizan a nivel 2 (L2) de red y pueden ser configuradas como punto a punto o punto a multipunto.

- **Descriptor de Servicios de Red [NSD]:** Describe a un conjunto de VNFs interconectadas para desplegar un servicio de red que está compuesto de múltiples VNFs.

Los descriptores de OpenMANO se utilizan para declarar las propiedades principales de una VNF o de una NS. De forma más sencilla se puede entender que un VNFD es un fichero provisto por un vendedor para describir las propiedades de su VNF con el fin de cumplir sus necesidades para las que fue diseñado, mientras que un NSD es un fichero creado por el operador que provee servicios, el cual describe la topología de una VNF.

Descriptor VNF [VNFD]: Estos ficheros se encuentran disponibles para ser utilizados y modificados por cualquier persona o vendedor que se encuentre interesado en desplegar servicios virtualizados. El modelo VNFD contiene las siguientes propiedades principales:

- Nombre único de la VNF
- Conexiones externas (interfaces) de la VNF para poder ser conectada dentro de un NS o hacia otras VNFs y/o redes.
- Conexiones internas que definen como los componentes internos (VM) de una VNF son interconectados, solo cuando se utilizan varias VMs.
- Lista de componentes (VMS) que comprenden el VNF, el cual detalla cada componente sus capacidades de CPU, memoria, interfaces, vCPUs, RAM, etc.

A continuación en la figura 5.51 se muestran los dos escenarios básicos de VNFD que se pueden utilizar, simple o múltiple. En las gráficas se detallan las conexiones de red e interfaces necesarias para su interconexión y despliegue:

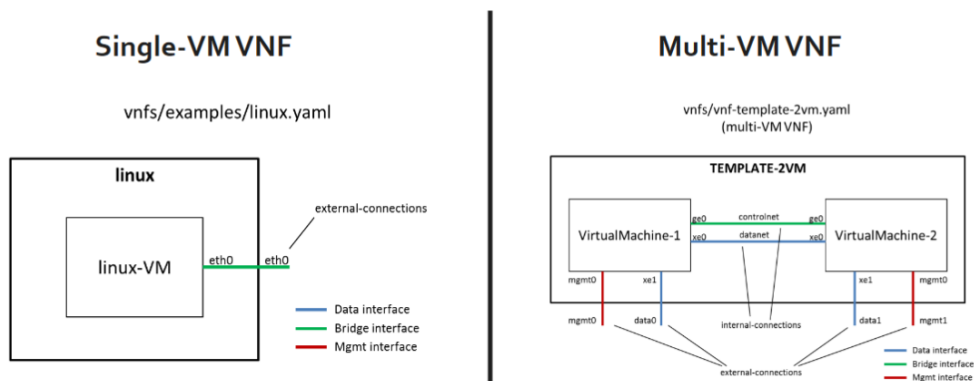


Figura 5.51. Especificación de estructura interna de VNFs.

A las VMs dentro de VNFD se las denomina como VDUs (Virtual Deployment Units), las mismas que se interconectan por medio del link VL (Virtual Link). Estas conexiones internas se las establece por medio de los CPs (Connection Points). En consecuencia el fichero de VNFD que es de formato YAML⁴⁰, contiene la lista de las VDUs y de las VLs que las interconecta. En la siguiente figura se muestra la estructura optada por ETSI para detallar la arquitectura de un VNFD.

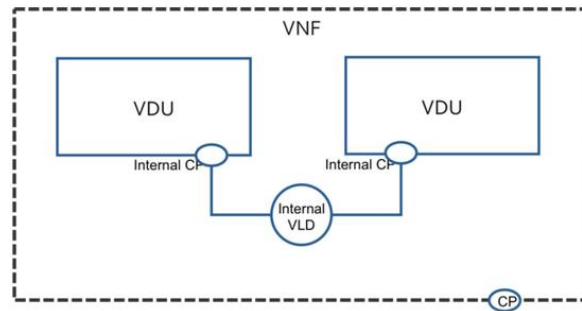


Figura 5.52. Estructura interna de Descriptor VNF (VNFD).

Para aplicar sobre el entorno de despliegue de NFV, se necesita un conjunto de ficheros que describan su configuración y además que permitan la instanciación. Este conjunto de ficheros se los conoce como *Paquetes VNFD*, los cuales encapsulan todos los artefactos necesarios para el funcionamiento de los descriptors, los mismos son distribuidos en formato *tar.gz*. A continuación se describen los ficheros que involucran dichos paquetes:

- `vnfd.yaml` (Descriptor NS en formato YAML)
- `README` (información respecto a NS)
- `Checksum.txt` (Contiene checksum de cada fichero en el paquete)
- `Scripts\` (Directorio que contiene scripts de NS con eventos de ciclo de vida)
- `images\` (Directorio que contiene todas las imágenes para la VNF)
- `icons\` (Directorio que contiene íconos en formato `.png`)
- `charms\` (Directorio que contiene la configuración de charms y plugins)

Descriptor NS [NSD]: Del mismo modo un fichero NSD puede ser desarrollado y provisto por un vendedor para cumplir con sus servicios requeridos, y también describe un conjunto de propiedades principales:

- Nombre único del escenario de red
- Define su topología y sus conexiones (VNFs y Redes)

⁴⁰ <http://www.yaml.org/start.html>

- Cada nodo tiene sus propias características como tipo de VNF o de red, modelos de VNF, modelo de redes para los casos de datacenters externos o públicos y en caso de una red interna privada el modelo debe ser establecido como un `bridge_net` (plano de control) o `dataplane_net` (plano de datos).
- Cada conexión tiene un nombre y un conjunto de nodos interconectados entre sí.

En la figura a continuación se muestra la estructura de conexión de un template de escenarios. En donde se detallan sus interfaces y conexiones:

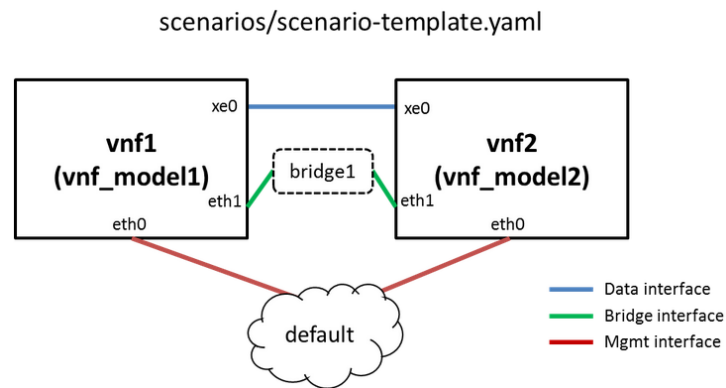


Figura 5.53. Estructura interna de un descriptor de servicio de red.

En consecuencia los NSDs definen las conexiones de las VNFs mediante VLs. De igual forma, estos enlaces se los realiza por medio de los puntos de conexión CPs. Por lo tanto el fichero NSD, también en formato YAML, contiene una lista de las VNFs, de las VLs, y de los puntos de conexión NS. En la figura a continuación se muestra la arquitectura de NSD optada por la ETSI.

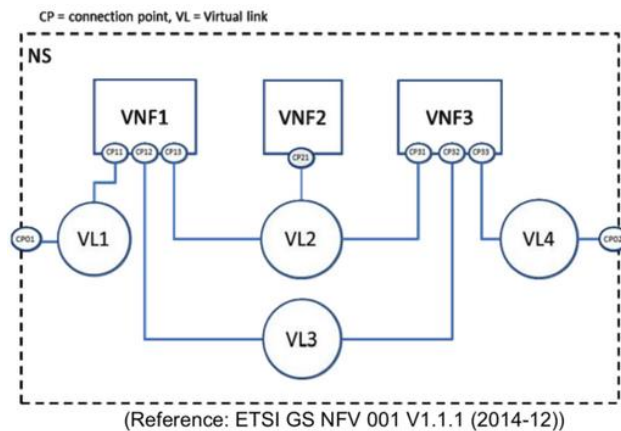


Figura 5.54. Estructura interna de Descriptor NS (NSD).

Al igual que los paquetes VNFD, también se necesita disponer de paquetes NSD, los mismos que contienen ficheros similares dentro de su paquete. A continuación se detallan:

- nsd.yaml (Descriptor NS en formato YAML)
- README (información respecto a NS)
- Checksum.txt (Contiene checksum de cada fichero en el paquete)
- Scripts\ (Directorio que contiene scripts de NS con eventos de ciclo de vida)
- icons\ (Directorio que contiene los íconos y logs para el servicio de red)
- ns-config\ (Directorio que contiene ficheros con valores por defecto de NS)
- vnf-config\ (Directorio que contiene ficheros con valores por defecto de VNF)

Del conjunto total de ficheros que conforman el paquete, los más importantes que deben existir son los archivos con formato YAML y el archivo **checksum.txt**, sin ellos OSM no puede cargar y desplegar los escenarios y VNFs.

Escenario de Red [NS]: A continuación se muestra la arquitectura interna de los escenarios simples y complejos detallados por sus templates. OpenMANO pone a disposición estos ficheros para su uso y adaptación a un nuevo si se desea crear:

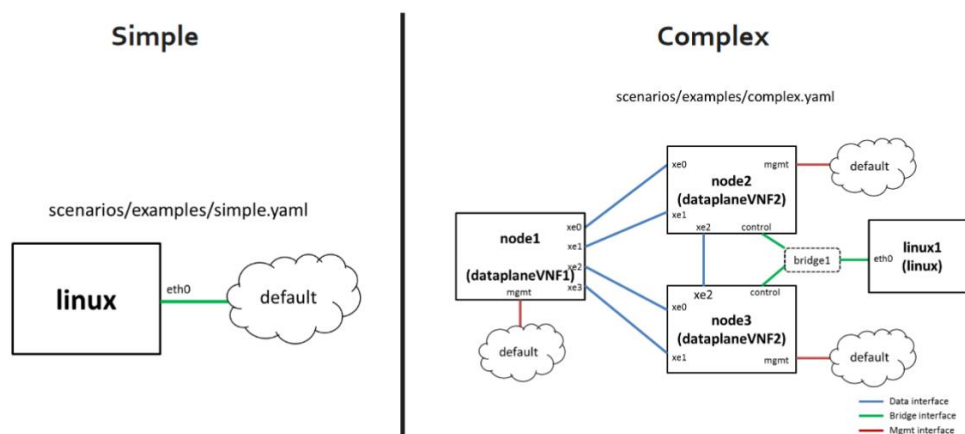


Figura 5.55. Estructura interna de escenarios de red. Simple y complejo.

5.2.4 Proceso de Interconexión de Componentes

Una vez detallado la estructura interna de conexiones de las diferentes capas de abstracción para el despliegue de VNFs, en esta sección se explica el proceso de acoplamiento entre las dos plataformas OSM y Openstack. Cada una de las mismas

posee sus componentes que las gestionan internamente. OSM se enfoca solo en el plano de orquestación y gestión, por tal motivo, la arquitectura y funcionamiento del cloud Openstack debe encontrarse debidamente configurado y funcional.

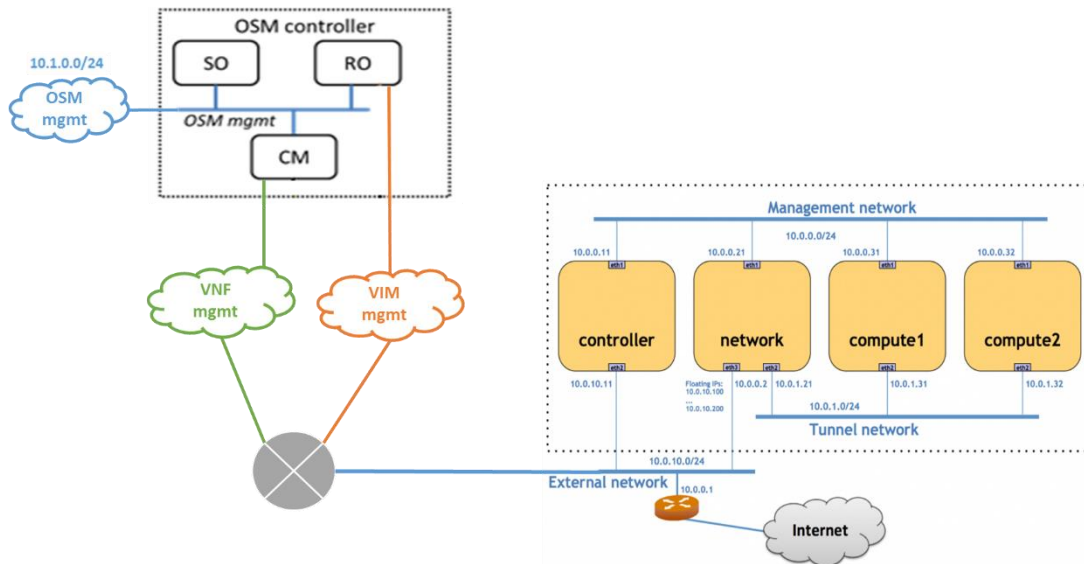


Figura 5.56. Acoplamiento entre plataformas OSM y Openstack.

Para realizar el proceso de acoplamiento entre las dos plataformas se aconseja seguir los siguientes pasos con el fin de tener un mejor control y en el caso de tener problemas, identificarlos por separado:

- 1) Identificar y establecer redes independientes de conectividad entre componentes de OSM y el End Point de Openstack. Una vez realizado, verificar su conectividad.
- 2) Acoplar independientemente los componentes de OSM (RO y CM) con Openstack mediante procesos de configuración e intercambio de información entre ellos. Una vez realizado este proceso verificar en cada uno de los componentes de OSM si se ha completado dicho proceso. RO y CM poseen sus interfaces gráficas en donde se puede verificar visualmente este proceso.
- 3) Y finalmente establecer el control de estos componentes por medio del orquestador general SO, el cual es la interfaz gráfica principal por donde se gestionará todo el proceso de comunicación dentro de virtualización de funciones de red.

1. Identificar Redes:

Con el fin de obtener un mejor control sobre el VIM de Openstack, se establecen dos redes independientes que se interconectan con el cloud gestionado por openstack. La una **“VIM mgmt”** se encarga de interconectar el componente RO con Openstack. Por medio de la misma se despliegan las configuraciones e instancias de VNF.

Otra red que se establece es **“VNF mgmt”**, encargada de enlazar los servicios que se despliegan sobre el cloud de openstack. Se ocupa principalmente en unir todos los servicios desplegados en los nodos de cómputo de la nube y mediante ello agilizar las comunicaciones entre las distintas instancias desplegadas. Dentro de este componente se pueden cargar varios servicios sobre los nodos de cómputo como por ejemplo bases de datos, contenedores, controladores Nova, etc., hasta servicios de tipo VoIP, virtualizando la parte de IMS de las redes móviles.

Y finalmente las red **“OSM mgmt”**, que provee la comunicación entre los distintos componentes de OSM, además que permitirá el intercambio de información entre los mismos para luego ser desplegada sobre la interfaz GUI de SO. El resto de redes dentro de Openstack ya se encuentran definidas por el escenario y no entran dentro del enfoque de OSM.

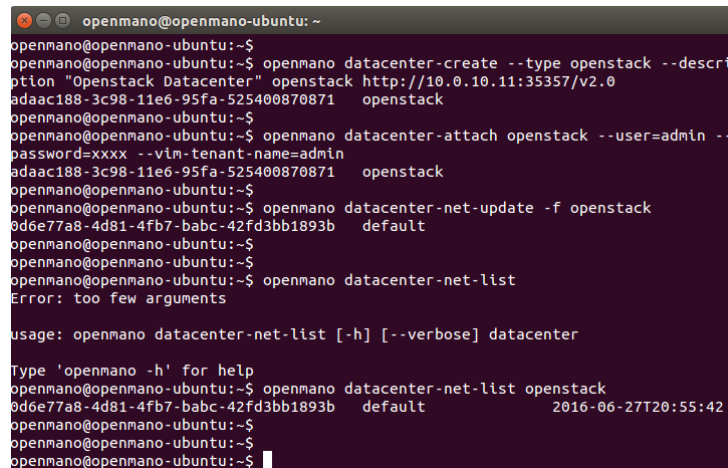
2. Acoplamiento Independiente:

Acoplamiento RO - Openstack: Como se explicó en los puntos anteriores cada uno de los componentes de OSM realizan funciones específicas sobre VIM. El primero de ellos, RO es el encargado de desplegar y configurar las redes e instancias de las VNFs sobre el VIM, entonces se lo debe acoplar al gestor del cloud, que en este caso es el componente **“Controller”**, para que el internamente pueda ejecutar las órdenes dictadas por RO.

Primero se debe establecer su red de gestión por donde se va a comunicar y gestionar el despliegue de VNFs. La misma que se le denominó como **“VIM mgmt”**, por medio de ella dentro de RO se crea el nuevo **datacenter** que va a gestionar y se lo adhiere a su inquilino aislado (tenant), ya que el tenant dentro del cloud es el encargado de proveer la información de las conexiones internas, hosts, y redes internas. Este proceso se lo hace por medio de los siguientes

comandos ejecutados en RO, y en la figura a continuación se muestra el proceso exitoso de acoplamiento:

- `openmano datacenter-create <name> <openstack-url> --type=openstack`
- `openmano datacenter-attach <name> --user=<user> --vim-tenant-name=<openstack tenant> --password=<password>`
- `openmano datacenter-netmap-upload -f -datacenter <name>`



```
openmano@openmano-ubuntu: ~
openmano@openmano-ubuntu:~$ openmano datacenter-create --type openstack --description "Openstack Datacenter" openstack http://10.0.10.11:35357/v2.0 adaac188-3c98-11e6-95fa-525400870871 openstack
openmano@openmano-ubuntu:~$ openmano datacenter-attach openstack --user=admin --password=xxxx --vim-tenant-name=admin adaac188-3c98-11e6-95fa-525400870871 openstack
openmano@openmano-ubuntu:~$ openmano datacenter-net-update -f openstack 0d6e77a8-4d81-4fb7-babc-42fd3bb1893b default
openmano@openmano-ubuntu:~$ openmano datacenter-net-list
Error: too few arguments

usage: openmano datacenter-net-list [-h] [--verbose] datacenter

Type 'openmano -h' for help
openmano@openmano-ubuntu:~$ openmano datacenter-net-list openstack
0d6e77a8-4d81-4fb7-babc-42fd3bb1893b default 2016-06-27T20:55:42
openmano@openmano-ubuntu:~$
openmano@openmano-ubuntu:~$
```

Figura 5.57. Acoplamiento entre RO y Openstack.

Acoplamiento CM - Openstack: Para poder establecer el acoplamiento entre los dos componentes, JUJU trabaja con la arquitectura de cliente servidor, por tanto en el componente de CM se carga el servidor JUJU que será el encargado de gestionar los servicios que se despliegan en openstack y lo visualizará por medio de la interfaz GUI que posee. Aquí se muestran todos los nodos que conforman el cloud, así como los servicios que poseen cada uno de ellos.

Pero para alcanzar este punto se necesita cargar el paquete cliente de JUJU. Existen dos opciones para hacerlo, la primera se lo puede hacer dentro del controlador de Openstack, en este caso sobre el nodo "Controller". O a su vez el cliente Juju puede estar cargado sobre la misma máquina virtual donde se encuentra el servidor Juju. Pero de forma general el cliente Juju se encargará de distribuir todos los servicios de forma horizontal entre los distintos nodos de computación que se encuentran a su mando. Por medio de él también se pueden agregar nodos, componentes y servicios según sean las necesidades.

Todo este proceso se lo puede realizar mediante la ejecución de los siguientes comandos básicos:

- juju bootstrap -v
- juju-deployer charm tools
- juju add-machine ssh:<name>@<host>
- juju add-unit <name>
- juju deploy <host>
- juju add-relation <service name >-<service name>
- juju add-relation <host name>-<service name>

De forma general JUJU y OpenMANO se encargan de la gestión y despliegue de VNFs sobre el datacenter en el que trabajen, de tal modo que se podría analizar dicha gestión como un paquete VNF el cual posee distintas secciones que cumplen independientemente los componentes CM y RO, estas secciones son el traspaso y ejecución de los ficheros explicados en la sección anterior sobre los descriptores. En la figura a continuación se muestra un resumen sobre el modelo de datos que manejan estos componentes en el proceso de acoplamiento con el VIM. [34]

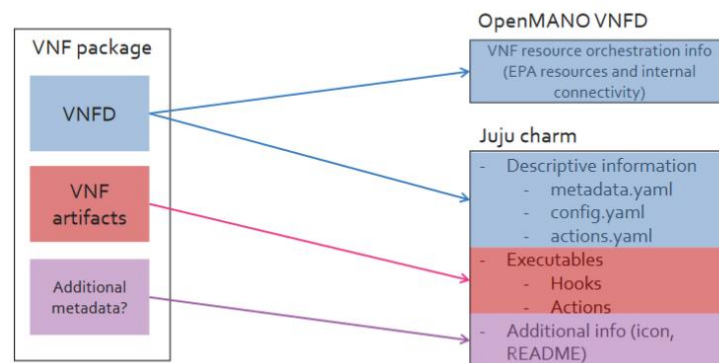


Figura 5.58. Modelo de datos del paquete VNF. [34]

3. **Acoplamiento SO con CM y RO:** Dentro de la arquitectura de OSM, SO es el componente principal de orquestación, de tal modo, es el encargado de coordinar el ciclo de vida en la creación y despliegue de escenarios virtualizados, así como la gestión ordenada de la compleja arquitectura.

SO en su arquitectura presenta varios componentes que permiten la interacción entre los distintos componentes de gestión, como por ejemplo en la sección de MANO provee plug-in de acoplamiento con OpenMANO, llamado NSO

(Network Service Orchestrator). De igual forma posee otro plug-in para comunicarse con JUJU, el mismo que se encuentra detallado en la sección **Configuration Manager** de MANO.

En el núcleo de arquitectura de SO incorpora varias librerías que se encargan de proveer los métodos y configuraciones para generar procesos y servicios como métricas de NFV, gestión de contabilidad, Catálogos, modelos de traducción, etc. Dentro de estas librerías una de las que se destaca es **CAL**, la cual proporciona métodos para la orquestación de Openstack y plug-ins de AWS (Amazon Web Service)⁴¹. En la gráfica 5.59 se muestra la arquitectura detallada del componente SO utilizado en OSM [35]:

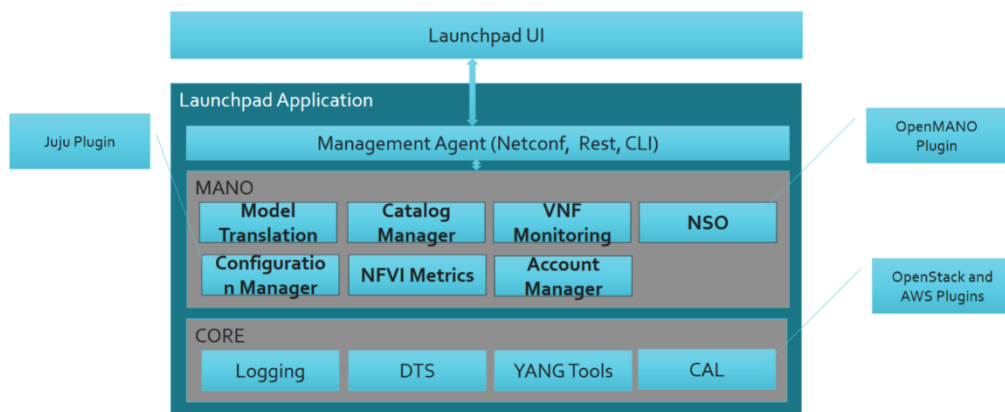


Figura 5.59. Acoplamiento de SO con componentes CM y RO[35].

A. Acoplamiento SO - RO: El proceso de acoplamiento entre estos dos componentes se lo hace desde el dashboard de OSM, es decir desde la interfaz de usuario de Rift. Para completar la operación nos dirigimos a la sección de “Accounts”, y simplemente ponemos agregar “Cloud Accounts”, aquí nos pide que especifiquemos algunos parámetros, que dependiendo del tipo de cloud que se configure, solicitará los parámetros de configuración. En este caso ocupamos un cloud Openmano, el cual gestiona a su VIM Openstack. En la figura 5.60 se muestra el proceso exitoso de acoplamiento entre SO y RO y los parámetros de configuración como:

- Tipo de Cloud: Openstack, OpenMano, Amazon, etc.
- Dirección IP del host controlador del cloud, es decir dirección IP de la máquina donde se encuentra OpenMano.

⁴¹ <https://aws.amazon.com/es/ec2/vcenter-portal/>

- Puerto de Conexión.
- Y el identificador del tenant configurado en OpenMano

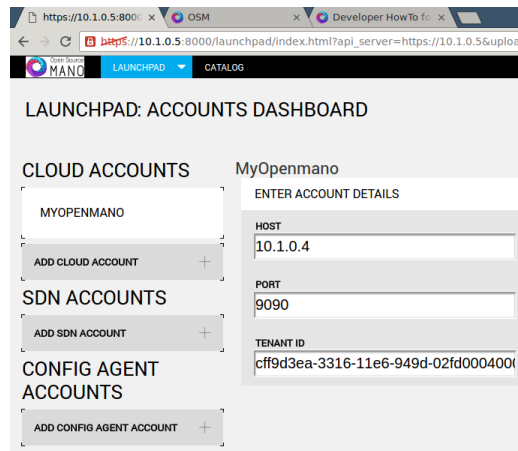


Figura 5.60. Acoplamiento entre SO y RO en dashboard OSM.

- B. Acoplamiento SO - CM:** Al igual que el acoplamiento entre SO y RO, para CM es parecido, pero un tanto más específico. Accedemos igual a la sección “Accounts”, e ingresamos a “Configuration Agent Accounts”, aquí ingresamos los parámetros necesarios de Juju. En la figura 5.61 se indica el proceso exitoso de acoplamiento exitoso entre SO y CM, además de los parámetros necesarios para su acoplamiento como dirección IP, su puerto de comunicación (opcional) y sus credenciales de acceso.

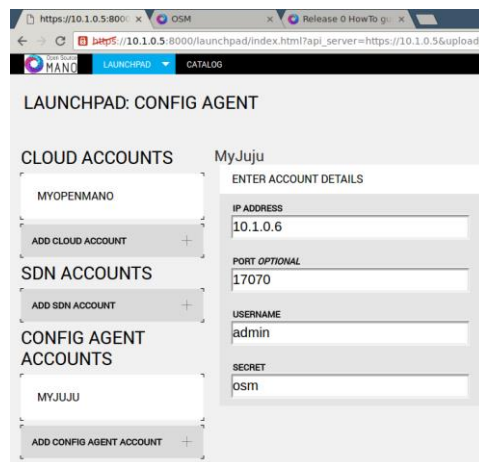


Figura 5.61. Acoplamiento SO y CM en dashboard OSM.

5.2.5 Flujo de Trabajo de OSM

En la figura a continuación se muestra un diagrama de las conexiones de comunicación entre los distintos componentes del escenario, se lo estructura resaltando

las distintas capas que involucran, más sus enlaces a donde apuntan con sus servicios específicos, para más adelante explorar el proceso mediante el cual se puede generar y desplegar un VNF sobre el entorno VIM.

En el gráfico se observa claramente que el encargado de interactuar directamente con VIM es OpenMANO, de tal forma se puede entender que será el encargado de generar las órdenes de creación y despliegue de máquinas virtuales sobre los nodos de computación del cloud Openstack, así como configurar las instancias y escenarios VNF.

Dentro del componente Juju se cargan varios servidores proxy, donde gestionará la implementación de servicios sobre las VNF del entorno desplegado. Nótese que por cada VNF creada se tiene un servicio asignado, interconectado mediante un Proxy Charm. En la figura 5.62 se muestra como interactúa de forma directa Juju con las VNFs cargadas en los nodos de cómputo del cloud. Pero en la práctica, su conexión se lo hace en primer lugar hacia el controlador del entorno, debido que él es el encargado de distribuir las órdenes sobre todos los nodos a su cargo. Además el gráfico está diseñado para una orquestación sobre dos tipos distintos de VIM, como Openstack y OpenVIM propio de OpenMANO. [36]

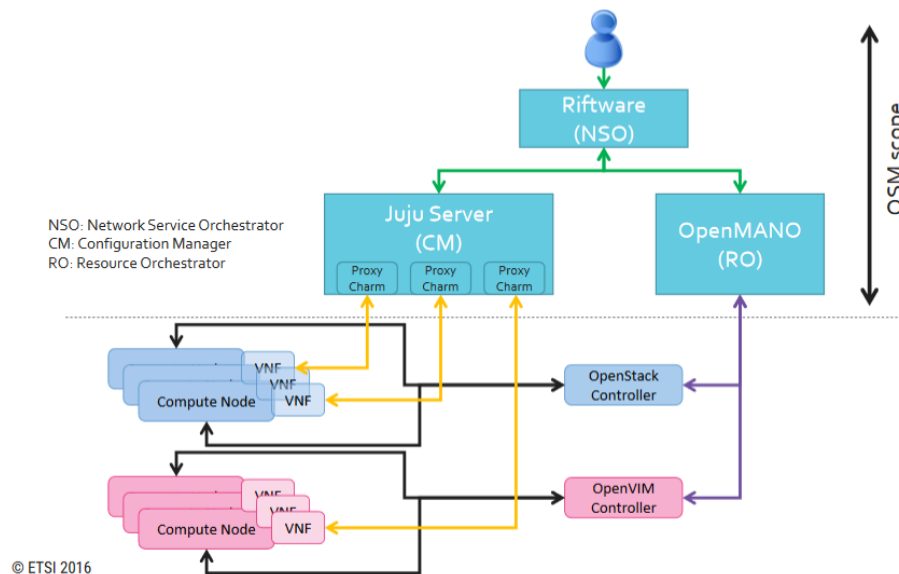


Figura 5.62. Arquitectura actual del modelo OSM.

Cuando se inicia GUI de OSM se procede a cargar los ficheros NSD y VNFD para, después de ser instanciados, se los despliegue dentro del entorno.

Una vez puesto a funcionamiento el escenario de OSM, el usuario deseará agregar ficheros VNFD y NSD, ya sea para funcionamiento real o simplemente como ensayo de despliegue. En la interfaz GUI se observan los ficheros cargados, los cuales pueden ser inspeccionados, para analizar sus configuraciones y parámetros establecidos. Antes de continuar con el estudio del ciclo de vida de instanciación, se muestra como se cargan los distintos ficheros de los escenarios y VNF, en la figura 5.63.

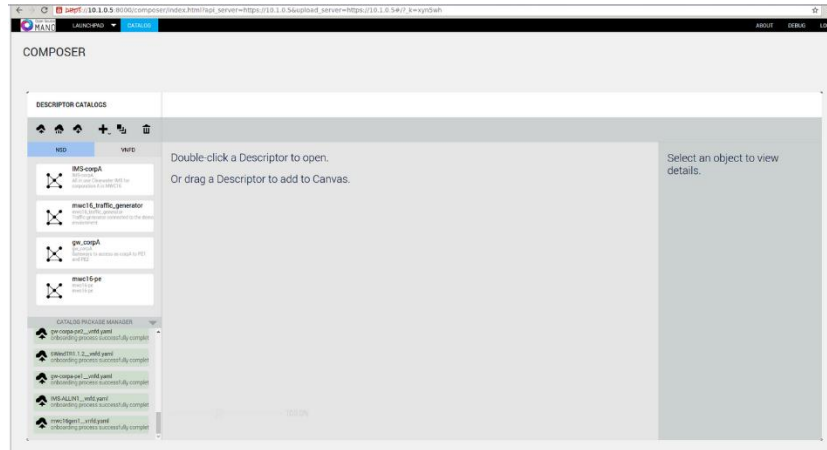


Figura 5.63. Carga de ficheros NSD y VNFD en OSM.

Entonces se debe analizar el ciclo de vida de la gestión del componente SO, al cual se lo divide en cuatro bloques para diferenciar los procesos que se generan a cada instancia. Los pasos que se producen se los detalla a continuación:

1. El primer bloque es **UI (Interfaz de Usuario)**, es por donde se emite la orden de gestión al entorno a orquestar. Como primer paso genera la instancia del servicio de red (NS), el cual llega hasta el bloque de gestión de ciclo de vida.
2. Este bloque valida la petición recibida y también revisa si tiene cargado todas las VNFs necesarias para establecer la petición.
3. Una vez verificadas las VNFs, se procede a iniciar la requerida, y a su vez pasa ésta información hacia el plugin de Openmano.
4. Receptada la instancia VNF, se realiza el proceso de traducción del descriptor VNF y envía un mensaje de asentamiento (ACK) hacia el bloque de gestión del ciclo de vida.
5. Una vez que se realizó la traducción el plugin crea la VNF solicitada y lo envía hacia el bloque de Openmano. Todo este proceso se lo realiza para cada petición de creación y despliegue de una VNF.

6. Finalizado el proceso de creación de VNF, el bloque de gestión de ciclo de vida, procede a crear una instancia de servicio de red (NS), que de igual forma lo envía al plugin de Openmano.
7. Éste recibe la instancia, crea el escenario (NS) para enviar al bloque de Openmano y envía un asentamiento (ACK) hacia el bloque anterior.
8. Como penúltimo paso, el mismo bloque de gestión de ciclo de vida despliega el servicio de red NS y envía la información hacia al plugin Openmano,
9. El cual finalmente crea la instancia y despliega hacia el bloque de Openmano, para que sea aplicado dentro de la VIM que opera.

En la figura 5.64 se muestra el proceso de ciclo de vida para generar un servicio de red.

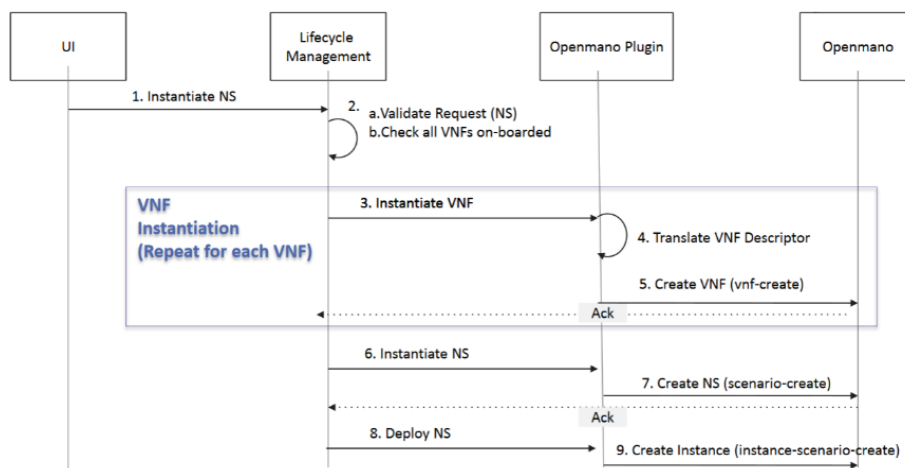


Figura 5.64. Gestión del ciclo de vida de red de servicio (NS). [36]

De forma visual, en el entorno gráfico de OSM se cargan los ficheros VNFD y NSD, y nos dirigimos hacia la sección de instanciación, donde editamos parámetros como dar el nombre a nuestro despliegue, seleccionar al orquestador de recursos (por defecto Openmano), y escogemos el datacenter en donde se ejecutarán los escenarios y sus respectivas VNFs. Como se puede observar en la figura 5.65, una vez acoplado el datacenter con OSM (explicado en puntos anteriores), reconoce que trabajamos con cloud Openstack.

En la parte inferior nos muestran los distintos NSDs que hayamos cargado para su instanciación y despliegue dentro de Openstack. Adicionalmente se pueden verificar

los parámetros de cada uno de ellos para observar sus definiciones y parámetros de utilización.

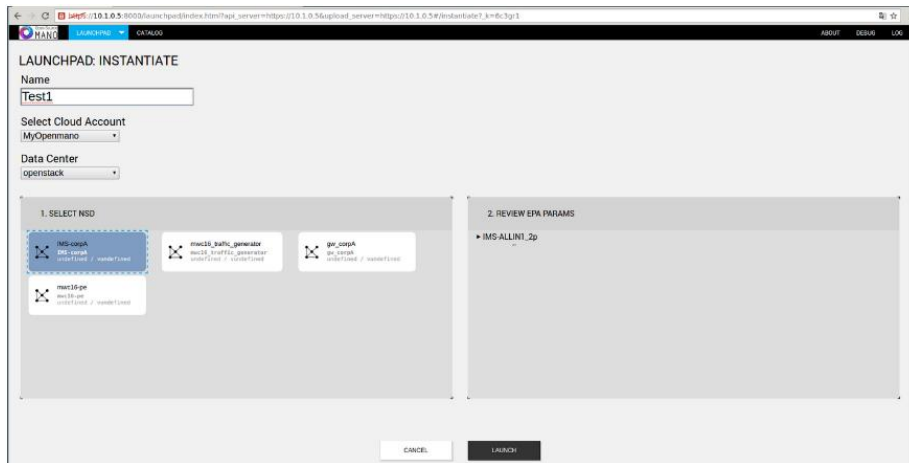


Figura 5.65. Despliegue NSD en GUI de OSM.

De igual modo, una vez creada la instancia NS, se la puede desplegar. Para ello existe también un conjunto de pasos que se realiza entre los distintos componentes hasta llegar a la VNF desplegada dentro del cloud.

1. El usuario despliega la NS, enviando la orden hacia el componente Riftware, el mismo que traduce el script VNFD.
2. Luego de traducir, crea la VNF, la NS se inicializa, todo esto es enviado hacia el componente OpenMANO.
3. Openmano recibe todos esos módulos de información e instanciación de redes y máquinas virtuales (VMs), las mismas que las despliega sobre el VIM que controla. Además si se completa con éxito este proceso, devuelve un asentamiento de proceso completado hacia el componente Riftware.
4. Riftware, una vez que recibe el ACK de instanciación, despliega un Proxy Charm sobre el componente Juju y además aplica la configuración inicial.
5. Finalmente Juju Server aplica ésta configuración directo sobre la VNF desplegada en la NFV del VIM operado por Openmano.

En la figura 5.66, se muestra en detalle el proceso de instanciación de un servicio de red por medio de los tres componentes de OSM, éste proceso se aplica para cada NS desplegada, e internamente existen procesos que se realizan por cada VNF iniciada.

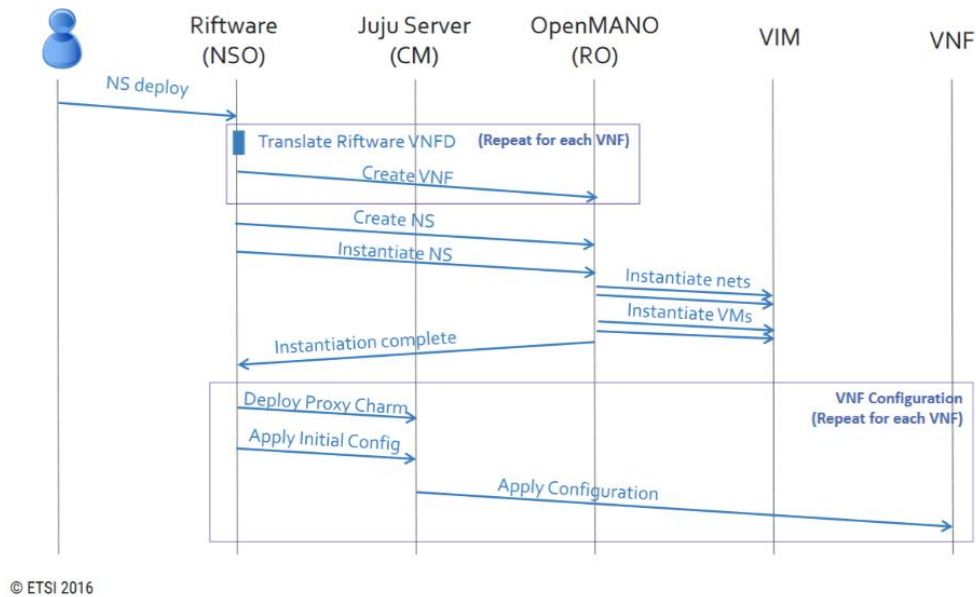


Figura 5.66. Proceso de Instanciación del despliegue de NS.

5.3 Pasos Para Creación de Descriptores VNF y NS

Para los dos casos tanto para VNFD y NSD, se los define en un template de formato YAML. A continuación se detallan los pasos para crear ficheros VNFD y NSD:

- **VNFD:**
 1. Detallar y definir información básica de VNF
 - Nombre de VNF
 - Descripción de VNF
 - Versión de VNF (opcional)
 - Versión de Descriptor
 - Tipo de VNF (Clasificador de Tráfico, Firewall, etc.)
 2. Crear la composición de la VNF. Aquí el proveedor puede seleccionar su "flavor" de preferencia, y realizar una composición para cada "flavor".
 3. Proveedor puede agregar VDUs a cada "flavor", y ahí define los recursos virtualizados de computación que le asigna a la máquina virtual. Con estos parámetros se pueden gestionar los eventos del ciclo de vida.
 4. Finalmente se establecen los enlaces virtuales que se encuentran disponibles en la composición para el enlace individual de máquinas virtuales o para brindarles acceso externo.

- **NSD:**
 1. En primer lugar se seleccionan las VNFs. El proveedor de servicios selecciona las VNFs que le gustaría aplicar a su servicio. El proveedor también será capaz de negociar con el proveedor de funciones para realizar peticiones y ofertas según sus necesidades.
 2. A continuación se agrega información básica del servicio de red (NS), como:
 - Nombre del servicio
 - Descripción del servicio
 - Versión del NSD
 - Versión del Descriptor
 3. Finalmente se describen los parámetros que garantizan el funcionamiento del servicio para cada “*flavor*” disponible. El proveedor puede mirar y gestionar los servicios creados. Ambos, tanto VNFD como NSD, pueden editar sus descriptores YAML generados.

6 Conclusiones y Líneas Futuras de Investigación

Después de haber realizado el análisis y descripción del proyecto en las secciones anteriores, en el presente capítulo se explican las conclusiones obtenidas a lo largo del desarrollo del estudio y de la puesta en ejecución sobre la herramienta de orquestación OSM. Además se plantean posibles líneas de investigación futuras, donde se pueda profundizar en ciertos puntos que quedan pendientes después de la realización del trabajo.

6.1 Conclusiones

La tecnología NFV aparece como una propuesta que principalmente busca el dinamismo y flexibilidad dentro de las redes de comunicaciones, permitiendo obtener grandes beneficios, como ahorros en gastos de inversión de infraestructura, así como en mantenimiento, operación y capacitación de personal especializado. La rápida convergencia de las redes ha hecho que se dispare la cantidad de creación y adquisición de equipos dedicados de propietarios para la implementación dentro de los entornos de red de alta demanda de servicios, como redes móviles. Por tal motivo, dentro de los distintos casos de uso de NFV, la ETSI se ha enfocado en establecer las distintas normas de uso y construcción de dicha tecnología para ser implementada dentro de la arquitectura de datos de las redes móviles.

El desacoplamiento de VNF de un equipo de red dedicado, así como ha generado grandes beneficios, también ha producido retos en cara a las nuevas exigencias y demandas que poseen las distintas operadoras. Dentro de los principales retos se encuentra garantizar el servicio de extremo a extremo con NFV, por tal motivo la instanciación y localización de las VNFs son factores que influyen en gran medida al momento de garantizar la disponibilidad y fluidez del tráfico. Con estos criterios, muchos desarrolladores y vendedores iniciaron proyectos de gestión y orquestación de redes NFV, debido a que se planteó el despliegue sobre infraestructuras de cómputo de alto rendimiento como servidores en centros de datos, los mismos que pueden permanecer localizados en cualquier sitio geográfico.

Distintas investigaciones se han realizado alrededor de los métodos y técnicas de aplicar NFV dentro EPC. Proponiendo virtualizaciones parciales que permiten la coexistencia entre redes tradicionales y redes NFV, de igual forma se han planteado técnicas para aplicar VNF a ciertos componentes específicos del núcleo de red, impulsando la agilidad en el plano de control y aislándola completamente del plano de datos, con el fin de mejorar el rendimiento de red, como es el caso de vMME. De igual

forma técnicas como la de agrupamiento por segmentos definidos por componentes específicos del núcleo de red, son casos que se encuentran en estudio y desarrollo para sus futuras implementaciones, ya que se ha demostrado, el éxito de NFV radica en su capacidad de gestión y dinamismo al momento de instanciación y despliegue.

La estructura por bloques que implementa OSM, hace que sus prestaciones sean mucho más amplias, escalables y de gran desempeño, debido que involucra códigos semillas de distintos proyectos importantes creados para manejar la gestión de servicios y de virtualización. Su modularidad permite tener un mejor entendimiento y control sobre los distintos procesos que se ejecutan al momento de instanciar y desplegar un escenario de red con funciones de red virtualizadas. Los ciclos de vida de los procesos pueden ser controlados fácilmente, agilizando la detección de errores o malos funcionamientos dentro de las instancias, todo esto gracias que al igual que los otros proyectos, presenta una interfaz gráfica amigable y bastante sencilla que permite configurar y establecer los parámetros de funcionamiento necesarios para crear un entorno de red VNF.

OSM fue diseñado en base a las normas dictadas por la ETSI dentro del entorno arquitectónico que se deben desplegar VNFs, pero como se ha analizado, existen varias capas que tienen que ser gestionadas, y es aquí donde radica la innovación de OSM, ya que al momento de involucrar distintos componentes, delega el trabajo específico a cada uno de ellos, optimizando procesos, agilizando la gestión y mejorando el desempeño de la orquestación total de las instancias, con el fin de garantizar alto rendimiento en servicios de extremo a extremo en entornos incluso de multi-VIM.

Finalmente en base a la experiencia de utilización e implementación de la herramienta OSM, se perfila como una de las mejores propuestas sobre la orquestación de NFVs, sentando una base firme para la gestión de redes futuras LTE y 5G, y a la vez presentando versatilidad al involucrar SDN dentro de su entorno, de este modo complementa el ecosistema de soluciones sobre el manejo de tecnologías de virtualización y de redes definidas por software. El futuro está encaminado en dinamizar el flujo de información y automatizar el control de los recursos de red, por tal motivo, los centros de datos se convertirán en piezas claves para soportar y/o servir como pasarelas de las operadoras de red. Aunque en la actualidad todavía se presentan desventajas en dichas tecnologías, las cuales limitan su implementación completa, como por ejemplo la falta de inteligencia en el despliegue de VNFs, y recuperación frente a fallos, que son temas de suma importancia al utilizarlos como plataformas distribuidas dentro de datacenters.

6.2 Líneas Futuras de Investigación

Como resultado de la investigación del proyecto, OSM abre distintas líneas de investigación en donde, vendedores, investigadores, desarrolladores, y comunidades de informáticos a fines a las telecomunicaciones, podrán incurrir en generar nuevas aportaciones a los retos planteados por las actuales tecnologías como NFV y SDN.

Para el despliegue de NFV, se necesita de varias capas de soporte, que garanticen su normal funcionamiento, por ende se desprenden objetivos puntuales para trabajos futuros. Por ejemplo en la sección de VIM, existen varios proyectos que se encargan de su gestión. Aquí Openstack es la herramienta más versátil y utilizada por vendedores, pero el crecimiento de datacenters ha generado el aumento de gestores, entonces una línea de investigación clara es desarrollar APIs estandarizadas, que sean capaces de gestionar cualquier tipo de cloud desplegado, con esto aumentaría el rendimiento y facilitaría la detección de fallos, asignando tareas específicas a cada cloud.

OSM involucra compatibilidad con dos tipos de cloud, gracias a que OpenMano, en su código base posee plugins y APIs que permiten la interacción con Openstack, y OpenVim. Pero OpenMano no es capaz de recuperarse frente a fallos y generar procesos de autoarranque sobre otro cloud. Éste trabajo sería muy interesante ya que al orquestador de recursos se lo estaría integrando inteligencia a su gestión, lo cual aumentaría el dinamismo del despliegue y funcionamiento de los entornos VNF.

Al nivel de Gestión y despliegue de VNFs, también se podrían efectuar trabajos futuros, por ejemplo diseño de templates VNFD y NSD que permitan generar soluciones de balanceo de tráfico, de seguridad de redes, de pasarela del plano de datos, y principalmente en control y autenticación de registro a redes móviles, como es el caso de MME y HSS. El análisis y desarrollo de dichos “*templates*” requiere dedicarles líneas de investigación ya que son la base fundamental dentro de los entornos virtualizados y de SDN. En la actualidad se están realizando pruebas con escenarios dedicados para evaluar el rendimiento del servicio de extremo a extremo mediante un entorno completamente NFV.

Finalmente en el campo académico, el desarrollo de un escenario base NSD donde se pueda implementar el despliegue de distintas VNFs, que puedan ser gestionadas con OSM, sería un aporte investigativo de gran importancia dentro de entornos NFV.

Bibliografía

- [1] M. Darkaoui, "DISEÑO Y MONITORIZACIÓN DE UNA ESTACIÓN MÓVIL MULTISISTEMA EN ENTORNO RURAL," p. 15, 2013.
- [2] E. Dahlman, S. Parkvall, J. Skold, and P. Beming, *3G Evolution: HSPA and LTE for Mobile Broadband*, Elsevier L. San Diego, California: Elsevier Ltd., 2010.
- [3] G. He, "Overview of UMTS," 2001.
- [4] H. Holma and A. Toskala, *LTE for UMTS Evolution to LTE-Advanced*. 2011.
- [5] F. Bernardo, F. Casadevall, and O. Sallent, *LTE Nuevas tendencias en comunicaciones móviles*. 2010.
- [6] G. P. Project, T. Specification, G. Radio, A. Network, E. Universal, T. Radio, and A. Network, "3gpp ts 36.300," vol. 0, no. Release 12, 2016.
- [7] T. Specification, "Etsi ts 123 401," vol. 0, pp. 0-279, 2011.
- [8] T. Specification, "based Mobility and Tunnelling protocols ;," vol. 0, 2012.
- [9] M. Sauter, *From GSM to LTE-Advanced*, 2nd Editio. United Kingdom: Wiley, 2014.
- [10] A. Singh and A. Rights, "kernel thread . com An Introduction to Virtualization," pp. 1-27, 2003.
- [11] E. Engineering, "Future Mobile Communications : LTE Optimization and Mobile Network Virtualization," no. May, 2012.
- [12] A. Hakiri and P. Berthou, "Leveraging SDN for The 5G Networks: Trends, Prospects and Challenges," *arXiv Prepr. arXiv1506.02876*, pp. 1-23, 2015.
- [13] A. Introduction, C. Action, M. Chiosi, D. Clarke, P. Willis, A. Reid, J. Feger, M. Bugenhagen, W. Khan, M. Fargano, C. Cui, H. Deng, D. Telekom, and U. Michel, "Network Functions Virtualisation," *Citeseer*, no. 1, pp. 1-16, 2012.
- [14] N. O. Perspectives and I. Progress, "Network Functions Virtualisation (NFV)," no. 1, pp. 1-16, 2013.
- [15] "4G Americas Bringing Network Function Virtualization to LTE November 2014 0," no. November, pp. 0-56, 2014.
- [16] M. Chiosi, S. Wright Bell Canada, J. Erfanian, B. B. Smith, B. Briscoe, A. Reid, P. Willis CableLabs, D. Clarke, C. Donley CenturyLink, M. Bugenhagen, J. Feger, J. Benitez, N. Fischbach Deutsche Telekom, K. Martiny, U. Michel DOCOMO, T. Nakamura, J. Triay Marques KDDI, K. Ogaki, T. Matsuzaki KPN, S. Zhang, A. K. de Boer, K. Ok, E. Kyoung PAIK NTT, K. Shimano, T. Shimizu Ooredoo, M. Stura Orange, B. Chatras, C. Koliass Portugal Telecom, J. Carapinha, A. S. Gamelas Telecom, D. Lee, J. Han Park Softbank, R. Wakikawa, K. Nishi, S. Matsushima Sprint, L. Laporte, F. Feisullin Swisscom, M. Brunner Telecom

- Italia, E. Demaria, A. Pinnola Telenor, P. Waldemar, G. Millstein Telefonica, D. López, F. Javier Ramón Salguero Telstra, D. Kirkham Turk Telekom Argela, M. Ergen, M. Ahmet Karaman, A. Ulas, E. Lokman Verizon, N. Khan, R. Morera Vodafone, S. Sabater, A. Neal Windstream, and A. Nichols, "Network Functions Virtualisation (NFV) CONTRIBUTING ORGANISATIONS & AUTHORS PUBLICATION DATE," no. 1, pp. 1-20, 2015.
- [17] Dialogic, "Dialogic Cloud-Ready Solutions : A Smart Approach to NFV," *Tbd*, 2015.
- [18] E. G. N. 001 V1.1.1, "Network Functions Virtualisation (NFV); Use Cases," *IEEE Netw.*, vol. 1, no. 5, pp. 1-50, 2013.
- [19] B. J. Marc Vorwerk, "Communications as a cloud service: a new take on telecoms Communications as a cloud service : a new take on telecoms," 2014.
- [20] P. Ram and S. Esfandiari, "NFV : La Virtualización de las 'telcos,'" pp. 20-25.
- [21] J. Prados-Garzon, J. J. Ramos-Munoz, P. Ameigeiras, P. Andres-Maldonado, and J. M. L. Soler, "Latency Evaluation of a Virtualized MME," 2015.
- [22] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "NFV: State of the Art, Challenges, and Implementation in Next Generation Mobile Networks (vEPC)," *Network*, vol. Nov, no. December, pp. 18-26, 2014.
- [23] A. N. F. V Isg, P. Proposal, A. N. F. V Poc, and P. Participants, "NFV ISG PoC Proposal - virtual EPC with SDN Function in Mobile Backhaul Networks 1."
- [24] S. Matsusgima, "Stateless use-plane architecture for virtualized EPC (vEPC)," p. 20, 2013.
- [25] M. R. Sama, S. Ben, H. Said, K. Guillouard, and L. Suci, "Enabling Network Programmability in LTE / EPC Architecture Using OpenFlow," pp. 395-402, 2014.
- [26] Etsi and J. Quittek, "GS NFV-MAN 001 - V1.1.1 - Network Functions Virtualisation (NFV); Management and Orchestration," vol. 1, pp. 1-184, 2014.
- [27] R. Mijumbi, J. Serrat, J. Gorricho, S. Latre, M. Charalambides, and D. Lopez, "Management and orchestration challenges in network functions virtualization," *IEEE Commun. Mag.*, vol. 54, no. 1, pp. 98-105, 2016.
- [28] ETSI-OSM, "End-to-End Service Instantiation Using Open-Source Management and Orchestration Components - White Paper," pp. 1-12, 2016.
- [29] D. Fernandez, A. Cordero, J. Somavilla, J. Rodriguez, A. Corchero, L. Tarrafeta, and F. Gal??n, "Distributed virtual scenarios over multi-host Linux environments," *2011 5th Int. DMTF Acad. Alliance Work. Syst. Virtualization Manag. Stand. Cloud, SVM 2011*, p. 3, 2011.
- [30] STANDARDS INSTITUTIONS, "NFV-Architectural Framework," *Tbd*, vol. 1, pp. 1-21, 2014.

- [31] O. Foundation, "OpenStack Icehouse Installation Guide," *OpenStack Docs*, no. Icehouse, pp. 1-140, 2015.
- [32] R. Ware and S. V. Platform, "Rift.ware™," 2016.
- [33] O. Mano, "RIFT . ware Management &."
- [34] G. Garc, "OSM CODE SEEDS , RELEASE ZERO AND BEYOND," 2016.
- [35] B. T. Rift, "Network service orchestrator (nso)," 2016.
- [36] G. Garc and A. Hoban, "MWC ' 16 ARCHITECTURE OVERVIEW AND WORKFLOW," 2016.

Anexos

Una vez instalados los componentes de forma independiente, siguiendo las instrucciones de las páginas web de OSM y Openstack, se detallan los pasos a seguir para realizar el acoplamiento entre los distintos entornos.

```

                                UNIVERSIDAD POLITÉCNICA DE MADRID
                                MASTER UNIVERSITARIO EN INGENIERÍA DE REDES Y SERVICIOS TELEMÁTICOS

AUTOR:   ANDRÉS CÁRDENAS CÓRDOVA

                                CONFIGURACIONES ADICIONALES PARA ACOPLAMIENTO ENTRE PLATAFORMA OSM Y OPENSTACK

                                ****OPENSTACK****

## Keystone tenant (Alternativo)
$ keystone --os-auth-url http://controller:35357/v2.0 tenant-create --name ktenant

-----

## Creación de Red Compartida externa ====> Red Shared llamada "default"
$ neutron router-create public
$ neutron net-create public --router:external --provider:physical_network external --
provider:network_type flat
$ neutron subnet-create public --name tenant-subnet --gateway 192.168.1.1 192.168.1.0/24
-----

$ neutron net-create default --shared
$ neutron subnet-create default 192.168.10.0/24 --gateway=192.168.10.1 --enable-dhcp --dns-
nameserver=8.8.8.8 --name=default_subnet ## Ingresar cualquier IP
#router default ----> public #public es el nombre de la red publica
$ neutron router-gateway-set default2public public #se asume que ya se tiene una red
llamada "public"
$ neutron router-interface-add default2public default_subnet #subnet de la red default

-----

                                ****OPEN SOURCE MANO****

## Acoplamiento Openmano con Openstack
-----
## Testear configuraciones en Openmano con los parámetros configurados de Openstack

$export OS_USERNAME=admin #se puede utilizar un usuario con derechos para create/delete
flavors/images
$export OS_PASSWORD=admin
$export OS_AUTH_URL='http://<openstack ip>:35357/v2.0'
$export OS_TENANT_NAME=admin
```



```

#Red Física, Usada por la etiqueta para [m12_type_vlan]:network_vlan_ranges at
'/etc/neutron/plugins/m12/m12_conf.ini'
$export OS_CONFIG="dataplane_physical_net: physnet_sriov"

#Imágenes a ser usadas. Esto sobrescribirá el "VNFC":"VNFC image" campo del descriptor VNF

$export OS_TEST_IMAGE_PATH_CIRROS="http://download.cirros-cloud.net/0.3.3/cirros-0.3.3-
x86_64-disk.img" #igual como "locación" de imagen de metadata
$export OS_TEST_IMAGE_PATH_LINUX="http://download.cirros-cloud.net/0.3.3/cirros-0.3.3-
x86_64-disk.img"
$export OS_TEST_IMAGE_PATH_LINUXDATA=/mnt/repository/linuximage.qcow2 #same as image
"location" metadata

-----

## Proceso de Acoplamiento

$ openmano tenant-create mytenant
$ export OPENMANO_TENANT= <id tenant>
$ echo "export OPENMANO_TENANT= <id tenant>" >> .bashrc
$ openmano datacenter-create --type openstack --description "Openstack Datacenter"
openstack http://10.0.10.11:35357/v2.0
$ openmano datacenter-attach openstack --user=admin --password=xxxx --vim-tenant-name=admin
$ openmano datacenter-netmap-upload -f --datacenter openstack

-----

## Ejecución de Juju
$ sudo juju bootstrap -v
$ sudo juju juju deploy juju-gui --to 0

-----

****RELOAD OSM****

## Volver a poner en marcha OSM en conjunto con Openstack
## Tener en cuenta el correcto funcionamiento de los distintos componentes
## Revisamos el estado de cada uno con los siguientes comandos:

====>>> En RO
$ service-openmano status
$ service-openmano restart # En el caso de que el servicio openmano se encuentre en
estado "stopped"
$ openmano datacenter-list # Para verificar si todavía está configurado el VIM en
OpenMano

```

```
=====>>> En CM
# Acceder a la url: https://<jujuserver-ip> ==> (10.1.0.6)
# Si no existe buen resultado ejecutar el siguiente comando y volver a intentar con la URL
$ sudo service squid-deb-proxy restart
```

```
=====>>> En SO
$ systemctl status rulp      #verificamos si se encuentra activo el servicio, en el caso
negativo aplicamos el siguiente comando
$ sudo systemctl start rulp  # Reiniciamos el servicio
# Procedemos a verificar ingresando a la URL: https://<rftserver-ip>:8000 ==> (10.1.0.5)
```

****COMANDOS ADICIONALES****

En Openstack

```
$source bin/admin_openrc.sh      #script de ejecución de credenciales de seguridad
$nova tenant-network-list        #Enlista redes inquilinas creadas
$neutron router-create <router-name> #Creación de router
$neutron router-list             #Enlista routers creados
```

En Juju

```
$juju add-machine ssh:root@<dir-ip-host>      #Agregar equipo al cloud
$juju add-unit <nombre-unidad>                #Agrega unidades
$juju deploy <nombre-unidad>                  #Despliega unidades
```

##En OpenMano

```
$openmano vnf-create ~/PATH/<fichero.yaml>    #Creación de VNFS
$openmano escenario-create ~/PATH/<fichero.yaml> #Creación de escenarios
```