



OpenStack Foundation Report

# Accelerating NFV Delivery with OpenStack

Global Telecoms Align Around Open Source Networking Future



## CONTRIBUTORS

Russell Bryant, *Senior Principal Software Engineer*, **Red Hat**

Kathy Cacciatore, *Consulting Marketing Manager*, **OpenStack Foundation**

Stephen Gordon, *Senior Technical Product Manager*, **Red Hat**

Eric Ji, *Senior Manager, Partner Marketing and Technology Alliances*, **Mirantis**

Armando Migliaccio, *Project Technical Lead (PTL) for OpenStack Neutron; Software Architect*, **HP Networking**

Iben Rodriguez, *NFVI Architect*, **OPNFV Consulting**

Alan Sardella, *Senior Technical Marketing Manager, OPNFV Project*, **Linux Foundation**

Tapio Tallgren, *Lead SW Architect in MBB Architecture*, **Nokia**

Brandon Wick, *Integrated Marketing Manager, OPNFV Project*, **Linux Foundation**

Ashlee Young, *Distinguished Architect*, **Huawei Technologies**

## NFV USER CONTRIBUTORS

Truman Boyes, *CTO Office, Head of Networks*, **Bloomberg LP**

Axel Clauberg, *Vice President, IP and Optical Architecture*, **Deutsche Telekom**

Toby Ford, *OpenStack Board of Directors; AVP, Cloud Technology, Strategy and Planning*, **AT&T**

Takayuki Kamei, *NFV / OpenStack /Application Engineer*, **NTT Communications**

Toshikazu Ichikawa, *Senior Research Engineer*, **NTT Software Innovation Center**

Jincheol Kim, *Professional Research Scientist, Software Engineer and Data Scientist*, **SK Telecom**

Fernando Oliveira, *Fellow SDN/NFV/Cloud Architect*, **Verizon**



This work is licensed under the Creative Commons Attribution-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nd/4.0/>

“Network Functions Virtualization (NFV) is now synonymous with OpenStack. When people say NFV, there is an implication that they are talking about OpenStack.”<sup>1</sup>

## Executive Overview

Expensive. Proprietary. Inflexible. These were some of the pain points business leaders experienced with traditional networking, and what prompted a consortium of network operators to develop something new. Network Functions Virtualization (NFV) allows telecom and enterprise network operators to control their networking functions—physical, virtual and functional domains—using commercial off-the-shelf hardware, and open source software as a single control pane for management and orchestration.

Early on, telecommunications companies and networking vendors recognized the potential for OpenStack as the platform for NFV, so they began working with vendors and developers in the OpenStack community to optimize OpenStack software for NFV use cases.

Both the European Telecommunications Standards Institute and Linux Foundation collaboration project OPNFV have defined specifications and released reference platforms for NFV that select OpenStack as the Virtualization Infrastructure Manager. Additionally, OpenStack is the dominant choice for additional management and orchestration functions.

An independent evaluator<sup>2</sup> tested the interoperability between four NFV infrastructure platforms that use OpenStack and various virtualized network functions. While the evaluator noted that interoperability is still a work in progress, especially with emerging NFV technology, the majority of configurations surpassed the evaluator’s expectations with a “great result.”

This paper describes NFV, its business value, and how OpenStack supports NFV. It details specific projects, use cases, and the experience of major carriers and enterprises including AT&T, Verizon, NTT Group, SK Telecom, and Bloomberg.

Although NFV is in its infancy, NFV on OpenStack offers an agile, scalable, and rapidly maturing platform with compelling technical and business benefits for telecommunications providers and large enterprises.

---

<sup>1</sup> “Dimensioning OpenStack Neutron for NFV Systems”, Mark Lambe writing for SDx Central, September 2014. <https://www.sdxcentral.com/articles/contributed/dimensioning-openstack-neutron-nfv-systems-mark-lambe/2014/09/>.

<sup>2</sup> [http://img.lightreading.com/downloads/NIA-Test-Report-Final.pdf?p\\_redirone=yes&pidl\\_promo=](http://img.lightreading.com/downloads/NIA-Test-Report-Final.pdf?p_redirone=yes&pidl_promo=).

# Exploring NFV

What is Network Functions Virtualization (NFV)? Simply put, it's a new way to define, create, and manage networks by replacing dedicated network appliances with software and automation. To put it into context, it's a continuation of the IT mindshift away from physical hardware that's inflexible, proprietary, and expensive. In an NFV environment, a virtual network function (VNF) takes on the

responsibility of handling specific network functions that run on one or more virtual machines (VMs), on bare metal, or in containers, on top of the physical networking infrastructure.

VNFs range from mobile deployments, where mobile gateways (e.g. SGW, PGW, etc.) and related functions (e.g. MME, HLR, PCRF, etc.)<sup>3</sup> are deployed as VNFs, to deployments with "virtual"

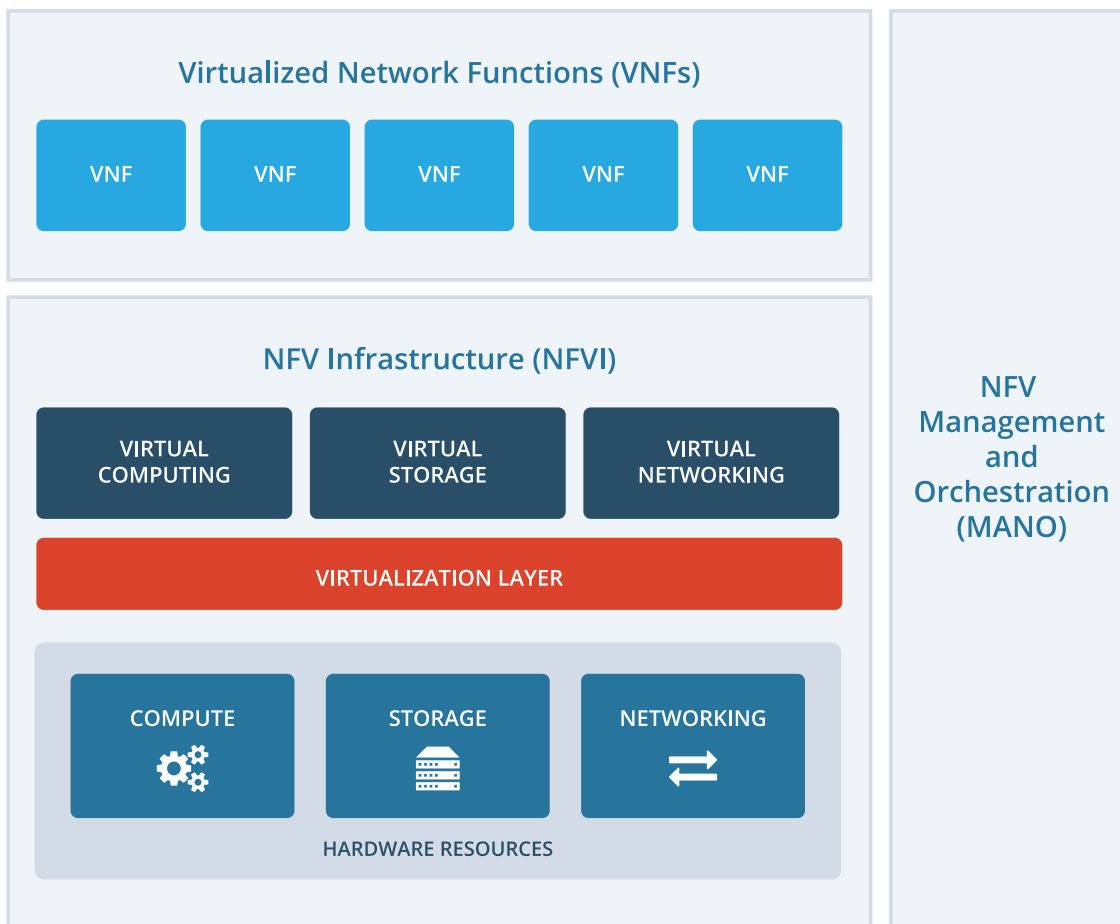


Figure 1. NFV functional overview

<sup>3</sup> A fuller description of these mobile gateways and functions, which are part of the System Architecture Evolution, can be found at [https://en.wikipedia.org/wiki/System\\_Architecture\\_Evolution](https://en.wikipedia.org/wiki/System_Architecture_Evolution).



Figure 2. Example physical network functions for VNF replacement

**NFV and software-defined networking (SDN)** complement each other, but solve different problems in different environments across different domains. SDN emerged to make network devices programmable and controllable from a central element. NFV aimed at accelerating service innovation and provisioning using standard IT virtualization technologies. SDN requires new interfaces, control modules, and applications, while NFV typically involves moving networking applications to virtual machines or containers that run on commodity hardware. NFV is highly complementary to SDN, but not dependent on it (or vice-versa), although the two concepts and solutions can be combined and potentially greater value accrued.

customer premise equipment (CPE), tunneling gateways (e.g. VPN gateways), firewalls or application level gateways and filters (e.g. web and email traffic filters) to test and diagnostic equipment (e.g. to monitor service level agreements).<sup>4</sup>

For a deeper exploration of NFV through the original ETSI NFV white paper and a technical overview of OPNFV, visit [https://portal.etsi.org/NFV/NFV\\_White\\_Paper.pdf](https://portal.etsi.org/NFV/NFV_White_Paper.pdf) and <https://www.opnfv.org/software/technical-overview>.

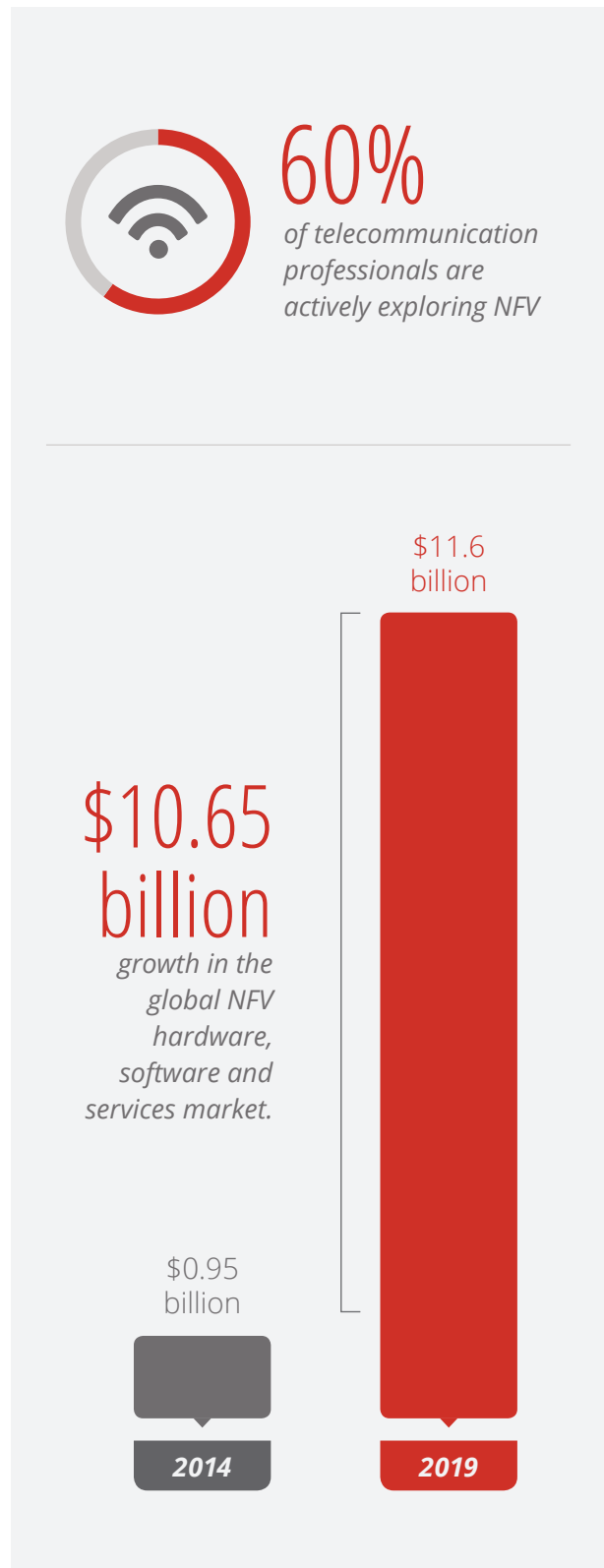
The benefits of NFV stem from the fact that it runs on general purpose servers and switches in virtual machines or containers and is built with standard open APIs. NFV relies on open source development and offers a wide range of networking capabilities dynamically and adaptively. In general, the aim of NFV is to offer agility, flexibility, and simplicity. The detailed operational and technical benefits that network operators expect from NFV implementation include:

<sup>4</sup> See <https://www.opnfv.org/software/technical-overview>.

- Network flexibility via programmatic provisioning
- Taking advantage of the open source pace of innovation—ever-emerging improvements in both the telecom and the traditional IT space
- Full choice of modular drivers and plug-ins
- Accessibility via API, enabling faster time to market for new capabilities
- Lower costs by replacing with COTS hardware, better price/performance
- Reduced power consumption and space utilization
- Operational efficiency across datacenters via orchestration: managing thousands of devices from one console
- Visibility: automated monitoring, troubleshooting and actions across physical and virtual networks and devices
- Boosts performance by optimizing network device utilization
- SLA-driven resource allocation (initial and ongoing)
- QoS: performance, scalability, footprint, resilience, integration, manageability
- Policy-driven redundancy
- Application level infrastructure support

On a business level, NFV users gain agility and efficiency alongside CapEx, OpEx, power and space reductions, and also gain the potential for additional service revenues.

NFV is a mainstream capability that's being rapidly adopted by telecom operators. A 2015 Heavy Reading global survey commissioned by OPNFV found that nearly 60 percent of telecommunication professionals are actively exploring NFV.<sup>5</sup> Consulting firm IHS ® Infonetics forecasts the global NFV hardware, software, and services market will increase more than tenfold to \$11.6 billion in 2019 from \$0.95 billion in 2014.<sup>6</sup>



5 <https://www.opnfv.org/news-faq/press-release/2015/11/opnfv-and-heavy-reading-release-results-survey-evaluate-project%E2%80%99s>

6 "NFV Market to Grow More than 5-Fold Through 2019, Says IHG", July 2015, © 2015 IHS Inc. All rights reserved. <http://www.infonetics.com/pr/2015/NFV-Market-Highlights.asp>.

“OpenStack plays a key role in OPNFV—in the community, the projects, and the platform. We look forward to continued collaboration as we move the industry closer to open source NFV.”

**Jonathan Bryce, Executive Director, OpenStack Foundation, January 2016**

OPNFV, built on widespread developer collaboration across many telecommunications providers and enterprises, is well-positioned to integrate the work of standards bodies, open source communities and commercial suppliers to deliver a functional, standard open source NFV platform. OPNFV project members, including AT&T, China Mobile, NTT DOCOMO, Telecom Italia, Vodafone, Ericsson, Huawei, Red Hat, Intel, CenturyLink, KT, Orange, SK Telecom, and Sprint initially focused on the NFV infrastructure, including the virtualization infrastructure (NFVI) and the virtualized infrastructure manager (VIM). In December 2015, OPNFV expanded its scope to include MANO in future releases. OPNFV offers the industry end-to-end integration and bare metal deployment of a modular platform for NFV. In addition to OpenStack, open source projects OpenDaylight, Open vSwitch and KVM are used in the first release of the reference platform.

OPNFV has an “upstream first” philosophy which states that necessary adaptations to the included projects will be developed as much as possible within the scope of such projects—OpenStack for example. The OPNFV community contributes developers and code to accelerate the features in OpenStack.

NFV is incorporated across the board within OpenStack. OpenStack NFV feature proposals and development are integral to all projects and processes:

- Telecom users bring their requirements to the User Committee
- ETSI NFV ISG requests for OpenStack features are published to their website
- The OPNFV and OpenStack communities are in close contact, and processes are in place to succinctly identify and advocate NFV features<sup>7</sup>

---

<sup>7</sup> See <https://wiki.opnfv.org/community/openstack> for more information.

# Why OpenStack for NFV

The OpenStack platform provides the foundation for the NFV architecture, which is essentially a fit-for-purpose cloud for deploying, orchestrating and managing virtual network functions. OpenStack enables multiple datacenter management from a single pane of glass, complete with common security,

identity services, APIs, and user interfaces. The open, modular and interoperable framework of the OpenStack project offers telecoms and enterprises the ability to design the NFV system of their choosing, without unnecessary components.

## Why OpenStack is “synonymous” with NFV:

- + Proven architecture for largest public clouds, which are available to masses on COTS hardware (closest comparable implementation)
- + Standardized interfaces between NFV elements and infrastructure
- + Pluggable architecture with documented APIs, UIs, shared services, operations, automation options for VNFs and other function integration
- + All popular open source and commercial network plug-ins and drivers
- + Proven telecom and enterprise implementations: AT&T, China Mobile, SK Telecom, Ericsson, Deutsche Telekom, Comcast, Bloomberg, and more
- + Outstanding global community contributing to rapid pace of innovation, working on unique NFV requirements from users, related communities and standards developing organizations
- + OpenStack Neutron networking project mature and in production use in most all installations
- + NFV features in every release since 2013
- + Network/element deployment automation, rollout efficiency
- + Resource pools cover all network segments
- + Broad industry support



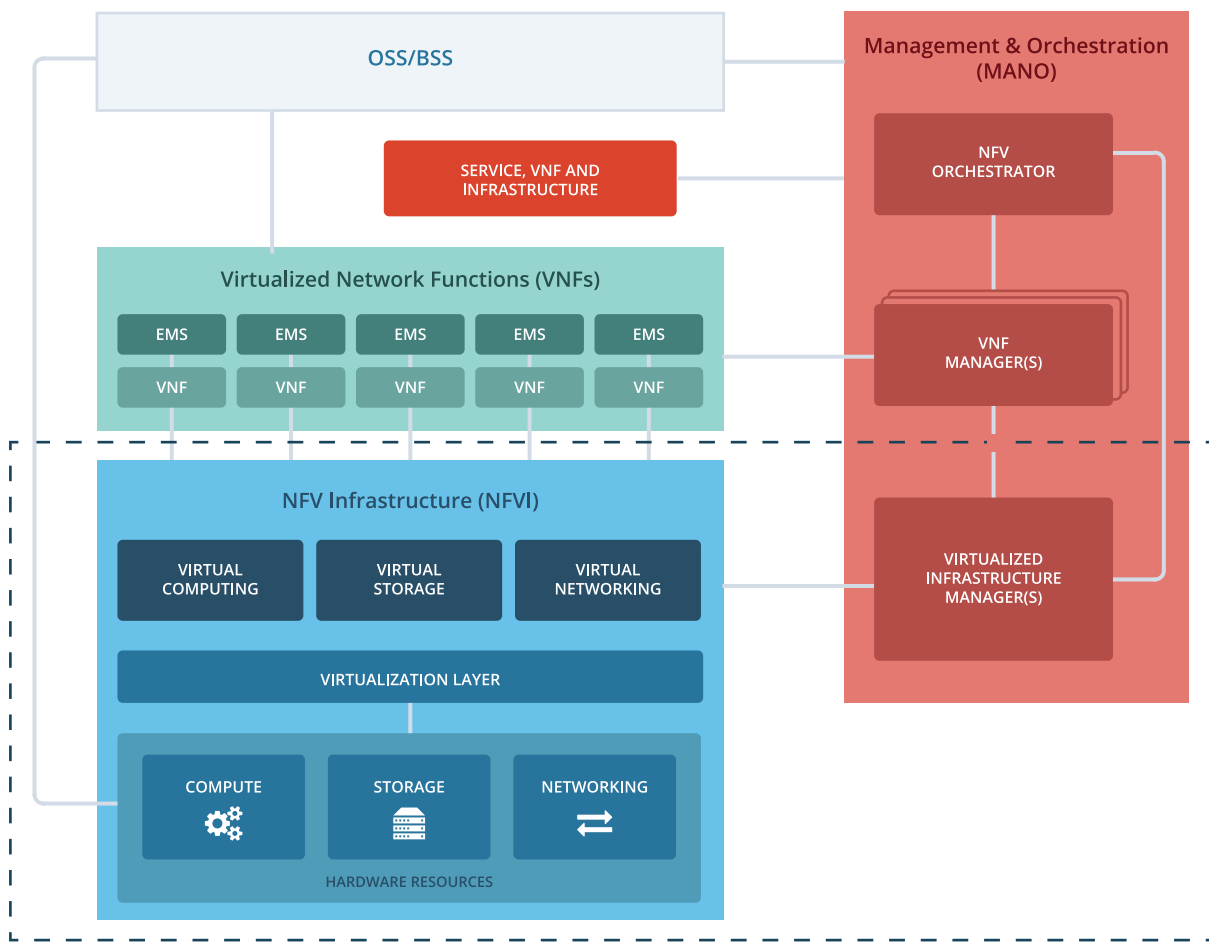


Figure 3: ETSI NFV Specification Model with OPNFV Arno Release Subset (dotted box)

OpenStack is identified as one main component in the ETSI NFV architectural framework. As a reference implementation of the ETSI NFV specification, the OPNFV project has implemented OpenStack for the Virtualization Infrastructure Manager (VIM) components in the first release, Arno (releases are named for rivers), on the lower right side of Figure 3.

It should be noted that the ETSI NFV diagram wasn't meant to be a formalized architecture—it's a model. This is why Figure 4 (Arno) doesn't look exactly like an implementation of Figure 3 (ETSI spec), although it is.

With the release of Arno, OPNFV has taken the first step toward implementing the vision of NFV that ETSI first described. The first release focuses on the VIM and NFVI with five projects.

This early open source implementation aims to accelerate the development of VNFs and other NFV components by defining a consistent, functional stack that developers will adopt as a de facto standard. Arno also provides telecoms and enterprises with a base for integration and testing, enabling faster iteration in the future.

The next release, Brahmaputra<sup>8</sup>, slated for February 2016 delivery, is planned as the first "lab-ready" release, incorporating numerous enhancements in areas such as installation, installable artifacts, continuous integration, improved documentation, and sample test scenarios. The project is evolving towards lab-readiness for the testing and interoperability of NFV functionality and use cases. In addition to support for more virtual networking solutions, Brahmaputra plans also include

8 OPNFV Brahmaputra Release Page, <https://wiki.opnfv.org/releases/brahmaputra>.

support for multi-site data centers, additional fault management, and upgradeability and deployment solutions. It will include approximately 30 projects, demonstrating effective interworking with several upstream communities. Brahmaputra will include the Liberty release of OpenStack; OpenStack remains the sole VIM platform.

The next section will describe the OpenStack projects key to NFV, including the “why and how.” It will also provide recently incorporated examples of NFV features and plans for the future.

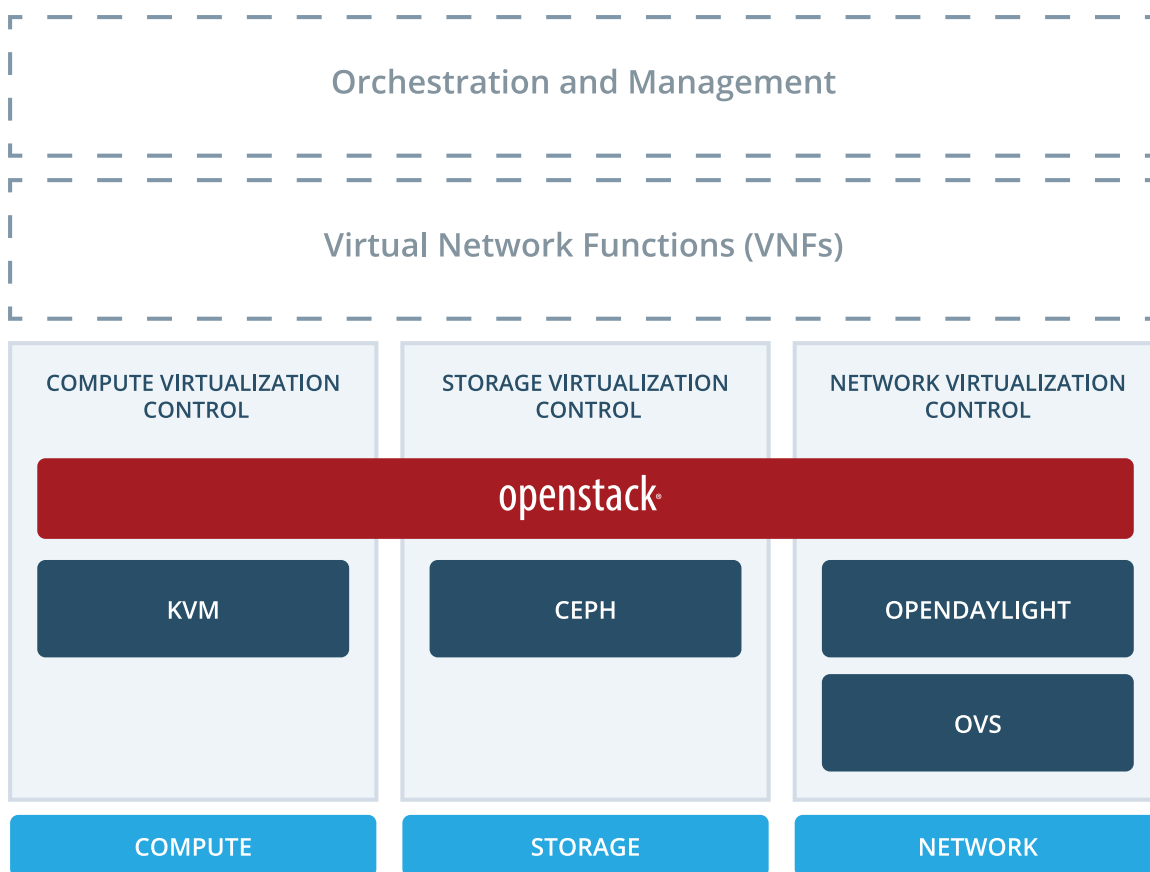


Figure 4: OPNFV Arno Instantiation with Open Source Software

# Telecom Requirements for NFV with OpenStack

Today, in ETSI NFV and OPNFV, **OpenStack is fundamental to the Virtualized Infrastructure Manager (VIM)**. The VIM is the part of the Management and Orchestration (MANO) function that controls the assignment of virtualized compute, storage and network resources from the NFVI to support the VNFs. The primary OpenStack projects involved are

- OpenStack Compute (code named Nova) for managing virtual or bare metal servers. A related project—Magnum—uses Nova instances to run application containers.
- OpenStack Block Storage (code named Cinder) for virtual storage, supporting practically any storage back end
- OpenStack Networking (code named Neutron) providing virtual networking.

From this point forward, we'll use the ubiquitous code names for each OpenStack project. Newer projects are only known by their code names.

There are too many NFV-enhancing features in OpenStack to list. Primary telecom requirements will be accompanied by a few examples of the features OpenStack includes today and future plans. Telecom environments and NFV have stringent requirements in the areas of scaling, performance, and faster and more deterministic responses to failures, as well as IPv6 and many more specific features. The requirement for deterministic responses refers to predictable performance, including how to avoid the vagaries of the hypervisor and host OS scheduler affecting performance-sensitive applications.

OpenStack has continually addressed carrier-grade performance, scalability, and resiliency requirements. All projects are charged with prioritizing features for scalability, resiliency, manageability, modularity and interoperability in the Mitaka release and beyond.<sup>9</sup> The OpenStack Technical Committee has begun the conversation to further define a consistent scaling framework based on user expectations. All projects will implement features with regard to this common framework as well.

## Performance

The performance requirement is largely about high-performance packet processing: how to get a packet off the network, into a VM, processed quickly and back out again on the network. One of the available techniques is to give VMs direct physical access to the network via SR-IOV. DPDK-accelerated Open vSwitch for fast packet processing and multi-queue vhost-user with accelerated virtual switches are also supported in **Neutron**.

A set of related enhancements allows operators to define flavors, and application owners to define image properties, which between them control things like vCPU topology, vCPU to pCPU pinning, the placement of applications in relation to NUMA nodes and making huge pages available to the applications.

In the future we can expect real-time enablement for Cloud Radio Access Networks and further performance tuning of the hypervisor and network (by Nova and

<sup>9</sup> Watch key Project Technical Leaders (PTLs) cover planned Mitaka features for their projects in video interviews accessible at this YouTube playlist: [https://www.youtube.com/playlist?list=PLKqaoAnDyfgpMle\\_oc7rRrjvQ80pDA1jE](https://www.youtube.com/playlist?list=PLKqaoAnDyfgpMle_oc7rRrjvQ80pDA1jE).

Neutron respectively) as well as the guest configuration. <https://wiki.openstack.org/wiki/VirtDriverGuestCPUMemoryPlacement> lists the key things that can help optimize compute performance.

While it would seem natural that Network Functions Virtualization comes under networking, and thus Neutron, in fact, much of the work involves **Nova**. Nova manages the life cycle of compute instances in an OpenStack environment. Responsibilities include spawning, scheduling and decommissioning of machines on demand. Most of Nova is relevant to NFV use cases. For example, the scheduler is crucial for performance and resiliency. It must launch new instances fast, both initially and especially in reaction to fault detection.

## Scalability, High Availability and Resiliency

Traditionally, telecoms emphasize the need for “carrier grade” infrastructure, demanding extremely high reliability for each infrastructure component. One of the architectural tenets of cloud platforms (including OpenStack) is that both scalability and reliability are achieved via massive horizontal scale. This is a new approach for many telecoms in that it pushes more of the HA requirement up to the application. In cloud environments, an individual host cannot in and of itself provide “five nines” of uptime, but an application with many instances across many hosts in a distributed cloud can. Recognizing that failures are bound to occur, the NFV conversation on OpenStack has to focus on resiliency monitoring, fault detection and response.

Cells (groups of hosts) management for geographic scaling is a Nova component. Cells enable the deployment of larger OpenStack clouds by providing a way to group together resources to be managed more easily. Administrators can partition existing resources into cells and the system will know where to find them.

As PayPal’s Anand Palmisani puts it, “[PayPal]... used the new and improved Nova Cells service for the first time to increase availability and scalability. For the uninitiated, the Nova Cells service adds scaling and geographic distribution capabilities without the complicated database or MQ clustering of Zones. They also allow operators to separate host scheduling from cell scheduling.”<sup>10</sup>

OpenStack continues to incorporate high availability and resiliency functionality in every release. Most recently, the Liberty release includes these network-specific examples:

- Router high availability (L3 HA / VRRP) when layer 2 population (l2pop) is enabled
- VPNaaS reference drivers with HA routers
- Networks used for VRRP traffic for HA routers may now be configured to use a specific segmentation type or physical network tag.
- An OVS agent may be restarted without affecting data plane connectivity.
- Event alarms trigger an action when an event is received.

But these enhancements may not reassure every company that’s considering NFV on OpenStack. An independent view of OpenStack’s role in NFV availability, entitled “Is Carrier-grade NFV Really Important?”<sup>11</sup>, addresses and debunks common beliefs—that users must architect HA with redundant components and VNFs, and enhance the platform to maintain state during failures. The author, Tom Nolle, President of CIMI Corp., concludes that OpenStack does exactly what it claims to do, but VNF developers need to build HA into their VNFs—just as other cloud-aware developers build HA into their apps. There’s a good example of this in Project Clearwater. Clearwater is an open source implementation of IMS (the IP Multimedia Subsystem) designed for massively scalable deployment in the cloud. Clearwater combines the economics of over-the-top style service platforms with the standards compliance expected of telecom-grade communications

10 “How PayPal Runs the World’s Largest Private OpenStack Cloud”, December 8, 2015, [http://blog.appformix.com/how-paypal-runs-the-worlds-largest-private-openstack-cloud?awesm=awe.sm\\_eN70t](http://blog.appformix.com/how-paypal-runs-the-worlds-largest-private-openstack-cloud?awesm=awe.sm_eN70t).

11 <http://blog.cimicorp.com/?p=2372>



## Recognizing that failures are bound to occur, the NFV conversation on OpenStack has to focus on resiliency—monitoring, fault detection and response.

network solutions, and its cloud-oriented design makes it extremely well suited for deployment in an NFV environment.

Resiliency is implemented in other, multiple ways as well. Bloomberg, a large network operator and OpenStack NFV user, is moving away from synchronizing states to less stateful and more autonomous functions across many computers, to spread resiliency throughout the environment. They are deploying many small firewall and load balancer VNFs as opposed to fewer large ones. This provides better SLAs at a higher level. The trade-off is management of more systems but the approach reduces risk.

For the OPNFV reference platform, Doctor is the fault management and maintenance project. The goal of Doctor is to build a fault management and maintenance framework for high availability of network services on top of virtualized infrastructure. The key feature is immediate notification of unavailability of virtualized resources from VIM, to process recovery of VNFs on them.

The Doctor community affects this mission primarily in OpenStack's Monasca project—reflecting OPNFV's "upstream first" approach. Monasca is a high performing, scalable, reliable and fault-tolerant Monitoring as a Service (MONaaS) solution that scales to service provider levels of metrics throughput. Performance, scalability and high availability have been designed in from the start.

Monasca can process hundreds of thousands of metrics/second and can offer data retention periods of greater than a year with no data loss while still processing interactive queries. Other key features include:

- Rest API for storing and querying metrics and historical information. Most monitoring solutions use special transports and protocols, such as CollectD or NSCA (Nagios). In Monasca, http is the only protocol used. This simplifies the overall design and also allows for a much richer way to describe the data via dimensions.
- Multitenant and authenticated
- Metrics defined using a set of (key, value) pairs called dimensions
- Real-time thresholding and alarming (single and compound alarms) on metrics
- Compound alarms
- Monitoring agent supporting built-in system and service checks and Nagios checks and statsd.

### Multisite

Telecoms require an NFV infrastructure distributed across multiple geographical locations. The platform must be able to support application-level redundancy across different datacenters, network management across multiple sites, and between physical and virtual infrastructure, multisite image replication, and global and per-site quota management.

Multiple connected OpenStack deployments as VIMs and high availability among them are required for a distributed, multi-geography NFV infrastructure. The **OPNFV Multisite** Virtualized Infrastructure project focuses on enhancements to the OpenStack Nova, Cinder, Neutron, Glance (Image Service), Ceilometer (Telemetry), and Keystone (Identity) projects, so that OpenStack as the VIM is able to support multi-site NFV clouds. Learn more about the OPNFV Multisite project at <https://wiki.opnfv.org/multisite>.

In current data centers, deployment of a service chain is through static, complex, and rigid configurations since they are tightly coupled with physical network topology and physical resources.

## Service Function Chaining

Service Function Chaining (SFC) is a mechanism for overriding the basic destination-based forwarding that is typical of IP networks. A simple example of a service chain would be to force all traffic from point A to point B to go through a firewall even though the firewall is not literally between point A and B from a routing table perspective. A more complex example is an ordered series of functions, each implemented in multiple VMs, such that traffic must flow through one VM at each hop in the chain but uses a hashing algorithm to distribute different flows across multiple VMs at each hop.

In current data centers, deployment of a service chain is through static, complex, and rigid configurations since they are tightly coupled with physical network topology and physical resources. The introduction of new services into a network usually requires the reconfiguration of most (if not all) network elements. The physical chain must be redefined for NFV with automatic setup according to service chain requirements. Scaling-out the network of these service functions to handle added load or scaling-in to reduce the resource usage is an integral part of an SFC solution.

In OpenStack, there are two steps in creating a service chain. First, services VMs, such as firewall VMs, need to be created and connected to a Neutron network via Neutron ports. After that, selected traffic flows need to be steered through an ordered sequence of these service VM ports.

Prior to OpenStack's Liberty release OpenStack already supported creation of service VMs and

attachment of these service VMs to Neutron network ports. In the Liberty release, Neutron was extended with the experimental SFC project `networking-sfc`. Features of `networking-sfc` are:

- Creation of Service Function Chains consisting of an ordered sequence of Service Functions. SFs are virtual machines (or potentially physical devices) that perform a network function such as firewall, content cache, packet inspection, or any other function that requires processing of packets in a flow from point A to point B.
- Reference implementation with Open vSwitch
- Flow classification mechanism (ability to select and act on traffic)
- Vendor neutral API
- Modular plugin driver architecture.

Detailed, subject-to-change documentation is available at <http://docs.openstack.org/developer/networking-sfc/>.

## Addressing the Rest of MANO: NFVO and VNFM

In December 2015, OPNFV elected to expand the scope of projects as needed to facilitate NFV deployments.<sup>12</sup> For instance, the community will now incubate and propose projects on additional topics, including MANO (the “heart and brains” of NFV). In addition to the VIM, covered earlier, NFV MANO includes:

---

<sup>12</sup> See the blog describing the removal of scope constraints at <https://www.opnfv.org/news-faq/blog/2015/12/opnfv-board-re-moves-scope-constraints>.

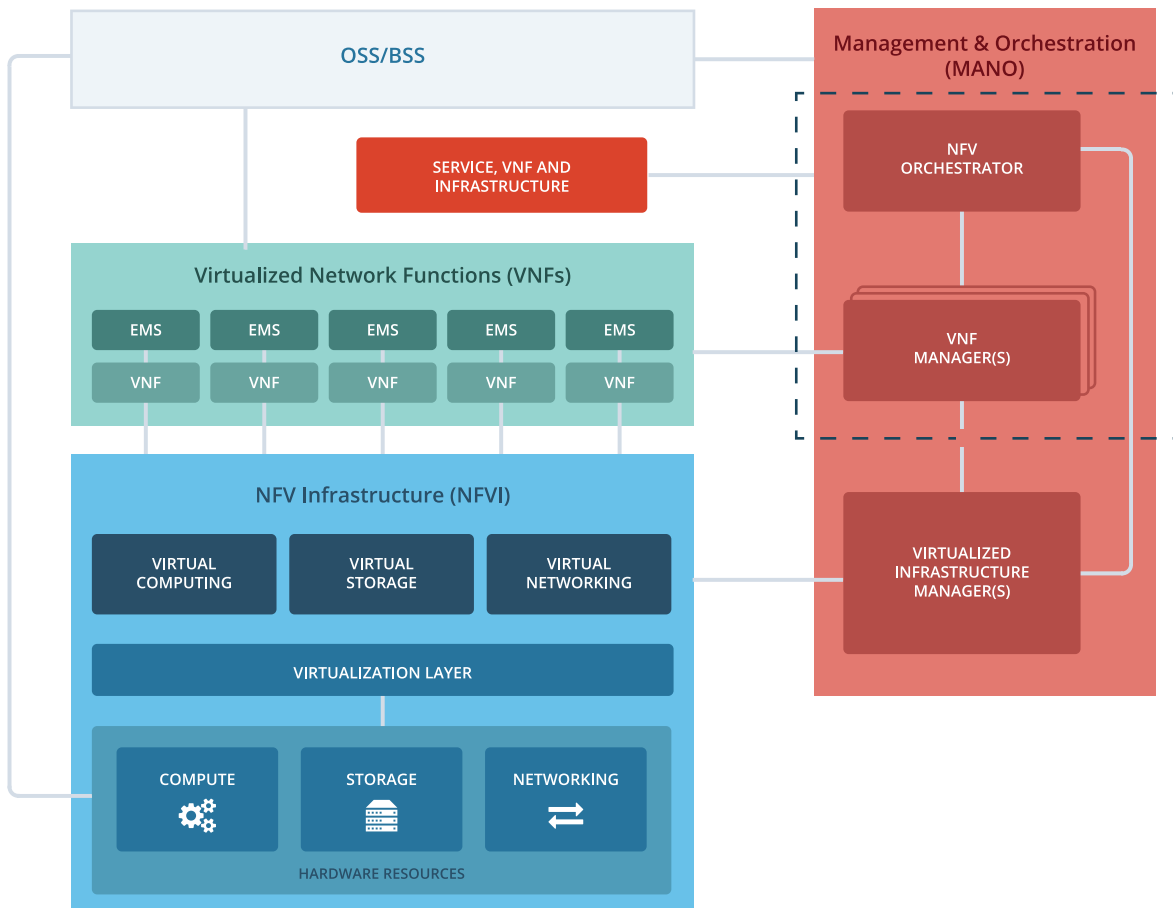


Figure 5: OpenStack Tacker addresses ETSI NFV MANO NFVO and VNFM functions

- **NFV Orchestrator:** Responsible for onboarding of new network services (NS) and virtual network function (VNF) packages; NS life-cycle management; global resource management; validation and authorization of network functions virtualization infrastructure (NFVI) resource requests
- **VNF Manager:** Oversees life-cycle management of VNF instances; coordination and adaptation role for configuration and event reporting between NFVI and E/NMS (element or network management system).

The OpenStack **Tacker** project addresses the NFVO and VNFM. Tacker is building an Open NFV Orchestrator with an integrated general

purpose VNF Manager to deploy and operate Virtual Network Functions (VNFs), with Service Function Chaining. It is based on the ETSI MANO Architectural Framework and provides a fully functional stack to orchestrate VNFs end-to-end (Figure 5). Tacker provides a VNF Catalog to on-board VNF Descriptors (written using OASIS TOSCA NFV standards) and provides APIs for Life-Cycle Management of VNFs along with capabilities like VNF monitoring, auto-scaling and self-healing. Tacker plans to expand to NFVO capabilities like Network Service Descriptors (NSD) and Forwarding Graph Descriptor (VNFFGD) support in upcoming cycles. Tacker utilizes OpenStack Compute (Nova), Neutron and Heat to execute the VNF life-cycle:

## “OpenStack NFV deployments are the wave of the future.”<sup>13</sup>

- Tacker API deploys VNF from the VNF Catalog
- Instantiates one or more VMs described in TOSCA NFV template
- Tacker facilitates configuration injection into VNFs and provides a loadable framework for KPI monitoring and healing
- Terminate VNF will delete all VMs and other resources associated with VNF instance

Tacker has many supporters and developers in the OPNFV community and is projected to be an option in a future OPNFV release. Although there are no definitive plans at this time, the Tacker PTL says “the OPNFV C-release could integrate nicely with Tacker’s OpenStack Mitaka release.” There are several technical Tacker presentation videos from the November 2015 OPNFV Summit.

Watch the Tacker presentation and demo at <https://www.youtube.com/watch?v=EfqWARz25Hg>. For more information on Tacker, visit <https://wiki.openstack.org/wiki/Tacker>.

### Other Key OpenStack Projects

Several additional OpenStack projects support NFV implementations. Here are the ones to watch.

- **Astara:** Provides integrated network service orchestration (routing, firewall, load balancing, VPN) for connecting and securing multitenant OpenStack environments. Wiki: <https://wiki.openstack.org/wiki/Astara>
- **Blazar (previously Climate):** Reservation-as-a-Service, including resource reservation, reservation update and “give me an offer”/best effort reservation, and reservation scheduling. Provides solution to requirements from OPNFV Promise project. Wiki: <https://wiki.openstack.org/wiki/Blazar>.
- **Congress: Policy-as-a-Service.** Provides governance as a service across any collection of cloud services in order to monitor, enforce, and audit policy over dynamic infrastructure. Wiki: <https://wiki.openstack.org/wiki/Congress>.
- **Mistral:** Workflow service. Allows description of complex business processes (workflows) as a set of tasks and task relationships, such that Mistral handles state management, correct execution order, parallelism, synchronization and high availability. Mistral also provides flexible task scheduling. Wiki: <https://wiki.openstack.org/wiki/Mistral>.
- **Neutron:** an OpenStack project to provide “networking as a service” between interface devices (e.g., vNICs) managed by other Openstack services (e.g., Nova). Neutron offers flexibility and choice with drivers and plug-ins from numerous leading telecom vendors so users do not have to worry about altering their APIs or modifying code if they decide to switch the underlying implementation technology. This paper covers Neutron features for NFV performance (above). 88 percent of OpenStack users have implemented Neutron. For more information, please visit <http://docs.openstack.org/developer/neutron/>.
- **Neutron “Stadium” subprojects:** An official set of Neutron subprojects, the Stadium includes many NFV-related projects and maintains support for most popular drivers and plug-ins. See the full list at <http://governance.openstack.org/reference/projects/neutron.html>.
- **Senlin:** A clustering service and libraries for the management of groups of homogeneous objects exposed by other OpenStack services. Wiki: <https://wiki.openstack.org/wiki/Senlin>.

<sup>13</sup> “Brocade Talks Real-World OpenStack NFV Deployments”, October 2015 <https://www.sdxcentral.com/articles/featured/openstack-nfv-brocade-demofriday-sign-up/2015/10/>.



# NFV to OpenStack

## Requirements Processes

NFV requirements, similar to others, are introduced in various ways: by user request, the ETSI NFV ISG or OPNFV community. The tables below show these requirements become OpenStack blueprints, which are used to track the implementation of significant features.

### OpenStack Users and Ecosystem

OpenStack users submit requirements via the OpenStack User Committee or Operators

Working Group. Requirements are also initiated by the OpenStack ecosystem, often based on relationships with and Request for Proposals (RFPs) from telecom clients. These requirements generally mirror the requirements from ETSI NFV or OPNFV, as the requesting companies begin implementing the standard framework.

Here are examples of telecom and ecosystem submitted requirements:

DESCRIPTION	SUBMITTED BY	OPENSTACK PROJECT	BLUEPRINT	STATUS
Support multiple IPv6 prefixes and addresses for IPv6 network	Comcast	Neutron	<a href="https://blueprints.launchpad.net/neutron/+spec/multiple-ipv6-prefixes">https://blueprints.launchpad.net/neutron/+spec/multiple-ipv6-prefixes</a>	Complete
Processor core affinity for a VM	Verizon	Nova	<a href="https://blueprints.launchpad.net/nova/+spec/virt-dedicated-cpus-placement-policy">https://blueprints.launchpad.net/nova/+spec/virt-dedicated-cpus-placement-policy</a>	Pending approval
Resource reservation	NTT	<ul style="list-style-type: none"> <li>Nova Reservation API</li> <li>Blazar</li> </ul>	<a href="https://wiki.openstack.org/wiki/Blueprint-nova-planned-resource-reservation-api">https://wiki.openstack.org/wiki/Blueprint-nova-planned-resource-reservation-api</a> <a href="https://blueprints.launchpad.net/blazar">https://blueprints.launchpad.net/blazar</a>	<ul style="list-style-type: none"> <li>Passed to Blazar project</li> <li>Varies</li> </ul>
NUMA topology awareness	Red Hat and Intel on behalf of Telefónica	Nova	<a href="https://blueprints.launchpad.net/nova/+spec/virt-driver-vcpu-topology">https://blueprints.launchpad.net/nova/+spec/virt-driver-vcpu-topology</a>	Complete
VLAN-aware VMs and support Neutron trunk ports in Nova	Ericsson	Neutron and Nova	<a href="https://blueprints.launchpad.net/neutron/+spec/vlan-aware-vm">https://blueprints.launchpad.net/neutron/+spec/vlan-aware-vm</a>	Started, planned for Mitaka
Support failure correlation	Huawei	Monasca	<a href="https://blueprints.launchpad.net/monasca/+spec/suppot-failure-correlation">https://blueprints.launchpad.net/monasca/+spec/suppot-failure-correlation</a>	New

## ETSI NFV ISG

The ETSI NFV Industry Specification Group (ISG) submitted formal requirements<sup>14</sup> to OpenStack in 2014, but has since made the process more open, publishing specification drafts and gaps<sup>15</sup>

for public review. The gaps result in OpenStack blueprints, which initiate and track OpenStack feature and project development through completion. Here are examples of how ETSI requirements are integrated into OpenStack's open design processes:

NFV ENTITY	ETSI NFV SPEC IDENTIFIER	DESCRIPTION	OPENSTACK PROJECT	OPENSTACK BLUEPRINT	STATUS
MANO Virtual Machine Descriptor	Vi-Vnfm and VDU in VNF Descriptor	PCI Passthrough SR-IOV support	Nova	<a href="https://blueprints.launchpad.net/nova/+spec/sriov-pf-passthrough-neutron-port">https://blueprints.launchpad.net/nova/+spec/sriov-pf-passthrough-neutron-port</a>	Complete
VIM & MANO Virtual Machine Descriptor	Vi Vnfm and Service VNF & Infrastructure Descriptor–Se–Ma	NUMA and IO locality	Nova	<a href="https://blueprints.launchpad.net/nova/+spec/sriov-pf-passthrough-neutron-port">https://blueprints.launchpad.net/nova/+spec/sriov-pf-passthrough-neutron-port</a>	Approved
Vi-VNFM, NFVO-Vi	Virtualized resource management	Operations comprising the management of resource reservations	Blazar	Blazar's blueprints at <a href="https://blueprints.launchpad.net/blazar">https://blueprints.launchpad.net/blazar</a>	Varies

## OPNFV Community

The OpenStack community receives NFV requirements via collaboration with OPNFV and directly from telecoms involved in the project. OPNFV can serve as an intermediary for telecoms that might not be contributing code upstream to OpenStack, yet want to share requirements and collaborate in a familiar language. These requirements are

assimilated into one or more OpenStack projects. OpenStack and OPNFV are broader in scope than one another, yet there is a large intersection between them. The OPNFV requirements and development processes are very much like OpenStack's. Many community members work on both projects. Here are examples of features defined, submitted and usually worked on by the OPNFV community:

OPNFV SUBMITTING PROJECT	DESCRIPTION	OPENSTACK PROJECT	OPENSTACK BLUEPRINT	STATUS (AS OF JAN. 2016)
Doctor	Change the state of compute service "down" immediately	Nova	<a href="https://blueprints.launchpad.net/nova/+spec/mark-host-down">https://blueprints.launchpad.net/nova/+spec/mark-host-down</a>	Complete
Doctor	Multiple blueprints for anomaly detection and sensor monitoring	Monasca	<a href="https://blueprints.launchpad.net/monasca/">https://blueprints.launchpad.net/monasca/</a>	Varies
Promise	Implement support of volume reservation	Blazar	<a href="https://blueprints.launchpad.net/blazar/+spec/basic-volume-plugin">https://blueprints.launchpad.net/blazar/+spec/basic-volume-plugin</a>	Approved, under evaluation
Multisite	Expose quiesce/unquiesce API	Nova	<a href="https://blueprints.launchpad.net/nova/+spec/expose-quiesce-unquiesce-api">https://blueprints.launchpad.net/nova/+spec/expose-quiesce-unquiesce-api</a>	In progress for Mitaka
Copper	Multiple blueprints for event-driven policy engine	Congress	<a href="https://blueprints.launchpad.net/congress/">https://blueprints.launchpad.net/congress/</a>	Varies
ONOS	Add a Neutron/ML2 plugin for ONOS	Neutron	<a href="https://blueprints.launchpad.net/neutron/+spec/onos-neutron-interaction">https://blueprints.launchpad.net/neutron/+spec/onos-neutron-interaction</a>	Complete

<sup>14</sup> [https://wiki.openstack.org/w/images/c/c7/NFV\(14\)000154r2\\_NFV\\_LS\\_to\\_OpenStack.pdf](https://wiki.openstack.org/w/images/c/c7/NFV(14)000154r2_NFV_LS_to_OpenStack.pdf)

<sup>15</sup> ETSI NFV Drafts, <http://www.etsi.org/technologies-clusters/technologies/nfv>.

# Production Ready in 2016

OpenStack for NFV will be production ready in 2016 based on development blueprints of documented telecom, OPNFV and ETSI NFV requirements. Having said that, the world's largest communications companies and enterprise network operators are implementing NFV with OpenStack today because of compelling benefits. Here are a few examples. Others include China Mobile, Telus Communications, Telecom Italia, Wells Fargo, and Telefónica.

## AT&T

In the last eight years, data traffic on the AT&T network has increased a staggering 100,000 percent, driven primarily by video. Keeping up by using more sophisticated, complex routers, switches and other vertically scaled gear just isn't feasible for much longer—performance, inefficiency and cost are huge issues.

AT&T's next generation network emulates the function of complex hardware with software that runs on standard off-the-shelf hardware. Powered by SDN and NFV technologies, AT&T can add capacity faster and push out upgrades at the speed of the Internet.

John Donovan, the company's senior executive vice president of technology and network operations, said there are now millions of AT&T wireless subscribers connected to virtualized network services—many will be relying on the AT&T Integrated Cloud (AIC), which is based on OpenStack. AT&T's internal tools and the customer-facing applications share the same code in the cloud. OpenStack is the most important of the SDN projects to AT&T's

current requirements and the company is also contributing to OpenNFV, OpenDaylight, and ONOS.<sup>16</sup>

AT&T is well on its way to implementing a common infrastructure for all VNFs, and the first VNFs are in production with many soon to follow. By 2020, AT&T plans to virtualize and control more than 75 percent of its network using this new software-defined architecture to meet the growing demands of data- and video-hungry users.

## Verizon

The amount of Verizon's traffic is exploding, generally driven by demand for video and cloud services. Contrary to popular belief, the network itself delivers a low ROI because carrier networks are built for peak usage, and as such are over-provisioned most of the time.

Verizon is taking NFV very seriously, seeing it as a way to build lower-cost network agility and flexibility without support staff for proprietary network functions. It is building a company-wide common OpenStack platform for running VNFs (Virtual Network Functions), as well as other internal applications. Production is around the corner.

Verizon counts on OpenStack because:

- It offers de facto implementation of a VIM (Virtual Infrastructure Manager)
- A critical mass of vendors are porting and developing applications (VNFs) targeted at OpenStack

<sup>16</sup> "Half of AT&T's networks are controlled by open-source SDN code", Richard Chirgwin writing for The Register, January 2016. [http://www.theregister.co.uk/2016/01/08/att\\_expanding\\_sdn/](http://www.theregister.co.uk/2016/01/08/att_expanding_sdn/).

- Integrators have developed the necessary deployment expertise using OpenStack
- OpenStack is a common environment that reduces vendor dependencies
- OpenStack components are being tuned to the needs of carriers, a trend that is essential to Verizon's ongoing efforts
- Fixes can be pushed upstream so patches do not have to be retrofitted again and again, so Verizon can focus on innovating.

Note that Verizon's test lab is built on Kilo, which is one release behind the current Liberty release. Verizon acknowledges all the improvements they can now take advantage of in the areas of high availability, SR-IOV and DPDK support, NUMA memory and scheduling, and SSD as cache.

## NTT Group

NTT Communications is the long-distance & international communications and ICT solution provider company of Nippon Telephone and Telegraph Corporation (NTT), headquartered in Japan and the third-largest communications company in the world by revenue. It desires NFV platforms that can federate NFV services between distributed heterogeneous sites: Carrier network, clouds and user sites. NTT Communications deployed a large proof of concept, built with the same architecture and topology as its commercial ISP backbone and cloud services.

NTT Communications plans to launch OpenStack-based NFV services with VNFs to their enterprise customers, including managed network functions like firewall and load-balancer. Customers can use the VNF vendors' features, for example rules, since vendor-native APIs will be exposed. Customers will be able to change their own network topology by attaching and detaching NTT Communications-managed VNFs.

They tested deployment, orchestration, interoperability to disparate distributed sites, monitoring of commercial VNFs: vFirewall, vRouter, vDPI, WAN Acceleration/Optimization, URL filter, and DDoS detection and mitigation.

They not only tested OpenStack as the Virtualized Infrastructure Manager, but also as the MANO components NFV Orchestrator and VNF Manager using Heat and early stage projects Tacker and Mistral. They also tested commercial MANO products. The conclusions on MANO are that the commercial solutions are difficult to use completely, and introduced vendor lock-in and flexibility issues. Conversely, OpenStack advantages are openness, flexibility and interoperability for VNFs.

NTT Group won the OpenStack Superuser Award in part due to its in-depth testing of NFV, identifying places for improvement, submitting user stories and blueprints, and contributing upstream to fill these gaps (see the table of requirements submitted and resolved by OpenStack telecoms in a prior section). These videos are available for more detail on NTT Communications' NFV PoC and other production implementations of OpenStack:

- NTT's Journey with OpenStack: [https://www.youtube.com/watch?v=Cu-MF8k7G\\_A](https://www.youtube.com/watch?v=Cu-MF8k7G_A)
- NTT Communications: NFV Service Federation Across Heterogeneous Sites: <https://youtu.be/IsqwpslERys>
- NTT Communications: Enhancement of OpenStack Networking for Carrier Cloud Platform: <https://www.youtube.com/watch?v=u1VKXU00LcE>
- Gohan: An Open Source Service Development Engine for SDN/NFV Orchestration: <https://www.youtube.com/watch?v=CEkhGUxD2oM>
- Delivering an End-to-End Automated and Carrier Class NFV (Network Functions Virtualization) Use Case: <https://www.youtube.com/watch?v=uwX27wsWPww>

## Deutsche Telekom

Deutsche Telekom (DT) is embracing OpenStack as the optimum platform for NFV. DT understands the need to avoid specialized hardware made obsolete by new paradigm services. NFV allows DT to deploy virtual network functions and scale them up and down quickly, without new investments in hardware.



Deutsche Telekom is well on its way to NFV in production. In March 2015, it announced its first production NFV workload running OpenStack, a cloud VPN service available in Croatia, Slovakia and Hungary. Deutsche Telekom supports OpenStack 100 percent, and contributes requirements and code to help advance NFV features more rapidly.

## SK Telecom

SK Telecom, the largest wireless carrier in South Korea, is moving toward Chief Technology Officer Dr. Alex Jinsung Choi's vision of the all-IT telecom infrastructure, which is to operate all of the telecom network functions on the cloud core in their software-defined data center. When complete, all components in the core networks in data centers and local network operation centers will be running as virtualized network functions in OpenStack cloud infrastructure.

Currently, SKT is focusing on virtualization of the traditional Telecom network functions such as IMS and EPC for elastic scale-out and control for service traffic explosion. These VNFs will be used for providing customer-specific, dedicated multi-tenant telecom services with orchestrated service chaining. The VNFs will also be used for enhancing service quality and reliability with elastic VNF resource management and load balancing control.

SKT's Network R&D Center succeeded in deploying parts of the operating IMS services as vIMS in its commercial operation environment with OpenStack and is currently operating them successfully. It will also commercialize more parts of the IMS services to vIMS soon, and put it into production. Next up is vEPC. The company's Network-IT Convergence R&D Center is establishing the foundational NFV infrastructure with OpenStack as an advanced VIM and NFVO with advanced VNF controls.

SK Telecom expects to realize these benefits from NFV using OpenStack:

- Reduce service downtime and cost, and improve network utilization by using automation

- Create more business opportunities with an agile, flexible, and programmable infrastructure
- Evolve more diversely, opening the company to various business and technology options.

## Bloomberg LP

Telecoms aren't the only organizations to benefit from NFV. The financial services industry is proving the relevance of NFV for large enterprises. Bloomberg has been running OpenStack in production for more than three years providing self-service and rapid delivery to more than 3,500 developers. Bloomberg is implementing SDN and NFV to allow programmatic networking and to lower costs.

OpenStack SDN and NFV capabilities allow every computer to participate in network scaling. Today, Bloomberg has implemented DNS in software, Firewall-as-a-Service and Load Balancing-as-a-Service, with much more to come. Deep Packet Inspection (DPI) hardware is one of many physical appliances that will be replaced and virtualized with NFV software. Others include Customer Premise Equipment (CPE), CDNs, WAN acceleration, Network Address Translation (NAT), Tester/Quality of Experience monitors, Provider Edge (PE) routers (these sit at regional levels, e.g. Japan), and more.

In addition to the benefits others have established, Bloomberg has a few unique observations:

- More servers running networking functions require management, but not specialists for each specific physical appliance.
- Automation further reduces the operator effort.
- In the financial markets industry, a specific agility advantage is that modern networking allows Bloomberg to make application decisions based on in-packet inspection with application protocols, such as FIX, a market data protocol.

# How To Get Involved

There are many ways to get involved, based on your needs and interest in contributing.

NEED	AUDIENCE	APPROACH
Strategic level involvement	Users or potential users	Join the User Committee list
Articles, interviews and news	Anyone	Read OpenStack Superuser publication <sup>18</sup>
Operational insight	Operators and admins	Join the Operators List
Specific vendor direction	Anyone	Engage with your vendors for roadmap, tested and verified offerings
Actual use case details	Architects	Review videos from the recent OpenStack summits
Networking, general discussions	Anyone	Join a nearby OpenStack User Group, attend meetups, or talk to your local OpenStack Ambassador <sup>19</sup>
Contributing to any OpenStack projects	Potential contributors and developers	How to Contribute wiki <sup>20</sup>
Technical NFV information	OpenStack enthusiasts and developers	#openstack-nfv IRC channel on irc.freenode.net.
Technical OPNFV information	OPNFV project members and interested parties	OPNFV OpenStack Community wiki <sup>21</sup>

17 <http://superuser.openstack.org/>

18 All found at <https://www.openstack.org/community/>.

19 [https://wiki.openstack.org/wiki/How\\_To\\_Contribute](https://wiki.openstack.org/wiki/How_To_Contribute)

20 <https://wiki.opnfv.org/community/openstack>



# Summary

As indicated by major telecommunications companies, ETSI NFV, OPNFV, and large enterprise network providers, OpenStack is the best fit infrastructure for NFV implementations. With support, requirements and community collaboration from all relevant sources, and its open source nature, ongoing rapid innovation for NFV users is guaranteed.

There are many ways to directly influence and contribute to OpenStack. Whether you choose to implement a solution based on the OPNFV framework, a vendor solution, or build it in-

house based on the ETSI NFV specification, OpenStack is at the heart of most offerings. OpenStack enjoys the support of more than 550 companies, many of which are incorporating OpenStack into their NFV solutions. Visit them in the OpenStack Marketplace at <https://www.openstack.org/marketplace/> or at <https://www.openstack.org/foundation/companies/>.



OpenStack is a registered trademark in the Unites States and in other countries.  
IHS is a registered trademark of IHS Inc. © 2016 IHS Inc. All rights reserved.  
All other company and product names may be trademarks of their respective owners.



*This work is licensed under the Creative Commons Attribution-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nd/4.0/>*