Towards a Network Operating System

V. López[1], O. González de Dios[1], B. Fuentes[1], M. Yannuzzi[2], J.P. Fernández-Palacios[1], and D. López[1]

[1] *Telefónica I+D c/ Don Ramón de la Cruz, 82-84, 28006 Spain (email: vlopez@tid.es)*

[2] *Networking and Information Technology Lab (NetIT Lab), Technical University of Catalonia (UPC), Spain*

Abstract: A Network Operating System (NetOS) is a novel paradigm for developing a next-generation network management and operation platform. As we shall describe, NetOS not only goes far beyond the SDN concepts but also constitutes a fundamental enabler for NFV.

OCIS codes: (060.0060) Fiber optics and optical communications; (060.4250) Networks.

1. Introduction

The convergence of networking and IT is an established trend in today's ICT landscape. This trend has been recently reinforced by the virtualisation of IT infrastructures through cloud technologies, the evolution of network services by means of Network Functions Virtualisation (NFV) and the decoupling of network control and data planes proposed by Software Defined Network (SDN), with OpenFlow [1] as flagship.

While the cloud paradigm has had a profound impact on the IT and software market, the NFV and the SDN approaches are called to change the game and landscape of the networking industry. An analysis on how networks are run today is sufficient to realize that, in order to exploit virtualization and software innovations in a truly effective manner, the networking community must first undergo a profound transformation and really begin to think as "software people".

NFV and SDN bring a promising solution to operators, providers and users for reducing the complexity and costs of deploying and managing their heterogeneous networks and services. However, network-wide abstractions that will lay the foundation for true network programmability are still missing. Inspired by the success of Operating Systems, we encourage the research and industrial communities to design, build and deliver a running and stable Network Operating System (NetOS).

To motivate why a NetOS is required, let's consider the following example. Imagine for a moment a world in which for basic operations such as configuring, controlling, using, and managing any device in a computer the user must have specialized knowledge, and understand the internals of each item. For instance, imagine that for plugging and configuring a new device through an USB interface, the user must know the low-level instructions needed to plug, to install, and to use the device. Imagine a world in which the user will need to open a terminal, and then type dozens of commands just to burn a CD or to print a single page. Fortunately, IT has developed plug and play solutions that hide the underlying complexity, but these examples look very familiar to those dealing with network infrastructures. Indeed, the networking community knows them very well. It is just another day in the life of a network administrator.

The results of the lack of a "software mentality" by the networking community are obvious, and have led to highly fragmented administrative competencies. In a nutshell, segmentation by itself does not remove the high complexity that is present within each segment, and hence this strategy has derived in: i) higher OPEX since each network segment requires personnel with high specialization; ii) slow time-to-market given the inherent complexity of building and deploying services across technologies (e.g., across Access, Metro, Core, etc.); iii) poor coordination and resource utilization across technologies; iv) resistance to the integration of new technologies—often leading to further segmentation; and v) the proliferation of proprietary solutions at all levels in a market dominated by the "lock-in strategy". The solution to overcome these problems lies with the software community. They have done this for decades, and not surprisingly, with remarkable efficiency and success in the IT industry.

This paper presents NetOS as the main technology enabler for the adoption of IT best practices in the networking domain. The NetOS reference model is described in Section 2. Section 3 and Section 4 presents NetOS as an enabler of two emerging network technologies: NFV and programmable optical networks. Finally, Section 5 presents the conclusions of the paper.

2. The NetOS Reference Model

As an initial effort toward a NetOS, the model that we devise is composed of three main components: i) drivers and devices; ii) the NetOS kernel; and iii) the User space. Figure 1 depicts the proposed reference model with its main components and building blocks, and Figure 2 presents a general architecture of a UNIX operating system[1] for

[1] Adapted from Operating Systems Concepts, Silberschatz et al., Wiley Ed.

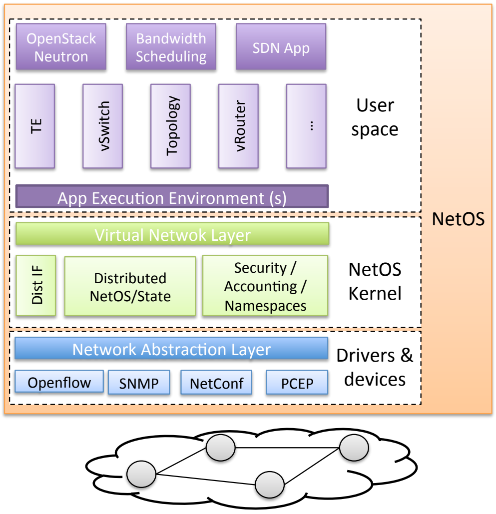

Fig. 1. Proposed layered structure of NetOS.

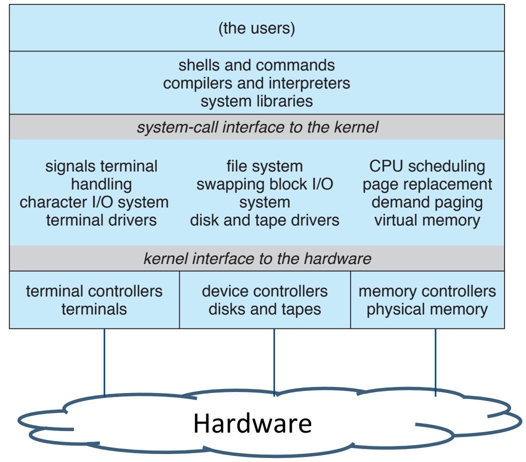

Fig. 2. UNIX System [1]

comparison. In the drivers and devices component, the Network Abstraction Layer (NAL) is the element responsible of providing a unified Southbound Interface towards the physical infrastructure. In other words, the NAL exposes the different Network Elements (NEs) to the NetOS kernel using drivers, thus enabling access to each specific element in a straightforward way—the drivers hide the inherent complexity of the NEs to the upper layers.

The NetOS kernel deals with the essential coordination functions, including common models for network resource virtualization, while providing uniform mechanisms in key aspects, such as security, tenant separation, namespaces and location, and resource lifetime management. Moreover, the NetOS kernel maintains the state of the network by interfacing with the network elements through the NAL. Finally, the Virtual Network Layer enables the deployment, control, and management of virtual networks over many possible underlying network technologies.

The User space must facilitate the integration with OSS (Operations Support Systems) and controller frameworks, such as OpenDaylight, but NetOS goes significantly beyond these. For instance: a) through the integration of multiple different Southbound interfaces; b) through the definition of layered interfaces between the NetOS kernel, libraries and application execution environments; and c) through the NetOS orchestration and NFV management services.

3. NetOS in the scope of Network Functions Virtualization

The virtualization of network functions aims at increasing the flexibility for launching and running new network services, while dramatically reducing the cost for network operators when offering such services. Network Functions Virtualization (NFV) is an activity founded by leading network operators around the world, and it is hosted at ETSI in the form of an ISG (Industry Specification Group). In contrast to current appliance-based models, which require the deployment of physical boxes—which entail additional demands in terms of physical gear, including complex deployment plans, and longer innovation cycles—NFV allows network functions to be provided as software components residing on commodity hardware.

The interplay between NetOS and the ETSI NFV architecture is shown in Fig. 3. The services currently under consideration at the ISG that will be run on NFV-enabled platforms cover practically all network segments and technologies. NetOS can make these services and their respective network functions accessible to programmers by providing suitable libraries and abstractions, thus enabling the utilization of NFV as a general approach to build new network services. In contrast with the appliance-based model, virtual network functions are not static, so they can be moved dynamically, and can scale out on demand to rapidly adapt to changing conditions and load characteristics. This allows for a richer programming environment for NetOS. In addition, the management and orchestration of such services, and their integration with non-virtualized services is greatly simplified by the application of the NetOS abstractions.

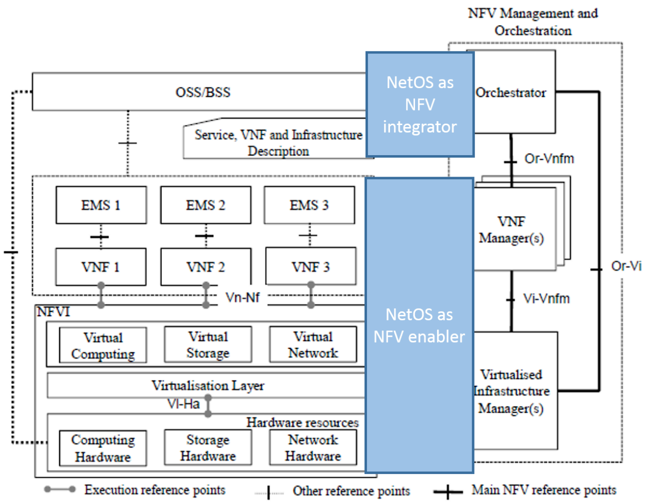

Fig. 3. NetOS.in the scope of ETSI NFV reference architectural model [2]

4. Southbound interfaces for Optical Networks

In order to effectively support cloud services, multiple distributed high-performance datacenters are interconnected by high-bandwidth dynamic optical networks. In such a cloud environment, optical network virtualization [3] plays a key role in interconnecting geographically distributed virtual IT resources (i.e. computing and storage) with high-capacity virtual optical network connectivity [4]. On the other hand, important research activities have work towards empowering optical networks with more elastic, flexible and programmable optical devices. For many years the research community has devoted significant efforts to design tools that could "automatically understand" how to configure and operate a variety of hardware appliances and "Network Elements" (NEs). There have been multiple attempts to enable a unified view of the network elements to facilitate the configuration processes. From Element Management Systems (EMSs) with their specific interfaces to the devices, up to the control plane, where the most important functions are done by the network elements. SDN is based on the idea of the separation between the control and the data planes. The controller is in charge of the control plane, while the forwarding elements are responsible for the functions in the data plane. SDN requires an open protocol between controllers and forwarding elements, allowing for a free combination of elements from different vendors to provide network functions, and of an open interface to the control plane, so the controller can be uniformly accessed by other components participating in the network (e.g., sources of network intelligence or applications in general). However, most of current implementations of SDN for optical elements are either based on vendor-specific extensions or on an abstracted view of the optical layer, where there is no place for resource optimization or network planning processes.

The IT and software communities have been much smarter and far more practical, with the adoption of the driver concept. However, developing a driver for a "Network Element" (NE) or pools of interconnected NEs, such as optical switches, routers, or hybrid IT & networking fabrics, is something that no one has done before. To this end, a NetOS requires defining the information models, protocols and semantics in the Southbound interfaces for the plug and play of NEs.

5. Conclusions

Network operators require an environment that enables fast and easy adaptation to customer needs, reducing the time for building and deploying services. There are ongoing efforts to achieve these goals, through technologies such as SDN and NFV, but NetOS goes far beyond these paradigms, and promises to enable an environment where network elements can be exposed to control and management applications and services in a much simpler way.

References

[1] Open Networking Foundation: http://www.opennetworkingn.org. Last accessed: 2013-November-21

[2] Network Functions Virtualization (NFV): Architectural Framework. ETSI GS NFV 002 v1.1.1 (2013-10)

[3] ADVA Optical Networking white paper, M. Ritter, "Virtualized Optical Networks for Sustainable Cloud Services," Jan. 2010.

[4] Shuping Peng, Reza Nejabati, and Dimitra Simeonidou, "Impairment-aware Optical Network Virtualization in Single-Line-Rate and Mixed-Line-Rate WDM Networks," Journal of OpticalCommunications and Networking, Vol. 5, No. 3, March 2013.