MEF White Paper

Simplifying NFV Operations in the WAN with LSO and Lean NFV

June 2020

Disclaimer

The information in this publication is freely available for reproduction and use by any recipient and is believed to be accurate as of its publication date. Such information is subject to change without notice and MEF Forum (MEF) is not responsible for any errors. MEF does not assume responsibility to update or correct any information in this publication. No representation or warranty, expressed or implied, is made by MEF concerning the completeness, accuracy, or applicability of any information contained herein and no liability of any kind shall be assumed by MEF as a result of reliance upon such information.

The information contained herein is intended to be used without modification by the recipient or user of this document. MEF is not responsible or liable for any modifications to this document made by any other party.

The receipt or any use of this document or its contents does not in any way create, by implication or otherwise:

a) any express or implied license or right to or under any patent, copyright, trademark or trade secret rights held or claimed by any MEF member which are or may be associated with the ideas, techniques, concepts or expressions contained herein; nor

b) any warranty or representation that any MEF members will announce any product(s) and/or service(s) related thereto, or if such announcements are made, that such announced product(s) and/or service(s) embody any or all of the ideas, technologies, or concepts contained herein; nor

c) any form of relationship between any MEF member and the recipient or user of this document.

Implementation or use of specific MEF standards, specifications, or recommendations will be voluntary, and no Member shall be obliged to implement them by virtue of participation in MEF Forum. MEF is a non-profit international organization to enable the development and worldwide adoption of agile, assured and orchestrated network services. MEF does not, expressly or otherwise, endorse or promote any specific products or services.

Table of Contents

List of Figures

1 Abstract

This document is aimed at Information and Communications Technology Service Providers (ICT-SPs) that are experiencing challenges scaling adoption of NFV in their WAN infrastructure. The White Paper describes the current complexity of integrating different components of NFV and the negative impact that has when introducing VNF and CNF solutions into the WAN to enhance service offerings for customers. The authors propose reducing this integration complexity by applying Lean NFV within the framework of MEF's LSO (Lifecycle Service Orchestration) through standardization of integration points between ICT-SPs and NFV vendors.

2 Introduction

In the last 15 years, the rise of "digital-first" players surely accelerated the overall TMT (Technology, Media and Telecom)[1] revenue pool, but the end result was that Telecom paid for the data growth, while "Silicon Valley" Technology captured most of the value.

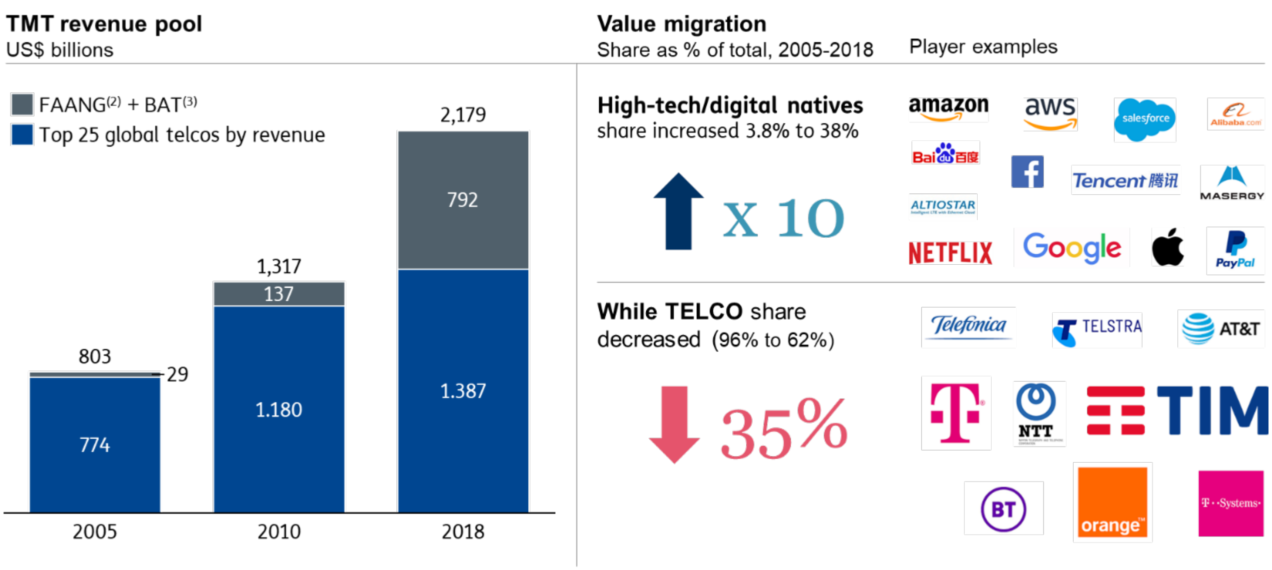

Figure 1 – TMT revenue pool vs. Value migration - Source: Capital IQ; McKinsey Analysis

[1] Tech, Media, Telecom
[2] Facebook, Apple, Amazon, Netflix, Alphabet (formerly Google)

For traditional telecoms the disruption enabled by the digitalization is an extraordinary challenge. Their latest round of investment was mostly unreturned, economic profit declined, margins deteriorated, and TSR (Total Shareholder Return) suffered. This was especially true compared with adjacent sectors such as media and technology, which maintained more consistent profitability and higher returns on smaller asset bases.

This was true both in North America, where between 2007 and 2018 telecom revenues increased 20%, but 28% less than the overall market, and even worst in Europe, with a 24% decline in telecom revenues over the same period, compared with an overall market growth of 18%.
And even the efficiencies achieved through the implementation of new operating models were not enough to compensate for declining revenues.

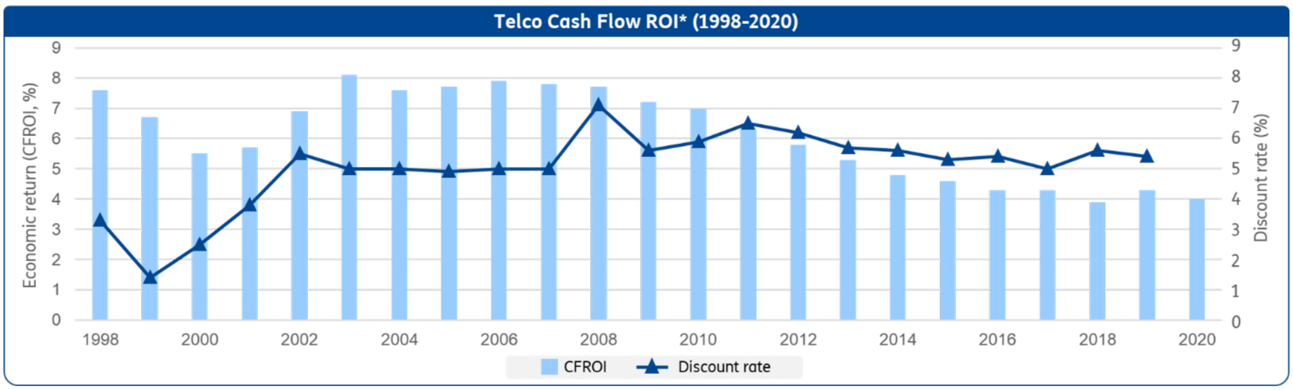

Figure 2 – Value destruction in European Telecom market: from 2012 ROIC is less than hurdle rate (Source: Sparkle Telecom)

But disruption usually fosters new opportunities, and prime players in the industry who embrace it are reimagining network services and capabilities to ride the wave of Telecom Transformation.

Many thought leaders concur that transforming a telecom all the way requires a bidimensional approach, often called "Dual Transformation Process":

- Process "A" – transformational process to reposition today's core business, to maximize its resilience and increase the capabilities. This dimension aims to restore business sustainability.
- Process "B" – transformational process to create new growth engines.

It's not simply using cash generated by "A" to speculate on targeted acquisitions and new ideas, but rather to identify and put at value unique capabilities to compete in new ways, starting from searching opportunities in adjacent markets.

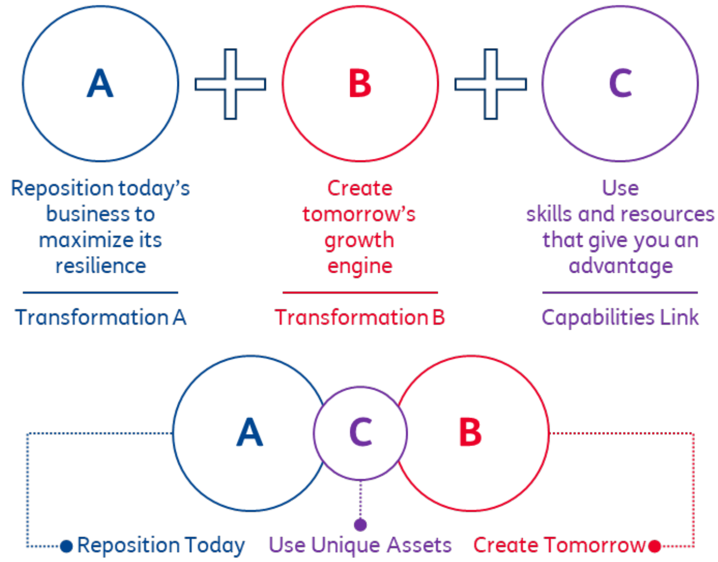

Figure 3 – Dual Transformation - Source: Innosight

Transformation "A" focuses on the infrastructure, on its evolution towards a Carrier Grade Trust Network, where services can be instantiated and consumed by end users on-demand and through API-enabled applications. Depending on the application, networks can be hardened and tailored to maximize security, resiliency, throughput and prioritization, while minimizing interference, latency, jitter and error rate. New alliances (ecosystems) between carriers that will implement East-West APIs to automate service ordering will be forged, generating the rise of a coopetition business model. At the same time, specific AI-based tools shall be introduced to evolve towards intent-based networks that can be instructed by natural language on the type of customers and services to be activated and equipped with self-healing capabilities. From a carrier provider standpoint, a new operations model will emerge, whereby Network Operations Centers, are progressively replaced by AI-driven Service Operations Centers, focusing on correlating customer service commitments with actual network performance.

Transformation "B" is to create new telecom services supporting new business models that leverage these enhanced core capabilities provided by Transformation "A". Product Teams will be challenged to learn to use unique assets and capabilities exposed both by their internal network factory and by partner carriers. They will also be challenged to integrate them with vertical solutions provided by non-carrier partners so as to design and build their version of "trust" capabilities, and be positioned to offer those solutions on a bespoke basis. Co-design and co-development will be the new way to tailor services to exactly "fit the size of the customer demands". For that reason Product Teams will have to support their sales organization to constantly meet the emerging needs of public entities, private enterprises, consumers and wholesale customers in the smart, connected, and IoT worlds. Of paramount importance will be the development of collaborative partnerships and investments - or acquisitions of disruptive and emerging growth companies who have the required hardware and suitable customer-facing applications and capabilities.

Software Defined Networking (SDN) and Network Functions Virtualization (NFV) are pillars for Transformation "A" that deal with the evolution of the so called "underlay" to build the basis of "digital overlays" mostly addressed by Transformation "B".

However, the maturity model of the two radically differ at the time of writing this white paper. SDN was born as a technique to abstract networks, by separating the control plane from the data plane and has reached a level of maturity whereby it is not anymore simply a “technology” topic, but rather an overall approach to network abstraction, automation, virtualization, programmability and optimization. The approach guarantees a focused common vision and understanding across Engineering, Operations, Customer Care, IT and Product teams. It represents a foundation for the holy grail of service velocity that envisages a model where Product teams are able to design a service offering, test that offering in labs and demo rooms, trial it in a defined market, review adoption, then make a decision on a wider launch of the offering to all markets in a matter of months with minimal investment. However, the same level of maturity and adoption does not yet exist in NFV ecosystems.

The foundational NFV white paper is already seven years old, yet the promise of NFV remains largely unfulfilled for a range of reasons relating to the current complexity of NFV operations:

- NFV solutions are **complex to deploy**, because they are closely coupled to how the rest of the infrastructure is managed.

- NFV solutions are **complex to automate**, because there are many NFV components that must be coordinated.

- New virtualized network functions (VNFs) and containerized network functions (CNFs) are extraordinarily **time-consuming to on-board**, because VNF and CNF developers have yet to receive clear and practical guidelines for designing VNFs and CNFs that can gracefully co-exist with general NFV management solutions.

Moreover, these problems will not just fade away with time, as they are a result of the current architectural approach to NFV. To make progress, a rethink is required of how to architect NFV solutions.

As we note later, this does not mean we have to throw out all our previous efforts; in fact, the Lean NFV approach should be seen as a complement to some of the existing open-source efforts. However, the core principles of Lean NFV, which we describe below, are very different from those the industry has followed until now.

3 Key Concepts and Definitions

To clarify our terminology, we revisit three main components of NFV solutions: NFV Manager; Computational Infrastructure; and VNFs. It is important to note that the Lean NFV architecture applies to Network Functions of any form factor, virtualized and containerized. For simplicity, this document will use VNFs to refer to both form factors.

NFV Manager

The NFV Manager is the entity that handles common lifecycle management tasks for both individual VNFs and end-to-end NFV service chains. The NFV Manager can be implemented either as a monolithic solution or, as recommended in this paper, as a distributed set of interacting subcomponents each of which specializes in handling one or more specific tasks.

Computational Infrastructure

The Computational Infrastructure comprises compute resources (bare metal or virtualized) and the connectivity between them (provided by a physical or virtual fabric); the former is managed by a *compute controller* (e.g., Openstack) and the latter by an *SDN controller*.

Virtualized Network Functions (VNFs)

VNFs can include both data plane and control plane components (e.g., an EPC's S/P-GW and MME). VNFs can optionally have an Elemental Management System (EMS), but only to handle VNF-specific configuration (leaving all other lifecycle management tasks to the NFV Manager).

In addition to these three well recognized NFV components, this document introduces a fourth component – the Key-Value store - that is central to the Lean NFV approach to reducing integration complexity between the first three components.

Key-Value Store
A Key-Value (KV) store, or Key-Value database, is a method for storing and managing data in a memory device. The stored data (i.e., the "value") and the key information for identifying that data (i.e., the "key") are stored as a pair. The KV store's API must support adding new key-value pairs to the store, retrieving the data associated with a given key, and providing notifications of when the data associated with a given key is modified.

KV stores are widely implemented in many ICT environments and examples are provided in a range of sources[3]. In the Lean NFV Framework, the KV store's purpose is to serve as a universal point of integration and reduce NFV integration complexity.

[3] Wikipedia

4 NFV Integration

The complexity currently hindering adoption of NFV arises from how the various components – NFV Manager, Computational Infrastructure and VNFs – are woven together into an overall system, not from their individual implementations. More specifically, the complexity arises when:

- the NFV Manager is integrated with the existing Computational Infrastructure
- the VNFs are integrated with the NFV Manager
- Coordination is required between the various components of the NFV Manager

Rather than standardizing all-encompassing architectures, or adopting large and complicated codebases, this document proposes focusing exclusively on using a KV store to simplify these three points of integration, leaving all other aspects of NFV designs open for innovation.

Integration: NFV Manager – Computational Infrastructure

Currently deployed infrastructure controllers can support a wide range of sophisticated features. Some NFV implementations exploit these capabilities to embed NFV Management capabilities into existing infrastructure systems such as OpenStack.

We propose that the NFV Manager be decoupled from the computational infrastructure as much as possible and *only* rely on a core set of capabilities that are supported by *all* infrastructure managers, namely:

- Provide NFV with computational resources,
- Establish connectivity between these resources, and
- Deliver packets that require NFV processing to these resources.[4]

All of the remaining functionality (e.g., chaining, auto-scaling, failure handling) should be established directly by the NFV Manager on the resources it has been provided without relying on additional infrastructure capabilities. Thus, the infrastructure can remain completely unaware of, and therefore undisturbed by, the presence of NFV functionality. Since only minimal infrastructure functionality is required, the Virtualized Infrastructure Manager (VIM) can be very lightweight.

We call this *plug-in integration*, because NFV solutions can be "plugged in" to any computational infrastructure. This decoupling makes deployment far easier for ICT-SPs than in current solutions, and enables innovation to proceed independently in both the computational infrastructure and the NFV Manager (e.g., ICT-SPs can change their SDN controller without impacting the NFV deployment, or switch to a new NFV Manager without modifying their infrastructure controllers).

[4] *Of course, an SDN controller can eventually play a more active role if desired, but our proposed starting point does not require that the SDN controller be aware of individual VNFs or chains; instead, the SDN controller is merely told to deliver NFV-bound flows to the compute nodes assigned to NFV. We expand on this point later in the paper. This simplicity helps avoid premature standardization of the features required in the computational infrastructure.*

Integration: Between NFV Manager Sub-Components

These components need to share information, but this sharing must be done in a way that is fully extensible, vendor-agnostic, and secure (i.e., supports strong access control). Modern KV stores provide exactly these capabilities. Thus, rather than implement the NFV manager monolithically, or build it out of separate components that coordinate via tightly-coupled pairwise APIs, we strongly recommend that NFV managers be comprised of components (which we call microcontrollers) that coordinate through the use of an open-source KV store. Each microcontroller can write information into the KV store (e.g., current status or various configuration parameters) in the form of key-value pairs; these entries can be watched by other microcontrollers which may take appropriate action (e.g., launch a new VNF instance or update internal configuration) as the values change. This preserves great flexibility of implementation while enabling interoperability of microcontrollers from multiple vendors: as long as they agree on key semantics, or use separate key spaces, they are compatible.[5]

Integration: VNFs – NFV Environment

On-boarding existing VNFs is very difficult, because most are merely repackaged versions of older implementations that rely on proprietary or obsolete interfaces. This situation has persisted because, despite seven years of NFV development, vendors still have been given little practical guidance for how to rewrite their VNFs to make integration easier. With Lean NFV, integration merely requires VNFs to use the KV store to read configuration information and expose their operational data. The KV store thus provides a universal and scalable mechanism for bidirectional communication between NFV management systems and VNFs (and also, as noted above, between microcontrollers). Thus, the use of a KV store makes it much easier to onboard new VNFs and introduce new management functionality. Moreover, it enables VNF authors to write once and know that their VNF can be used in many environments, thus promoting a much larger market of available VNFs for the ICT-SPs.

[5] *For scalability, large datasets (such as telemetry or other long-running time-series data) could be stored elsewhere, with the KV store allowing components to exchange information about where these large datasets can be found.*

5 Lean NFV Architecture

The resulting architecture is depicted in Figure 4 below. Reflecting the above design principles, the NFV manager is responsible for the management of both individual VNFs and end-to-end service chains, including lifecycle management tasks such as placement, launching, configuring, chaining, scaling, healing, monitoring, and upgrades. Note that while we show an NFV solution in a single cluster, the same principles can be applied in a hierarchical fashion for multi-site management.

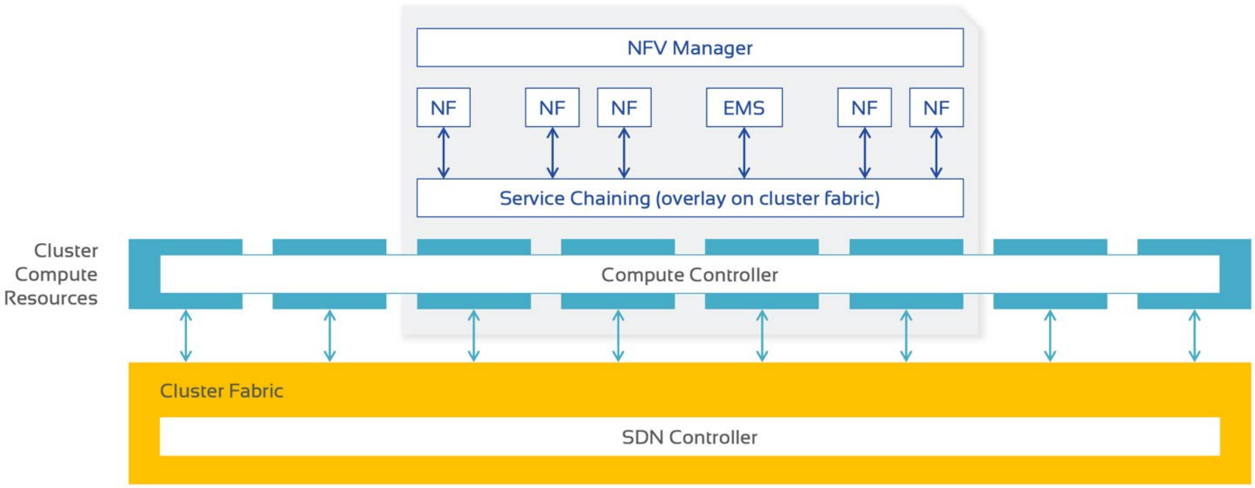

Figure 4 – The Lean NFV Architecture

All components of the NFV solution run on the infrastructure (compute resources and connectivity fabric) that are assigned to NFV processing by the infrastructure controllers as shown in Figure 5 below. The NFV manager is given a set of service chain definitions (in some form of declarative specification), and periodically reports on the status of these chains.

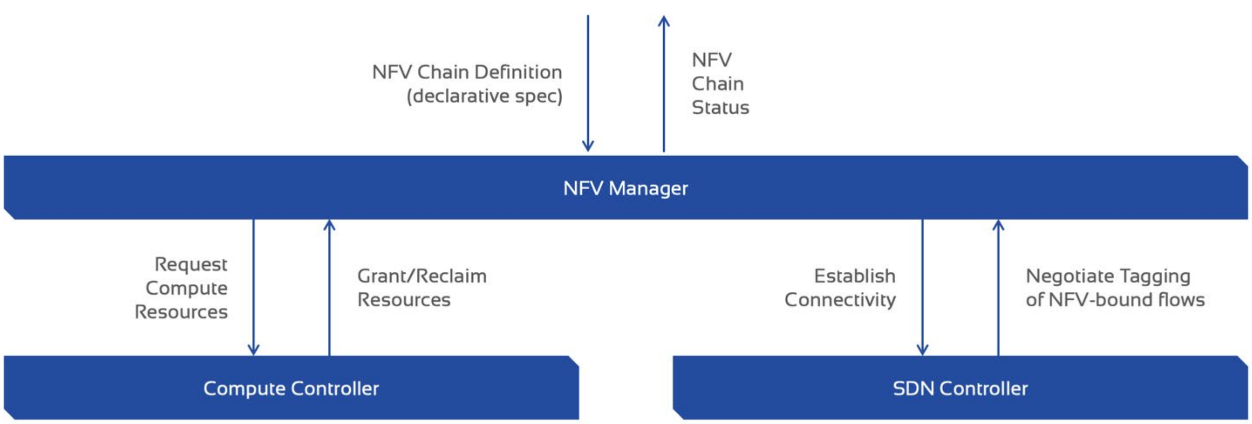

Figure 5 – Interaction Between the NFV Manager and Infrastructure Controllers

Finally, as mentioned before, the NFV manager itself should be architected as a collection of microcontrollers each of which addresses one or more specific aspect of management. These microcontrollers must coordinate only through the KV store, as shown in Figure 6 below (depicting an example set of microcontrollers). This approach improves modularity and simplifies innovation, as individual management features can be easily added or replaced. The KV store also functions as the integration point for VNFs (depicted on the left) and their (optional) EMSs; in such cases, coordination through the KV store can range from very lightweight (e.g., to notify the EMS of new VNF instances or to notify the NFV manager of configuration events) to actually storing configuration data in the KV store. Thus, use of the KV store does not prevent NF-specific configuration, and it accommodates a range of EMS implementations. Note that because the KV store decouples the act of publishing information from the act of consuming it, information can be gathered from a variety of sources (e.g., performance statistics can be provided by the vSwitch). This decreases reliance on vendor-specific APIs to expose KPIs, and hence allows a smooth migration from vendor- and NF-specific management towards more general forms of NFV management.

Note that this approach echoes the 3GPP notion of control and user plane separation (CUPS) in the 5G reference architecture, which also separates the publishing of information by VNFs from the consumption of information by other entities. Thus, Lean NFV already embodies the Service-Based Architecture required by 5G.

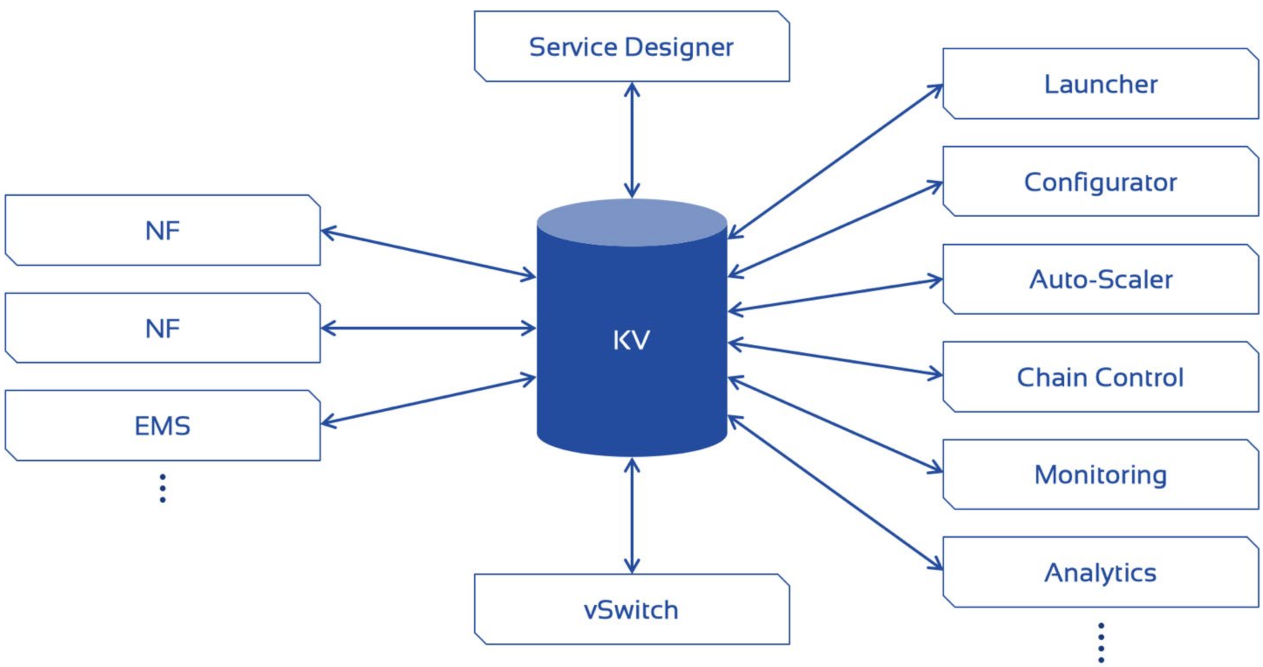

Figure 6 – Integration through KV Store

6 Summary

The Lean NFV approach relies on two technically straightforward design decisions: using an open-source KV store for coordination within NFV solutions (management components and VNFs), and using a plug-in approach to integration with the existing infrastructure. *Our central claim is simple: by adopting these two simple measures, the community can create a new NFV ecosystem that achieves the best of both worlds: easy deployment and rapid innovation.*

In terms of deployment, Lean NFV implementations are highly portable, in that such designs can be easily inserted into any reasonable computational infrastructure (including bare metal resources) merely by plugging them in. In terms of innovation, Lean NFV enables operators to transition from specialized EMSs to using general NFV management without changing the VNF itself. It also allows NFV managers to easily add new microcontrollers (whether from the same vendor or another) without adding any new APIs.

More generally, this approach is well-suited to cloud-native designs and the proposed 5G service-based architecture, both of which envision deeply disaggregated VNFs and highly modular management. Our Lean NFV proposal aligns perfectly with this vision by providing, via the KV store, a clean way of coordinating these various components.

In contrast to previous efforts, Lean NFV is neither a monolithic open-source effort nor a highly prescriptive architectural blueprint: instead, Lean NFV is an open architecture that only specifies the minimal requirements needed for interoperability. We expect that the components of the resulting ecosystem will come from both commercial vendors and open-source efforts, and can be mixed and matched as desired by operators. To initiate this process, we are developing open-source VNFs compliant with the Lean NFV approach so that all interested parties can see operational examples of this new approach.

The technical barriers to adopting Lean NFV are few and relatively minor. The far more challenging task is to create a community consensus around this new approach, so that vendors and operators can focus their efforts on creating this new ecosystem. We ask all interested parties to endorse this document, engage in a broader discussion of this approach, and consider providing components adhering to its design principles.

7 About MEF

An industry association of 200+ member companies, MEF has introduced the MEF 3.0 transformational global services framework for defining, delivering, and certifying assured services orchestrated across a global ecosystem of automated networks. MEF 3.0 services are designed to provide an on-demand, cloud-centric experience with user- and application-directed control over network resources and service capabilities. MEF 3.0 services are delivered over automated, virtualized, and interconnected networks powered by LSO, SDN, and NFV. MEF produces service specifications, LSO frameworks, open LSO APIs, software-driven reference implementations, and certification programs. MEF 3.0 work will enable automated delivery of standardized Carrier Ethernet, Optical Transport, IP, SD-WAN, Security-as-a-Service, and other Layer 4-7 services across multiple provider networks. For more information, visit https://www.mef.net and follow us on LinkedIn and Twitter @MEF_Forum.

8 Acknowledgements

We acknowledge the authors and many signatories of the original Lean NFV white paper[6] on which this document is based.

- Sylvia Ratnasamy (Nefeli Networks and UC Berkeley)
- Scott Shenker (Nefeli Networks and UC Berkeley)
- Christian Martin (Arista Networks)
- Krish Prabhu (Formerly with AT&T)
- Mirko Voltolini (Colt)
- Nagesh Nandiraju (Comcast)
- James Feger (F5)
- Ying Zhang (Facebook)
- Bikash Koley (Google)
- Uri Elzur (Intel)
- Raj Yavatkar (Juniper)
- Shyamal Kumar (Lavelle Networks)
- Travis Ewert (LightRiver)
- Nabil Bitar (formerly with Nokia)
- Jian Q. Li (NTT Global)
- Ravi Srivatsav (formerly with NTT)
- Daniele Mancuso (Sparkle)
- Constantine Polychronopoulous (VMware)

[6] Lean NFV White Paper – www.leannfv.org