HEAVY READING

White Paper

Cloud-Native NFV Architecture for Agile Service Creation & Scaling

Prepared by

Roz Roseboro
Senior Analyst, Heavy Reading
www.heavyreading.com

On behalf of

Mellanox Technologies

www.mellanox.com

January 2016

Introduction

While the initial focus of network functions virtualization (NFV) is often efficiency gains and cost savings, more mature communications service providers (CSPs) are motivated to leverage NFV for agile service creation and scaling, and ultimately boost the organization's top line and competitiveness. To achieve that goal, the virtualized network function (VNF) suppliers are working with CSPs to ensure that VNFs not only maintain their functionality and performance, but also be cloud-native to address agility, reliability and scalability at the application level. Most services can now operate in a virtualized environment on a commercial-off-the-shelf (COTS) server. The end state, however, is a fully automated cloud environment, as nearly all VNFs will ultimately be delivered from the cloud.

Today, most VNFs are not "cloud-native" and must be re-architected to support auto-provisioning, auto-scaling and auto-healing from error conditions in the cloud environment. This encompasses a number of different aspects. One is decomposing VNFs into microservices, which offers the opportunity to centralize common functions and remove unnecessary functionality, while also improving the application's overall agility, maintainability and resiliency through software modularization and greater distribution of functionality. Another potential gain is the possibility of decoupling state from the application transaction processing in order to facilitate scaling. Lastly, open application programming interfaces (APIs) and open source can help to ease integration and facilitate more rapid innovation in service creation.

VNF suppliers are actively working with their CSP customers to educate them on why the transformation to cloud-native VNFs is imperative for agile service creation and scaling, what that transformation entails, and its impact on the cloud infrastructure itself. The more advanced CSPs understand the necessity of re-architecting VNFs, while others remain cautious about moving toward cloud architectures, due to concerns about resiliency. With proper monitoring and analytics tools, however, CSPs should gain the confidence they need that the cloud's flexibility, agility and cost benefits outweigh any near-term concerns about performance and reliability.

This report is structured as follows:

- **Section 2** explains how the move to agile service creation and scaling necessitates changes to network functions' architectures, taking into consideration the diversity and unique characteristics of VNFs. It also highlights the growing awareness among CSPs of the need to re-architect VNFs and the role open APIs and open source play in NFV evolution.
- **Section 3** describes the steps that VNF suppliers are taking as they transition to the cloud. It details why decomposing VNFs into microservices is critical to help CSPs increase service agility, and discusses the tradeoffs and impacts that need to be considered.
- **Section 4** discusses the automation benefits that come from a cloud model, and the transition from vertical scaling to horizontal scaling. It highlights the role of the VNF Manager and cloud orchestration systems in facilitating auto-scaling, and the applicability of service statelessness for VNFs.
- **Section 5** presents recommendations for CSPs as they work with their VNF suppliers on moving to a cloud-based model for NFV.

Network Functions Need to Change

Moving Beyond Cost Optimization to Agile Service Creation & Scaling

CSPs are struggling to counter mounting pressures from over-the-top (OTT) providers. On one end, the price of basic voice and data services is declining, directly impacting CSPs' top line. On the other end, value-added services from OTT providers have captured most of the value from digital transformation. Growth in these services is driving network capacity expansion, increasing the CSPs' cost of operations and putting further pressure on their bottom line. On top of cost control, CSPs are also motivated to increase the pace of innovation, deliver and scale services faster, and improve infrastructure efficiency. This is driving their move to NFV and service delivery in the cloud.

The initial focus of NFV entails leveraging standard IT virtualization technology to consolidate many network equipment types and run network services as virtual machines (VMs) on industry-standard high-volume servers, switches and storage platforms in order to reduce the underlying cost base. These platforms tend to rely on merchant silicon rather than dedicated ASICs, which restricts the ability to accelerate media/packet processing as directly as had been done in the past. Also, the design of specialized chips such as digital signal processors (DSPs) is much different from COTS chips, forcing changes in the applications to accommodate those differences. Multi-threaded cores are becoming more commonplace, which helps to mitigate some of the potential impact. Most agree that the benefits of using a standardized platform in a more disaggregated model outweigh the performance concerns.

The more mature CSPs are starting to look beyond cost optimization and put more emphasis on service agility and scalability. The future of NFV must converge to a cloud-native architecture, so that VNF provisioning, scaling and error recovery can be handled in a fully automated fashion.

Agile Service Creation & Scaling Demands Cloud-Native NFV

A cloud-native architecture is usually of scale-out design, and involves tight teamwork among VNFs, NFV MANO, and NFV Infrastructure to deliver superior agility, scalability and resiliency. It entails open VNFs that have decomposed into microservices and decoupled states from transaction processing, NFV MANO that can intelligently monitor the state of VNFs and close the feedback loop for service resource adjustment, and efficient NFV infrastructure that can effectively support the cloud-native VNF and MANO layers.

The expectation is that CSPs will be able to mix and match microservices to create new types of services, and do so in a more automated and streamlined manner. The NFV MANO will play a key role in enabling this more agile service creation. It will also support auto-scaling of resources in a cloud environment.

In this new paradigm, resiliency and high availability need to be rethought to take into account that standardized server, switches and storage platforms are less reliable than dedicated hardware platforms. This has led to a change in focus from providing high availability at the hardware level to high availability at the service level. VNF suppliers are of differing opinions as to where the responsibility for achieving high service availability lies. Many say that resiliency needs to be designed into the applications, but others say the cloud management system should be responsible for it, not the application. The VNF Manager (VNF-M) monitors the health of VNFs, and re-instantiates a VM in case of failure, or takes action if degradation is occurring. Some VNFs have been designed to accommodate failures, especially those that traditionally relied upon many boxes to deliver. Others suggest that COTS platforms may not be as unreliable as some think, and in any case, most expect that platform reliability will improve over time.

With cloud, resources are not as constrained as with dedicated hardware. By pooling resources, they are – theoretically, at least – unlimited. This is in contrast to a hardware-based environment, where applications were designed with an understanding of the constraints of fixed resources. It also meant that it was expensive to keep resources available in active standby. In the cloud environment, it becomes cheaper/more efficient to have backup resources, so applications should account for this change.

Telco Functions Differ From IT Applications in Important Ways

Many telco applications are different from IT applications in that they have data plane/packet processing functions in addition to control, signaling and media processing functions. Each has different, and in many cases, more stringent performance, latency and capacity requirements than a typical IT application. CSPs are pushing their vendors on the performance front, but appear willing to accept a slight performance degradation because of the agility and flexibility benefits of the cloud.

Real-time applications are by definition latency-sensitive, making it imperative to get packets onto the network as quickly as possible, and be able to reconfigure network traffic under failure conditions. Data plane-intensive network functions, such as routing, are characterized by a need for both high capacity (how many sessions/how much traffic can be supported at one time) and high performance (how many packets/sessions/transactions can be processed per second). Control (e.g., Session Border Controller) and Signaling (e.g., PCRF, IMS) functions are less performance-sensitive, but are equally concerned with capacity and scalability. Many telco applications require maintaining state, which is a particular challenge in distributed architectures.

"Open" Is an Important Element in Re-architecting VNFs

CSPs are pushing their VNF suppliers to embrace a more open approach to software development. They have seen the cost and time-to-market advantages that Web-scale Internet players have achieved, and aim to achieve the same benefits.

"Open" in this context can refer to open APIs, which are key to addressing concerns around integration, which consistently ranks among CSPs' greatest challenges in the transition to cloud-based architectures. "Open" can also refer to "open source," which CSPs find valuable in that it can reduce costs, speed development, and accelerate innovation by harnessing the wisdom of the community. Most VNF suppliers are, indeed, participating in numerous open source initiatives and are beginning to leverage open source and introduce more open APIs in direct response to this emerging CSP requirement.

Growing Awareness Among CSPs of Need to Re-architect VNFs

CSPs' level of understanding differs, as each is at a different stage on their transformation to the cloud. The more sophisticated CSPs understand that most VNFs need to be re-architected to achieve true agility, scalability and optimal performance in the cloud. They see the opportunities a cloud model offers and are pushing their suppliers toward the cloud. They realize that they will not get all of the benefits of NFV without the automation and flexibility the cloud provides. Those who "get it" recognize the disruptive nature of the cloud, and see it as a critical component of addressing competitive challenges. They value the flexibility and choice that comes with disaggregating the applications from the underlying hardware. They expect the cloud will enable self-service models, and help generate new revenue streams.

Still, the move to the cloud is in the early stages. Most of the activity is in the proof-of-concept state, with Layer 4-7 appliances being the first VNFs being trialed. CSPs and their VNF suppliers are together learning how functions operate in a virtualized environment, and just beginning to see how they will operate in the cloud. The vendors are helping to educate their CSP customers about how I/O, CPU, memory and storage contribute to the performance of the VNFs – issues that previously had been the domain of the IT group.

Key Milestones to Cloud-Native NFV Architecture

Port to Virtualized High-Volume Server

For many network functions, the first step is to port the software from a dedicated platform to a virtualized high-volume server. To achieve this milestone, network function owners usually have to deal with two major challenges: the lack of special-purpose hardware assists, and performance penalties associated with virtualization. For data plane-intensive VNFs, efficient server I/O can potentially kill two birds with one stone: enhance packet processing performance, and enable the system to run more efficiently through CPU offload.

For more details, please refer to Using Hardware to Improve NFV Performance.

Decompose

Most network functions are deployed as a single monolith, whereby all components are bundled together into one entity. This can lead to maintenance challenges, as well as slow down trialing of new technologies. Decomposition is therefore a significant step in evolving VNFs to be cloud-native and much more agile and scalable. It will be undertaken for the majority of data plane, control and signaling VNFs. (Some connection-oriented VNFs do not benefit from being run in the cloud, since resources cannot be shared, so are unlikely to perform this step.)

In this context, we define decomposition as breaking a monolithic VNF into a set of collaborating services, often referred to as "microservices." Each service implements a set of related functionality, such as session management, subscriber billing, stats collection, etc. Services normally maintain their own states and communicate with each other through well-defined APIs, but each service can be developed and deployed independently. It's important to note that the goal of decomposition is not to have tiny services simply for the sake of it; instead, the goal is to address the problems and limitations of the monolithic architecture described above. Some services could very well be tiny, but others will be substantially larger.

Benefits & Challenges of Decomposing Monolithic VNFs to Microservices

First, microservices offer the ability to scale independently of one another. Moreover, each service can be deployed on hardware that is best suited to its resource requirements. This is quite different than when using a monolithic architecture, where components with wildly different resource requirements – e.g., CPU intensive vs. I/O intensive – must be deployed together. CSPs value the flexibility and efficiency of this approach, presuming it will help them maximize resource utilization and respond to changing market and business situations more quickly. Smaller, less complex elements may also be easier to integrate, provided that the interfaces between services are stable and standardized.

Second, the microservice architecture shrinks the failure domain and improves fault isolation, facilitating easy troubleshooting and high availability. For example, a memory leak in one service only affects that service; other services will continue to handle requests normally. In comparison, one misbehaving component of a monolithic architecture will bring down the entire system.

Third, decomposition also allows for common functions, such as load balancing and DPI, to be stripped away from the core logic of the applications. This allows the applications themselves to be more lightweight – which makes them easier and quicker to develop, manage, and deploy, perhaps as containers – as well as centralize core functions and operate in an "as-a-service" model. CSPs want to be able to reuse common functionality, but not have to pay for it multiple times. The process of decomposition also gives VNF owners the opportunity to remove unneeded functionality, such as support for obsolete protocols, from the application logic.

There are challenges associated with this approach, as it introduces significant operational complexity. There are many more moving parts – multiple instances of different types of service – that must be managed in production. To do this successfully, CSPs need a high level of automation to be designed into the NFV MANO layer. In addition, potential interoperability issues can arise when vendors are forced to interact with another vendor's product for key functionality.

Once VNFs are decomposed into microservices, CSPs would be able to mix and match them into new and different services. Layer 4-7 VNFs tend to be discrete functions already, so some Layer 4-7 systems vendors believe they may not be decomposed much more before moving to the cloud. In addition, some functions, such as IMS and SBC, are so specialized that some mobile VNF vendors argue there is little to no value in having their sub-functions available for reuse elsewhere. Overall, CSPs do value the increased flexibility and agility that will be possible with microservices for service creation.

Lightweight VNFs are more suitable to be deployed in a distributed architecture, whereby application functionality is deployed on multiple servers, which are managed as a cluster. Many, but not all, VNFs are able to operate in this manner today. CSPs are looking at distributed architectures to achieve elasticity, redundancy and resiliency.

VNF suppliers are working to find the right balance of decomposition and centralization. They need to ensure that they don't create more problems than are being solved. More specifically, they aim to:

- Minimize the number of times packets need to be touched or moved, as well as the number of calls to a central data store so that I/O doesn't dramatically increase.
- Create so many microservices that integration, orchestration and managing state become unmanageable and/or too costly, and East/West traffic overwhelms top-of-rack switches.
- Minimize the number of hops so as not to introduce latency. This takes on increased significance in the cloud, as control over latency can be lost as VMs can be automatically moved to different data centers.

Automation Comes With Cloud

Automation is the main enabler of agile service creation, delivery and scaling, and the key difference between basic virtualization and cloud-native NFV architecture. Many CSPs are pushing their suppliers to support this delivery model. VNFs must work with VNF managers and cloud orchestration systems to achieve auto-provisioning, auto-scaling and resiliency.

Provisioning

The VNF-M and cloud orchestration systems handle the primary responsibility for managing the lifecycle of the VNF. The VNF-M abstracts the details of the VNF's requirements that the orchestrator then uses to instantiate new VMs as needed. The VNF itself has to find out what the VNF-M has assigned, then self-provision the configuration. It also monitors the health of the VNF and can scale up or down as necessary.

Analytics tools make sense all of the data coming in from the network, and allows the orchestration system to make more intelligent choices in real time. It remains critical, though, that the CSPs have tools to monitor and verify the actions that are executed, especially in the early days while operations teams get accustomed to working in an automated environment.

Scaling

Automatic scaling of resources – also referred to as "elastic resources" – is one of the main benefits of adopting a cloud-native architecture. There are, however, two possible ways to increase capacity: scale up, whereby more resources are added to a given VM; and scale out, whereby more VMs are created to support a given function. As with most factors discussed in this paper, each has different considerations and tradeoffs, and the decision is not necessarily dependent on the VNF type.

With scale up, performance can be improved by the addition of more resources. However, the VNF architecture needs to accommodate the possibility of more cores being available for it to leverage. It also presents the possibility of reaching the physical capacity of the host.

With scale out, redundancy is improved by having functionality spread over multiple VMs. However, as will be discussed, managing state across these independent transaction processing VMs can present a challenge. Unless clustering is used, it could also present a management challenge and complicate service chaining by having more elements to consider.

While VNF suppliers and their CSP customers are still learning about the impacts and tradeoffs of each approach, scale out appears to be the preferred option for most VNFs. Those that are throughput-intensive are more likely to scale up, while those that are sessions/capacity-intensive look likely to scale out. And indeed, some may need to scale both up and out, such as EPC, which may need to be deployed in more locations, with each instance handling more sessions. Analytics tools can be used to help recommend an approach based on expected network behavior.

Resiliency

Improving resiliency is one of the main drivers for distributed architectures. Application functionality is spread across VMs residing on multiple servers, which are managed as a cluster. This approach eliminates a single point of failure, while also limiting the blast radius in the event of a failure. That is, rather than millions of users impacted by a failure, only some small portion of them are impacted.

VNFs must be architected differently for horizontal rather than vertical scale. The elasticity of the cloud demands ephemerality. Not only must we be able to create new application instances quickly; we must also be able to dispose of them quickly and safely. To achieve effective horizontal scaling, the principle of service statelessness is often used to decouple service processing from their state data whenever possible. Stateless applications can be quickly created and destroyed, as well as attached to and detached from external state managers, enhancing our ability to respond to changes in demand. Of course, this also requires the external state managers themselves to be scalable. Depending on the performance required to access states, the distributed fault-tolerant state store can be implemented as in-memory data grid, cache and persistent object stores on various media, such as flash/SSD or hard disks.

As mentioned earlier in this report, maintaining state is particularly important in the telecom domain, and presents one of the bigger challenges for scaling VNFs in the cloud. VNF suppliers are of differing opinions about the best way to manage state, but most agree it would be preferable to decouple state from the application transaction processing logic itself. Some of the thought leaders in the VNF space have already demonstrated telco applications architected to be stateless, and as a result, can scale almost linearly to satisfy any load requirements.

Once state and service processing are decoupled, there can be multiple ways to manage state information. The most common and intuitive way is to store and manage state in a database, where it can be accessed by any VM in the event of a failure or when the application needs to be scaled out or in. There are many distributed, fault-tolerant database solutions to choose from, and many of them are open source, such as Cassandra or HBase. Most of the platform-as-a-service (PaaS) offerings have database as a basic service included.

On the positive side, the VNFs themselves can be freed from the complexity of managing state and focus on other service processing logic, which can significantly shorten VNF development time. Some of these database solutions were initiated at Web-scale companies such as Facebook, so they are mature and proven to work in highly scalable environments. However, there may be complexities when information in databases associated with their corresponding microservices need to be synchronized. There is also the potential performance concern when a large number of packets result in cache miss and need to call back to a separate location in order to acquire state information. This challenge is prominent mostly for data plane applications that have to run at line rate, but can be mitigated by using one of the distributed memory object caching solutions such as Memcached.

Others say it is more efficient to replicate state into numerous VMs, any one of which could then take over for a failed VM. The benefit here is that failover times are drastically reduced, as a backup VM would have what it needs to operate. It also provides redundancy, which is useful for disaster recovery. However, keeping everything synchronized could present overhead and introduce an element of risk. Scaling VNFs in and out can also be challenging with this approach, and it requires a lot of intelligence in the load balancer module to direct traffic to the appropriate VM. The code to manage states is also tied to specific implementation and make code reuse almost impossible, slowing down new feature development.

What This Means for CSPs

CSPs should encourage their VNF suppliers to move beyond virtualization to decompose VNFs into microservices. This approach is appropriate for nearly all VNFs. After decomposition, VNFs will be more lightweight by eliminating unnecessary functions and centralizing common functions, giving CSPs more flexibility and agility, and the ability to create new types of services faster and at lower risk.

They should also advise their suppliers to carefully balance decomposition and complexity. The number of microservices will impact integration, orchestration and demand for networking resources within the data center. To handle the expected increase in East/West traffic generated by increased volumes of VMs, CSPs need to carefully plan their data center network capacity to accommodate the anticipated traffic volume. They should also consider the number of hops in a service chain, and minimizing latency at each hop to maintain the SLA of the services.

Managing state will remain a challenge for the foreseeable future, which may constrain how distributed and scalable certain functions can be. For CSPs that anticipate highly elastic demand on their VNFs, or intend to drive up infrastructure efficiency by time-multiplexing multiple VNFs onto the same infrastructure, they should encourage their VNF vendors to decouple states from service processing as much as possible, so that the service instances can easily scale in and out.

CSPs should leverage the ability to pool resources and scale resources up or down automatically to maximize cloud resource utilization. They should use analytics tools to model network behavior to better understand how to optimize VNF functions in the cloud, and recognize that different VNFs have different resource requirements that must be taken into account when scaling up and out, so one cannot use the same approach for all VNFs.

To achieve service statelessness and real-time insights from analytics, CSPs need to carefully design their cloud infrastructure, especially choosing the appropriate storage options that meet the requirement of state access and data usage.