NETWORK FUNCTIONS VIRTUALIZATION (NFV) with A TOUCH OF SDN

RAJENDRA CHAYAPATHI | SYED FARRUKH HASSAN | PARESH SHAH

Network Functions Virtualization (NFV) with a Touch of SDN

hardware devices and their capabilities. This network is rigid and cannot easily adapt to any future changes that may be required for introducing new services. Even when changes are made, it requires physical access and human labor. NFV-based design is meant to remove this restriction and provide flexibility that is not restricted by networking hardware. Additionally an NFV design could meet changing network requirements by incorporating elasticity, scalability, and software-centric approach. With NFV's ability to offer speedy transition and agility, the resulting network can avoid longer lead times which has been plaguing new service adoption in the traditional networks.

To utilize the capacities of NFV and reap its full benefits a different approach is required for the design and deployment for NFV networks. As emphasized earlier, the network functions are decoupled from the hardware, so the choice of VNF types and vendors doesn't have a correlation with the design of the physical infrastructure. Similarly, the physical infrastructure can be designed without influence from the VNFs it will host and run. Another design dimension is added by the considerations for management and deployment of the network functions. Each of these blocks can be designed individually and fairly independently since they are influenced by different factors and involve a different thought process. Figure 3-1 reflects these three dimensions of NFV design. The subsequent sections provide the details of these design categories.

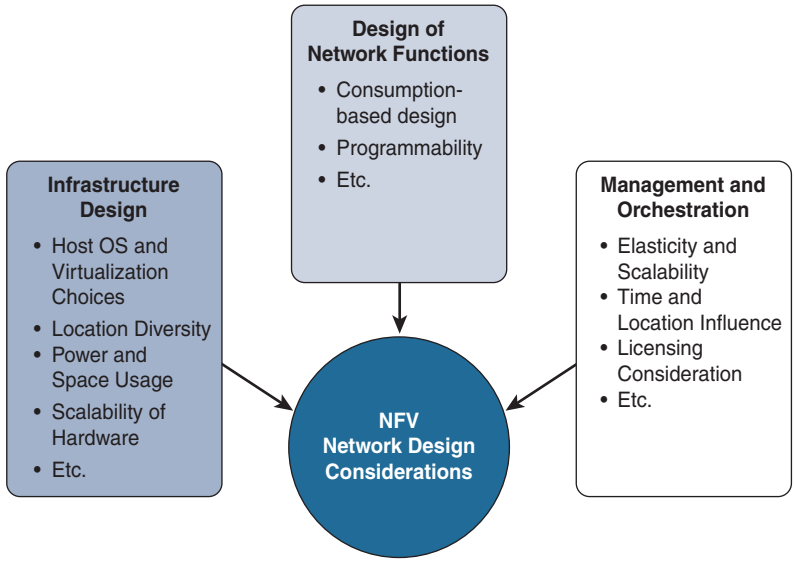

Figure 3-1 *NFV Network Design Considerations*

Despite the independence in the design of multiple blocks in NFV, the goal behind each layer's design should still have a common definition of the capabilities and performance that the resulting NFV network shall have. Any inefficiency or deviation from this goal at any one of the layers can create a bottleneck and lead to performance degradation for the entire network. Similarly, if one of the blocks is designed around higher performance network goals, that performance will not translate to the resulting network performance.

Designing NFV Infrastructure

The infrastructure for the NFV is not designed to meet the needs of a specific network and services. The infrastructure is meant to be generic and should ensure that it can allow for scalability and elasticity of the VNFs. It's also possible that this infrastructure is shared with server and data-center applications and not dedicated to NFV. The criteria for implementing a flexible and open platform for VNFs are discussed in the following subsections.

Scalable Hardware Resources

The infrastructure hardware should have flexibility to scale up and down if needed. Since the infrastructure is designed fairly independently of the overlying networking layer, it's not always possible to predict the hardware resource requirements that may arise. The primary approach to design around this is to pad the deployment resources as much as possible, as well as to design the resource pool so it can be shared across infrastructures. For example, use of shared disk pools instead of using servers that have built-in storage disks will result in reduced chances of wastage.

Even though operators want the initial deployment to have abundant hardware resources to avoid the need for future updates, it is still possible that the deployed hardware might prove insufficient for growing needs and require upgrades. To handle these situations, the operator should choose hardware equipment that can be easily scaled without impacting the current virtualized applications and VNFs that it may be hosting. This could mean that the servers chosen need to be capable of scaling up the hardware resources such as network interface cards (NIC) and memory.

Hardware Cost and Capital Expenses

The cost of the hardware is always an important selection criterion. Custom off the shelf (COTS) hardware is perceived to be the best way to achieve the optimal price point for hardware, but vendors like Cisco, HP, IBM, and Dell have been offering server products and pricing them competitively against COTS. Operators may be inclined to choose these commercially available servers, since the vendor built hardware would have been tested for any compatibility issues between components, and the servers are backed

by the support contract from the vendor. Essentially this choice is not much different from the choice an end user has to make between a custom-built personal computer using individual components, or a commercially built one from a vendor such as Dell, Lenevo, or HP. Whether the design results in choosing individual components separately, opting for COTS, or going with the vendor-built system, the choice impacts the overall capital expense for the deployment. This choice is also influenced by the expected reliability of the network and support available to resolve possible issues.

Choice of Host Operating System and Virtualization Layer

The host operating system (OS) and hypervisor must be compatible and integrate smoothly with the deployed hardware. Together they should offer a stable base to build the rest of the structure. When using COTS or a commercially available server, there is a wide range of choices for a host OS, hypervisor, and even orchestration tools. To narrow down these choices, consider the following:

- type of technical support available for these software pieces
- licensing costs
- procurement costs
- the roadmap for future support
- upgradability support
- stability
- ability to interact with open source and commercially available tools

Finding the right balance between all of these factors is a design decision. Some operators may prefer fully bundled software solutions from companies like VMware, RedHat, or Canonical. Others may find confidence in choosing from the other side of the spectrum for an open source, freely available OS like Ubuntu or CentOS running open source hypervisors such as a Kernel-based Virtual Machine (KVM). In the former case, the operator will incur licensing costs but will be comforted by the fact that the product has a proven track record, technical support structure, and has a secure future with a clear roadmap and upgrade path. In the latter case, however, they can eliminate the licensing costs and rely on in-house, third-party, or community-based support structures for future growth and issue resolution.

Efficiency in Power and Space Usage

Power and space requirements for the infrastructure hardware are reflected in the long-term operating expense of the network. This becomes much more critical in

parts of the world where real estate is hard to acquire and power tariffs are high. To get a perspective on how critical the space and power efficiency issues are, compare the deployment scale of the data centers being built today to host virtualized servers. These data centers are spread over many acres of land (or multiple floors of high-rises in densely populated locations) and consume hundreds of megawatts of power. Any improvements in the amount of space and power consumption for the individual servers can have a big impact in the operational cost of the NFV point of presence (PoP). It must be mentioned that the optimization in power consumption is not a direct result of virtualizing the network functions, but rather a result of utilizing the elasticity of the VNFs to scale on demand.

Common and Repetitive Footprint

The infrastructure can be designed in a way that the variability across different locations is minimal. Deployments can be simplified by designing for a common hardware and software footprint, which can be repeated across the NFV PoP. Achieving this simplification and the possibility of design replication requires that the power requirements, space needed, installation and commissioning expertise, provisioning tools, and methodologies stay unchanged. The common hardware infrastructure reduces the amount of redundant spare parts required to deal with possible replacement of failed hardware. On the other hand, creating a repetitive footprint might require some extra planning and care during the design phases.

Location Diversity

It is important to consider the choice of locations when designing the NFV infrastructure. Ideally the infrastructure deployment should be geographically diverse and more heavily deployed in possibly critical locations for traffic such as downtown areas. A metropolitan environment has more concentrated networking requirements compared to the suburbs.

One reason for diverse locations is to have redundancy against localized faults or disasters. However, another very important reason to diversify the locations is to ensure that the VNF has flexibility to be spun up where and when needed, without running into resource constraints. Later sections discuss the reasons behind requiring VNF placement based on geographical locations and the importance of availability of the infrastructure to make that flexibility possible when deploying VNFs. It is possible that a VNF may need to exist closer to the customer edge or that a particular location is expected to have an increased demand for VNFs during certain times or days.

Redundancy and High Availability

The design to mitigate failures in traditional networks is based on the assumption that one of the network functions can be lost if the device performing that function goes down, possibly due to even a single component failure. The redundancy in those traditional networks must be ensured at the device level to protect against possible network outages resulting perhaps from a single component failure. For instance if a single hard drive fails on a router, then it can affect the entire functionality of that router causing an network outage or traffic glitch. Depending on the level of importance of this device, a redundant device (or devices) or a backup traffic path is pre-provisioned and ready to carry traffic if needed.

In contrast, in NFV the high availability and redundancy is implemented per component; therefore the chances of loss of a network function due to a single component failure are highly minimized. For example, if a router is deployed as a VNF on a server using a Redundant Array of Independent Disks (RAID) technology, then the failure of one of the disks doesn't have any impact. Since an NFV infrastructure is shared, building redundancy in the infrastructure is cost effective, as multiple VNF are benefiting from this simultaneously.

In addition to the server hardware level redundancy, the infrastructure hardware design should also be able to offer redundancy to the virtual machines or container. Between the infrastructure switches, Spanning Tree Protocol and its variants like Rapid STP (RSTP), Per-VLAN STP (PVST), and Multiple STP (MSTP) have been used for a long time. More recent protocols that offer such redundancy are Transparent Interconnect of Lot of Links (TRILL) [1], Link Aggregation Control Protocol (LACP) [2], Multi-Chassis Link Aggregation (MC-LAG), and Ethernet VPN (EVPN). These and other similar protocols offer a number of choices and methods to provide redundancy for NFV Infrastructure hardware.

Redundancy and VM Mobility design and support offered by the virtualization layer should also be considered for a robust design. Some examples of this are VMware's VMotion and Openstack's Live-migration.

Infrastructure Life Cycle

The hardware devices used to form the infrastructure must be refreshed over a period of time. As shown in Figure 3-2, the life cycle starts with planning and procurement, and the hardware is discarded after their estimated life has passed. This life cycle duration of the hardware is based on the average amount of time the pieces of hardware are expected to operate without failures, as well as the duration of support contract and repair component availability.

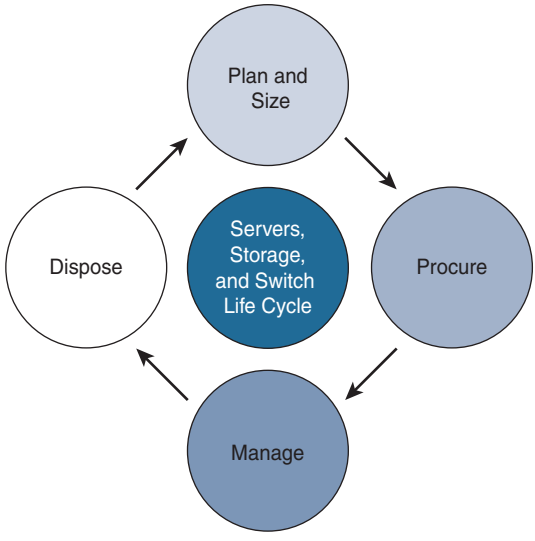

Figure 3-2 *NFV Infrastructure Life Cycle*

NFV Infrastructure (NFVI) design considerations should include the life cycle of the servers, storage, and switches used in the infrastructure. The servers and storage devices used in typical datacenter generally have a useful life cycle of three to five years [3]. Switches are considered to have a slightly longer shelf life spanning approximately six years. In this time frame, these devices have paid off the investment made on procuring them. To keep the odds of failure to a minimum, these devices are swapped out with newer ones once this life span has passed. The same time estimates and practice are applicable to NFVI as well. Additionally, the host OS, hypervisor, and VNF have their own life spans after which they will need to be upgraded, either for enhancements, support renewal, or bug fixes.

The design should therefore consider the coordination of these multiple factors to avoid possible pitfalls. For example, the software support and release cycle for the VNF may be one year, the hypervisor may have a suggested refresh time of two years, and switches and servers may have a six and three years life span respectively. The life span in this example do not align well with each other, and if proper design and planning is not done to handle this in the most optimal way, it can result in a constant network churn with upgrades. The design goal should be to minimize the impact due to upgrades, as well as to plan ahead to mitigate the possibilities of post-upgrade issues. These issues can be minimized through proper preintegration production testing.