Virtualized network function (VNF) on-boarding describes the process by which a VNF is made available to a network functions virtualization (NFV) platform, allowing its life-cycle operations, such as deployment, scaling, healing, software upgrade and termination, to be automated.

As an essential and critical stage in the transformation toward NFV, on-boarding is a topic of strong interest for the industry, which is working through the ETSI standardization body and open source movements, such as OpenStack®, in advancing common standards, formats and methodologies.

Nokia’s CloudBand™ platform is a market-leading NFV platform, compliant with the ETSI model. Our CloudBand solution is based on OpenStack technology and is actively contributing back to the open source movement. The CloudBand platform allows for different models of VNFs, allowing for a rapid initial on-boarding and launch, and then a gradual transition toward the full life-cycle management of the VNFs.
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>About the NFV Insights Series</td>
<td>3</td>
</tr>
<tr>
<td>Introduction</td>
<td>3</td>
</tr>
<tr>
<td>Foundations</td>
<td>4</td>
</tr>
<tr>
<td>VNF on-boarding as a multi-dimensional iterative process</td>
<td>7</td>
</tr>
<tr>
<td>The on-boarding process</td>
<td>10</td>
</tr>
<tr>
<td>Service orchestration</td>
<td>11</td>
</tr>
<tr>
<td>The Nokia experience</td>
<td>11</td>
</tr>
<tr>
<td>Future evolution</td>
<td>12</td>
</tr>
<tr>
<td>Acronyms</td>
<td>13</td>
</tr>
<tr>
<td>References</td>
<td>13</td>
</tr>
</tbody>
</table>
About the NFV Insights Series

NFV represents a major shift in the telecommunications and networking industry. NFV applies virtualization and cloud principles to the telecommunications domain. Until recently, this approach appeared to be impossible due to the stringent performance, availability, reliability and security requirements in communication networks. Many service providers are now keen to implement NFV to help them gain an advantage through automation and responsiveness in order to deliver an enhanced customer experience while reducing operational costs. This series of whitepapers addresses some of the key technical and business challenges on the road to NFV.

Introduction

Virtualized network function (VNF) on-boarding describes the process by which a VNF is made available to a network functions virtualization (NFV) platform, allowing its life-cycle operations, such as deployment, scaling, healing, software upgrade and termination, to be automated.

On-boarding is a critical and potentially the most complex step in the overall move toward NFV, one that needs to be properly completed to assure that the service provider can take full advantage of the benefits of NFV while retaining and even improving the reliability and robustness that the traditional implementation was offering.

Figure 1. Moving network functions from traditional hardware to a shared NFV infrastructure
This document offers an overview of VNF on-boarding by first looking into the industry approach and the common best practices being used and then describing the Nokia approach and methodology for on-boarding VNFs on the Nokia CloudBand NFV platform.

Foundations

ETSI NFV Industry Specification Group

The ETSI NFV Industry Specification Group (ISG) has defined NFV Management and Orchestration (NFV-MANO) as the framework for the management and orchestration of all resources in the NFV Infrastructure (NFVI). This includes compute, networking, storage, and virtual machine (VM) resources. The VNF Manager (VNFM), a component of MANO, manages the life cycles of VNFs. In addition, the NFV Orchestrator (NFVO) manages the life cycles of network services; this is outside the scope of this document.

The on-boarding of network functions is the crucial first step in the overall life-cycle management. MANO also includes the VNF Catalog, a repository of VNF Descriptors created during on-boarding.

Figure 2. ETSI: NFV Management and Orchestration framework

Source: ETSI GS NFV-MAN 001, V1.1.1 (2014-12), Network Functions Virtualisation (NFV); Management and Orchestration, p. 23
NFV-MANO is broken up into three functional blocks:

- **NFV Orchestrator**: Handles the automatic management of network services life cycle and of global NFV Infrastructure resources across multiple data centers. In addition, the NFV Orchestrator manages multiple NFV infrastructure and coordinates resource requests.

- **VNF Manager**: Automates life-cycle management of VNF instances. The life cycle may consist of some or all of the following operations: deployment, scaling, healing, software upgrade and termination; Fault, Configuration, Accounting, Performance, and Security (FCAPS) management may continue to be done by the network function's element management system (EMS).

- **Virtualized Infrastructure Manager (VIM)**: Controls and manages the NFVI compute, storage, and network resources typically within one point of presence or data center.

For example, when a VNF is to be instantiated (or scaled), the VNFM consults the corresponding VNF Descriptor, to perform the following:

- Allocate and configure NFVI resources (compute, storage and network resources)

- Load or install the software components of the VNF; for example, using a repository of software images

- Set up virtualized network connectivity between the VNF components and to other network elements

- Connect the VNF to the operations support system (OSS) layer and manage and monitor it during its lifetime

**VNF Descriptor**

The VNF Descriptor is a text-based file that describes the VNF components, their requirements and policies used for life-cycle automation. In particular it describes the following:

- All the infrastructure resources required to instantiate the VNF: servers, floating IPs, volumes, security groups, networking configuration, etc.

- Any specific infrastructure resources and configuration options required to achieve the required performance on off-the-shelf servers; for example, network interface cards capable of Intel SR-IOV, CPU pinning, huge pages, Intel DPDK

- How the auto-scaling and other life-cycle operations should be automated: which parameters should be monitored; which criteria trigger a scaling; which software components should be scaled; how the additional resources should be connected to existing ones, etc.
How the platform should react in case of failure of a software component or an NFVI and how the failed component should be recreated and the correct reconfiguration performed at the network level (auto-healing) to avoid or minimize the service outage.

Various industry initiatives (ETSI NFV ISG, OpenStack, TOSCA, YANG, etc.) are working toward the standardization of a common VNF Descriptor format.

**OpenStack project**

OpenStack is the de facto software standard of the Open Source Initiative for NFV-MANO solutions. It has been adopted by most vendors to implement elements of the ETSI NFV architecture.

Several OpenStack modules are available for managing the life cycle of VNFs. The VNF on-boarding process needs to include the proper adaptations of the VNFs so as to ensure that their life cycles can be managed. The most important OpenStack modules used by the CloudBand VNFM are:

**Heat and HOT**

Within OpenStack, Heat is the orchestration service that allows instantiation and modification of VNFs, virtual networks, and other cloud resources in an automated fashion. Heat relies on a standardized VNF Descriptor called Heat Orchestration Template (HOT) to perform these tasks. CloudBand uses the HOT format as the VNF Descriptor.

**Mistral**

Mistral is the OpenStack workflow service. Most life-cycle operations consist of multiple distinct interconnected steps that need to be executed in a particular order in a distributed environment. Such workflows can be described as a set of tasks and task relations that are uploaded to Mistral so that it takes care of state management, correct execution order, parallelism, synchronization and high availability. Mistral also provides flexible task scheduling so that processes can run according to a specified schedule. Mistral is a key component that can be used to perform a complex flow of tasks required for processes such as scalability and healing etc.

**Murano**

The Murano Project introduces an application catalog to OpenStack, enabling application developers and cloud administrators to publish various cloud-ready applications in a browsable categorized catalog. Cloud users — including inexperienced ones — can then use the catalog to instantiate application environments with the push of a button.
VNF on-boarding as a multi-dimensional iterative process

While the ultimate goal of NFV is the full life-cycle automation of VNFs and their management by the NFV platform, Nokia uses a pragmatic step-by-step approach to on-boarding of VNFs. This multi-step methodology allows for an agile, fast, initial on-boarding of VNFs, while providing a path that will ultimately realize the full benefits of NFV.

The on-boarding of a VNF is also a multi-dimensional activity aiming to gradually carry out transformation in various areas such as networking, hardware, applications and their management, but also in the way the service provider is internally organized, in order to progressively take more advantage of the benefits offered by NFV (Figure 3). This means some VNFs are more advanced in one dimension than others.

**Figure 3. The path toward the full potential of NFV**

At an early phase of on-boarding, not everything will be automated; the VNF may still require manual actions for deployment, scaling, recovery from failures (healing), etc. The VNF, although running as a virtualized software function on top of an NFV infrastructure, may still assume that the underlying hardware is dedicated to it and not be shared with other applications.
The iterative on-boarding process allows a gradual advance toward the ultimate scenario of fully automated life-cycle management of the VNF. While there is no rigid path toward this progression, we can identify some typical major milestones:

- **Deployment**: At this stage, new instances of the VNF can be automatically created and deployed by the NFV Platform, which uses the information in the VNF Descriptor to fetch the software binaries from the VNF Catalog, and then instantiates and configures them over the selected NFV-Infrastructure.

- **Automated healing**: At this stage, the NFV platform can automatically handle the healing process; for example, the recovery of an application after failure of a hardware or software component. Typically, once an alarm is triggered either by the OSS or by the NFV MANO platform itself, a set of automatic actions are automatically executed to restore the application’s functions. These actions can consist of the re-creation of the failed software component on a different server, the modification of the networking configuration, and other actions to connect the newly re-created software with the other components.

- **Automated scaling**: As the load on a network function varies, the network function may require more resources (or fewer resources) to be able to function without being slowed down. Therefore the resources allocated to the VNF need to be increased (scaled up) or be reduced (scaled down). At this stage of integration, the VNF platform is capable of identifying when the resources allocated to a VNF are close to being exhausted (typical example is high CPU usage due to high demand). It will then automatically allocate additional resources, create additional instances of the software components that need to be scaled out, and most importantly reconfigure the network so that traffic is distributed over a larger pool of resources. A similar set of actions are also performed, when an application no longer requires the extra capacity allocated to it, and resources are automatically freed up by CloudBand.

- **Automatic full life-cycle management**: The ultimate objective of NFV is to allow not only for the instantiation and automated management of an existing release of a VNF, but also to automatically update (application of patches) and upgrade (roll-out of new releases) the VNF with minimum or no outages, thus relieving the operational team from these tedious tasks that are often executed outside normal working hours. The automation of life-cycle management also protects the system from potential human errors that can happen during these critical maneuvers.
Deployment and automation models

A key factor in the on-boarding process is the type of VNF manager to be used. ETSI NFV allows both VNF-specific, dedicated VNFM, which are part of the VNF software package and generic VNFM capable of managing any vendor’s VNFs.

In particular, depending on the entity supplying the VNFM, two distinct models of deployment and automation are possible (Figure 4).

• Dedicated VNFM: In this model, based on its knowledge of the product, the VNF vendor supplies its own dedicated VNF manager, which directly accesses OpenStack services and optionally CloudBand services such as automated placement. In this situation CloudBand works in a mode called “Co-Manage.” This mode allows CloudBand, serving as the NFVO, to learn about what the VNFM-D is doing and provide services such as monitoring and display of VNF status, root cause analysis and virtual-to-physical mapping.

• Generic VNFM: In this mode, a common, unified VNFM built into CloudBand and based on Heat/HOT, Mistral, and Murano is used for managing the VNF. This is an optimized mode of operation, because the use of the same VNFM allows the scaling of a common NFV Infrastructure with the most efficient use of the available compute, storage and networking resources. It also contributes to the advancement of a more open system approach for NFV where the VNFs and VNFM supplied by different vendors can work together through the use of standard communication interfaces.

Note that different levels of automation can be achieved in both the dedicated VNFM model and the generic VNFM model described above. Obviously, as the iterative work of achieving a higher level of automation advances, it will become more compelling to also move toward a unified VNFM model in order to benefit from all the synergies that such a model offers.
The on-boarding process

Traditionally, network functions were designed for and tested in a dedicated software and hardware environment where a fixed set of dedicated resources (compute, storage, network) were allocated to the network function and managed in a proprietary manner by the application itself or by its element or network management system.

This approach is no longer possible in a cloud environment where these resources are shared among network functions and managed by a MANO system. The process of modifying and virtualizing the network function, the creation of the VNF Descriptor and the loading of the Descriptor and software resources onto the NFV platform is what we call on-boarding. The following are key steps necessary for on-boarding a VNF.

Virtualization

If the network function is only available as a physical network function, then it first needs to be virtualized; that is, it needs to be able to run on an NFV infrastructure, typically within a set of VMs, virtual storage volumes and virtual networks. If sub-functions of the physical network function were executed in hardware, then these need to be re-implemented in software and all dependencies on the specific hardware need to be removed. This may be the most difficult step in the adaption for the on-boarding process.

Virtualization can be done initially as a simple one-to-one conversion where each hardware blade or hardware module is mapped to a corresponding VM with the same function. But it may also lead to a modification of the overall network function architecture to take advantage of the cloud capabilities of support for capacity management, high availability and other characteristics.

Other network functions may have been developed for virtual environments right away (native VNFs); for them this first step does not apply.

Creation of a description of the resources making up the VNF

VNFs will need a set of resources in addition to VMs: storage volumes, security groups, virtual networks, floating IP addresses and others. These resources can be described using a domain-specific language such as OpenStack HOT. This resource description is the first part of the VNF descriptor and it is key to automating the deployment of the VNF.

Upload of resources

Each VNF will use a set of software resources including executable software, VM images, data, configuration files, license keys, etc. These resources need to be made available to the NFV platform and be stored either directly in databases on the platform, such as OpenStack Glance for VM images, or in external repositories.
Description of workflows

For life-cycle operations such as scaling, healing, and software upgrade, policies and scripts or workflows need to be defined to be executed for the different life-cycle operations. For example, a scale-out process may include not only the creation of a new VM but also the registration of this VM with a load balancer, management system, etc. These policies and workflows are the second part of the VNF Descriptor.

Upload of VNF Descriptor

When the VNF Descriptor is completed, it is uploaded to the VNF Catalog of the NFV platform. Operators can then select any of the available VNF Descriptors and deploy an instance of the VNF with appropriate parameters/policies.

Test and validation

Before making a VNF available for live deployments, it needs to be tested and validated. The VNF Descriptor with its life-cycle workflow needs to be exercised and validated based on scripted test cases including failure (rainy day) scenarios.

Service orchestration

An on-boarded VNF does not mean that the service is ready for delivery to customers. Customer-facing and resource-facing services need to be created in a service orchestrator.

The service orchestrator needs to know how to configure the VNF to deliver the service desired by the customer. In case of any service-impacting failures, the VNF needs to notify the service orchestrator, which may attempt to deliver the service using other means, and will notify the business support system and the customer of any service degradation.

The Nokia experience

Starting in 2011, Nokia has been an early adopter of NFV, working to make the vision of NFV a reality. Central to this vision have been the CloudBand Ecosystem Program and the Cloud Innovation Center (CIC) working together to advance NFV.

- The CloudBand Ecosystem Program is a community of more than 60 member companies working together toward the advancement of NFV. It includes VNF suppliers, service providers and key technology providers such as Intel and RedHat.
• The CIC is the operational team in charge of working with third-party VNF suppliers to onboard their solutions on the CloudBand NFV Platform. The CIC also has the broader mission of integrating a broad range of technologies in order to design and implement the vision of an NFV-enabled service provider taking full benefits of the advantages offered by NFV.

With the help of these two organizations, Nokia has on-boarded on the CloudBand platform dozens of VNFs, not only its own in-house network functions but many more third-party network functions — including those from competing telecom equipment vendors. Based on these efforts, Nokia is arguably the NFV platform supplier with the largest on-boarding experience.

Nokia and the CloudBand team are committed to an open approach for NFV. CloudBand is adopting OpenStack software and other open source software whenever appropriate. To the extent possible, on-boarding a VNF on CloudBand utilizes generic OpenStack technology, such as HOT and Mistral, and such on-boarded VNFs can be easily ported to other NFV platforms.

In some cases, however, OpenStack out of the box is not sufficient to fulfill service provider requirements for NFV. OpenStack is a collection of many different modules with many configuration options. CloudBand integrates a selection of these modules and makes them into a production-grade NFV platform. This includes the addition of functionality that is missing or insufficient in the open source software base. Nokia contributes enhancements back to the community upstream to avoid creating proprietary solutions.

Future evolution

The industry continues to work on streamlining the application on-boarding process and many additional improvements are planned for the near future. Nokia supports open source technologies and contributes to these and other projects to foster industry convergence on a small set of interoperable (de facto) standards.

A simplified, standardized and automated on-boarding process is also a key ingredient of a DevOps approach to building network functions. As developers and operators work together to automate VNF on-boarding, a more iterative development and deployment process can be achieved and new features can be introduced and monetized more quickly.

The transformation to NFV is changing the networking industry and the fundamental way in which network functions are designed, developed and deployed. The industry is still at an early stage of this journey and many innovations are expected, many of which will aim at facilitating the collaboration between different suppliers in an ever more open ecosystem.
Acronyms

CIC  Cloud Innovation Center  
COTS  commercial off the shelf  
DPDK  Data Plane Development Kit  
EMS  Element management system  
FCAPS  Fault, Configuration, Accounting, Performance and Security  
HOT  Heat Orchestration Template  
NFV  network functions virtualization  
NFV-MANO  NFV Management and Orchestration  
NFVI  NFV Infrastructure  
NS  network services  
OSS  operations support system  
SR-IOV  Single Root I/O Virtualization  
VIM  Virtualized Infrastructure Manager  
VM  virtual machine  
VNF  virtualized network function  
VNFM  VNF Manager

References

ETSI GS NFV-MAN 001, V1.1.1 (2014-12), Network Functions Virtualisation (NFV); Management and Orchestration