Management and Orchestration
of Virtualized Network Functions

Elisa Maini[1] and Antonio Manzalini[2]

[1] University of Naples Federico II, Via Claudio 21, 80125 Naples, Italy
[2] Telecom Italia Strategy - Future Centre, Via Reiss Romoli 274, 10148 Turin, Italy

Abstract. Software Defined Networking (SDN) and Network Function Virtualization (NFV) paradigms are driving a number of research activities aiming to develop virtual network infrastructures where Network Functions (NF) and services could be executed as applications in ensemble of virtual machines. This paper addresses the problem of managing and orchestrating said highly dynamic networks, where virtualized NFs and resources are created and destroyed depending on traffic demands, service requests, or other high-level governance goals such as the reduction of energy consumption.

1   Introduction

In the current networks most NFs (from layer 4 to layer 7) are provided by a number of middle-boxes. These nodes are based on closed software solutions running on specialized hardware representing a significant part of the Operational EXpenditures (OPEX) and CAPital EXpenditures (CAPEX) of network operators due to the management effort (e.g. for configuration) that they require [1], [2], [3], [4]. In principle, SDN and NFV are likely to offer the possibility to develop all NFs, e.g., middle-boxes, in open source software executed in standard hardware. If on one side, Future Networks will rely more and more on such software, which will, in turn, accelerate the pace of innovation and will reduce costs, on the other side, management and orchestration will be the true challenge. Automated operation processes (e.g. configuration of network and service equipment) could limit human intervention also wrong operations and a flexible and optimal provisioning of NFs and services could reduce equipment costs and allows postponing network investments. In this direction, SDN and NFV should be equipped with solid management and orchestration features, capable of automating networks and services operations. These paradigms will largely exploit virtualization technologies for sharing and aggregating resources, and for decoupling and isolating virtual network elements from the physical network. Further, the abstraction of virtualized network elements will enable and simplify the programmability of the network, increasing networking capabilities, allowing innovative service offerings: nevertheless this will increase also the overall management complexity of virtual NFs and virtual resources [7], [8]. In the following paper, we give an overview of the main research and challenges for the management and orchestration of virtualized NFs and, we provide an architectural

solution for the placement of the virtual resources. As proof of these concepts, some preliminary results are provided.

2   Concepts and Technical Challenges

Each Virtual Network (VN) is viewed as a *managed network service* by the management software [5]. All managed network services are mapped to the available resources which allows users of the network service to access it, just like they would access a physical network service. For full operation of the managed network services, the management and orchestration needs to ensure that each VN has the following attributes:

− *separation*: network functions and the network infrastructure are kept separate. A VN of a service needs to be separated from the infrastructure used by an infrastructure provider, similarly to the manner in which a virtual compute environment is separated from the physical host;
− *isolation*: a VN of one service is isolated from all other VN services. Isolated VN services need to be offered side by side while sharing network resources of the infrastructure provider;
− *elasticity*: a VN can grow or shrink as necessary. A VN service needs to enable network elasticity;
− *federation*: a VN can span over more than one domain. An interchangeable VN service needs to be offered across local areas such that network service consumers would not be concerned by the area of the infrastructure location, the physical network used, or its configuration.

Based on the previous description, the management and orchestration component should:

a) *use adaptive and autonomic methods and systems* for automatic management NFs applied not only to the physical resources, but also virtual resources located inside the physical/virtual resources;
b) *optimize the allocation and orchestration of virtual resources* for dynamically instantiating, orchestrating and migrating multiple virtual machines across networks and service infrastructures.

To fulfill the latter, the next section provide a high-level description of a distributed orchestrator prototype.

3   Architecture and Experimental Results

The architecture depicted in Figure 1, is based on an orchestrator that ensures the *automatic placement of the virtual routers and the allocation of the network services* on them, supported by a monitoring system which collect and reports on behavior of the resources involved. The orchestrator manages the creation
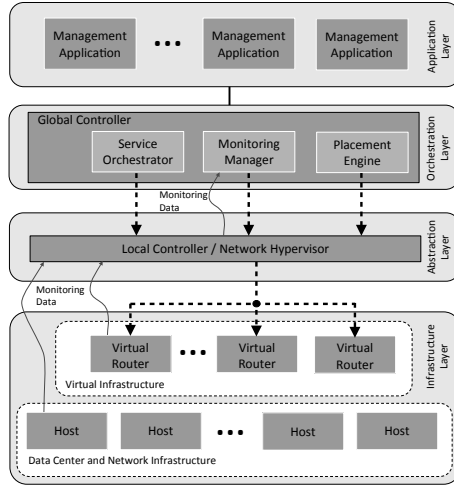
Fig. 1. Architecture overview

and removal of the virtual nodes, as well as configuring, monitoring, running and stopping software on them.

From an architectural viewpoint, NFs and services can be defined as a number of software components with their accompanying context together with configuration parameters. The provisioning of a service involves the creation of a IT infrastructure, followed by the installation of all necessary software components into the infrastructure, and finally configuring and starting those components. With SDN and NFV these processes can be simplified as the infrastructure provides a platform from which virtual machines can be run. SDNs can be directly manifested as virtual network topologies which need to be setup, have a managed lifecycle, and need to be shutdown - all under software control.
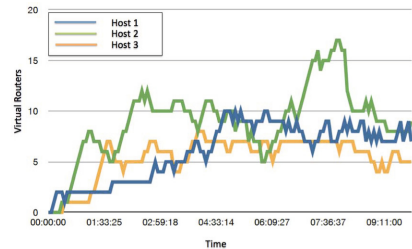
The architecture involves a *Global Controller*, in other words a distributed management infrastructure that has centralized functionality and it is responsible for the setup, configuration, optimization, and shutdown of the network entities. It takes input from various *Management Applications* regarding various requirements (e.g. network resources or response time) and then configures the network nodes through a set of *Local Controllers*. In order to manage the challenging and dynamic infrastructures of virtual networks there needs to be a monitoring system (*Monitoring Manager*) which can collect and report on the behavior of both the physical resources (e.g. cpu usage, memory usage) and the virtual resources (e.g. utilization level of the virtual links). These monitoring data items are sent to the Global Controller so that it can use the monitoring information in order to make decisions regarding network strategies. In particular, it may decide to add new nodes in order to fulfil the high-level policies and goals which are network operator requirements. In this way, the virtual network topology changes dynamically according to the network virtual resources usage.

The *Placement Engine* is the component in charge of performing the actual placement of the virtual routers according to the initial topology and the usage of the virtual network elements. This is an important feature because, when we configure a network, considering some initial information, some of these parameters may change during the course of the system's operation and a reconfiguration may be required to maintain optimized collection of information. The decision of the Placement Engine is encoded in an algorithm which can be rather simple, such as counting the number of virtual routers on a host, or it can be based on a set of constraints (e.g. the usage of the virtual links) and policies that represent the network properties. Also, the Placement Engine may be able to utilize or share some algorithmic elements from computer cloud placement algorithms such as [9] and [10].

Figure 2 provides some experimental results achieved using the Very Lighweight Network and Service Platform (VLSP) testbed [6]. It shows the number of virtual routers allocated on each host (shown on the *y-axis*) versus the time of the experimental run (shown on the *x-axis*). The hosts: host 1, host 2 and host 3 are represented by blue, green and yellow lines respectively. Here, we present two different placement algorithms: the algorithm used in Figure 2(a) is a kind of load balancing algorithm as it tries to get a similar number of routers on each host. The algorithm used in Figure 2(b) tries to determine the host that is *least busy* in terms of virtual network traffic. More details are explained in [11].
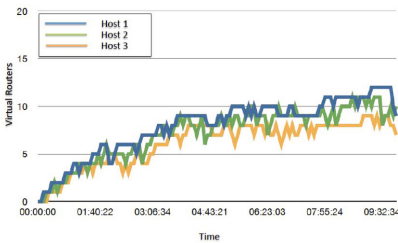
(a) Placement Engine: Least Used   (b) Placement Engine: Least Busy

Fig. 2. Behaviour of different placement algorithms

4   Conclusion and Prespectives

In this paper, we have argued the main research directions for the management and orchestration of virtualized NFs and services in Future Networks and provided an architectural description of our orchestrator prototype. The results presented here demonstrate that different embedded algorithms in each of the Placement Engines give very different placement strategies for the virtual routers. In future work will be considered other placement algortitms, which we expect to get more complex and to encapsulate multiple metrics.

References

1. Manzalini, A., Minerva, R., Dekel, E., Tock, Y., Kaemfer, E., Tavernier, W., Casier, K., Verbrugge, S., Colle, D., Collegati, F., Campi, A., Cerroni, W., Vilalta, R., Munoz, R., Casellas, R., Martinez, R., Mazzocca, N., Maini, E.: Manifesto of Edge ICT Fabric. In: Proceedings of 17th International Conference on Intelligence in Next Generation Networks (ICIN), Venice, Italy, October 15-16 (2013)
2. Manzalini, A., Saracco, R.: Software Networks at the Edge: a shift of paradigm. In: IEEE Software Defined Networks for Future Networks and Services (SDN4FNS), Trento, Italy, November 11-13 (2013)
3. Manzalini, A., Minerva, R., Callegati, F., Cerroni, W.: Clouds of Virtual Machines at the Edge. IEEE Com. Mag. Future Carriers Networks (July 2013)
4. Manzalini, A., Deussen, P.H., Nechifor, S., et al.: Self-optimized Cognitive Network of Networks. Oxford Journals The Computer Journal 54(2), 189–196 (2010)
5. Clayman, S., Clegg, R., Mamatas, L., Pavlou, G., Galis, A.: Monitoring Aggregation and Filtering for Efficient Management of Virtual Networks. In: 7th International Conference on Network and Service Management (CNSM), Paris (October 2011)
6. Clayman, S.: User Space Routing, Open Source Software, http://clayfour.ee.ucl.ac.uk/usr/
7. Galis, A., Clayman, S., Mamatas, L., Rubio-Loyola, J., Manzalini, A., Kuklinski, S., Serrat, J., Zahariadis, T.: Softwarization of Future Networks and Services - Programmable Enabled Networks as Next Generation Software Defined Networks. In: IEEE Software Defined Networks for Future Networks and Services (SDN4FNS), Trento, Italy, November 11-13 (2013)
8. Galis, A., Rubio-Loyola, J., Clayman, S., Mamatas, L., Kukliński, S., Serrat, J., Zahariadis, T.: Software Enabled Future Internet - Challenges in Orchestrating the Future Internet. In: Pesch, D., Timm-Giel, A., Calvo, R.A., Wenning, B.-L., Pentikousis, K. (eds.) Monami. LNICST, vol. 125, pp. 228–244. Springer, Heidelberg (2013)
9. Breitgand, D., Epstein, A.: SLA-aware Placement of Multi-Virtual Machine Elastic Services in Compute Clouds. In: IFIP/IEEE International Symposium on Integrated Network Management (IM 2011), Dublin, Ireland, May 23-27 (2011)
10. Breitgand, D., Epstein, A.: Improving Consolidation of Virtual Machines with Risk-aware Bandwidth Oversubscription in Compute Clouds. In: ACM/IEEE INFOCOM 2012, Track II (2012)
11. Clayman, S., Maini, E., Galis, A., Manzalini, A., Mazzocca, N.: The Dynamical Placement of Virtual Network Functions. In: 1st IEEE/IFIP International Workshop on SDN Management and Orchestration, SDNMO 2014 (to appear, 2014)