Multi-Domain Orchestration for the Deployment and Management of Services on a Slice Enabled NFVI

Francesco Tusa, Stuart Clayman, Dario Valocchi, Alex Galis
Dept. of Electronic Engineering, University College London, London, UK
Email: francesco.tusa@ucl.ac.uk, s.clayman@ucl.ac.uk, d.valocchi@ucl.ac.uk, a.galis@ucl.ac.uk

***Abstract*—This paper presents a unique and new scenario where services are deployed across distributed domains, and each domain MANO (for VNF Management and Orchestration) requests and creates an on-demand Data Center slice to run those service elements. Whilst each domain has its own MANO, the MANOs are configured in a north-south way creating a hierarchy of service provision capabilities, rather than configured in the more common peer-to-peer approach. This approach works particularly well where each domain, from the mobile edge, to the core DC, can be managed independently of the others, but needs to be combined to form slices. End-to-end slices across the whole infrastructure provide a more effective resource management and also better support the customers' mobility requirements. The details of the architecture to support this scenario are described.**

I. Introduction

In the 5G landscape, different Infrastructure and Service Providers can explore new business models and federate their resources and service offerings in order to provide the customers with the ability of instantiating end-to-end services across multiple technological domains. These services can be either located in separate geographical locations of one Provider, or outside of its own administrative borders.

In order to facilitate the allocation of resources for the end-to-end services, each 5G Provider / Operator may take advantage from using complex software systems that perform of the management, control and orchestration of all the resources to be allocated for the deployment of 5G Service Function Chains. These systems are usually referred to as MANO(s) (VNF Management and Orchestration [7]) as they are able to deal with the tasks required to build Network services by combining multiple Virtual Network Functions (VNFs).

In the 5G landscape, a Provider or Operator can deploy network services whose elements could potentially span the whole infrastructure from the mobile edge to the core. Moreover, the resources utilised in those segments of the infrastructure could be managed by a particular MANO system that an organisational unit of that Provider is not willing to replace or update. The instantiation of full end-to-end services can be achieved via an inter-MANO interaction, based on a north-south interface interaction between the involved orchestration systems. Each north-bound MANO system interacts with its south-bound counterpart as it would do with a VIM (Virtual Infrastructure Manager).

Slicing is a move towards segmentation of resources and deployment of NFV for the purpose of enhanced services and applications on a global shared infrastructure. In order to support service provisioning over a slice enabled distributed NFVI, it will be required to implement mechanisms that can support the slicing of the whole end-to-end infrastructure – from the mobile edge to the core DC – including network, compute and storage resources. In this paper we consider the extension of the above MANO-to-MANO north-south interaction to support the concept of slicing.

This new slice-based orchestration approach on the one hand will provide a more effective resource management thanks to the isolation introduced at the control and data planes; on the other hand, slicing will facilitate the implementation of new emerging 5G scenarios where customers look for seamless mobility across the whole infrastructure: the implementation of end-to-end slices that include different locations of the edge infrastructure will help supporting physical / virtual mobility of devices, network and services.

To manifest this slice approach, we have designed and built a DC Slice Controller which is able to allocate a slice of a DC and create a per-slice VIM in an on-demand fashion. The DC slice and the VIM are provisioned solely for use with the service. Each slice and its associated VIM are independent of the other slices and VIMs. In this way, customers will never share servers, and the worry of VMs of one customer interacting or spying on another customer will be eliminated. Also, the issue of one customer's VM consuming all the resources and starving other customer's VMs is ameliorated to some extent.

Each of these slices will be allocated and de-allocated in an on-demand fashion. A customer can interact with a Slice Controller and request the instantiation of a new slice. The resulting slice will be isolated from the others such that each slice will get its own VIM or WIM (Wide-area Infrastructure Manager), and will not have management as part of a shared VIM as the DC resources will be isolated from each other.

Having different types of VIMs in different segments of the infrastructure will allow creating an end-to-end slice also considering the type of resources that will host that particular VIM: lightweight VIM versions might be instantiated at the mobile edge, whereas the core DC might host more traditional VIM implementations, such as OpenStack.

Section II reports about the existing slicing approaches based on the analysis of current the state-of-the-art. Section III presents the concepts and the design characteristics of the components implementing the Slicing functionalities in the considered multi-MANO scenario. Sections IV and V describe the architectural elements, interactions and initial results that are relevant for the considered multi-MANO scenario. This HSP scenario over a slice enabled NFVI was analysed during a 5G-PPP cross-projects activity, which led to the implementation of a proof-of-concept demonstrator able to instantiate end-to-end network services via a 5GEx [14] based MANO and one or more instances of a SONATA [6] MANO (i.e., the SONATA Service Platform).

II. Related Works

The desire to provide services on top of slices, by logically partitioning resources of a multi-domain software-defined infrastructure can be obtained through different slicing strategies depending on which system elements provide the slicing and at what layer the slicing is introduced. The overall concept of network slicing is outlined by various standards organisations including ETSI [8], the IETF [11] and NGMN [4].

Slicing at lower layers, namely infrastructure slicing, means that upper layers, such as VIMs and Orchestrators do not need to directly know about slicing. If a slice is presented to one of these VIMs or Orchestrators, they can carry on working with no change or minimal change. Such approaches are presented in [13] and [9].

If slicing is done in the Orchestrator, which uses an inter-domain orchestrator API interaction and / or a peer to peer approach, a slice is closer to a small abstraction in the Orchestrator, rather than an infrastructure partition. There are various projects and initiatives that are doing slicing at the Orchestrator level. These include 5G-Transformer [1], 5G PAGODA [2], SLICENET [12]. Their approach has the easiest entry position, but all the main software elements need to be updated and adjusted to know about slices; and all of the APIs, the modules, and internal function paths, and the data structures need to be adjusted and adapted to factor in slices.

Consequently, there are inherent trade-offs when selecting one or the other slicing approach. The actual decision on which slicing approach will depend on various key aspects of the service requirements under consideration, and can be focussed on the technical desires of the provider, together with the technical abilities and technological choices of the tenants.

The recent survey [3] has been looking at approaches to 5G slicing. Additional characteristics, standard and research activities on slicing and references can be found in [10].

III. Slicing concepts and implementation

While working on the design and implementation of the multi-MANO scenario that we are going to describe in this paper, we were faced with a fundamental question about slicing. How is it possible to connect the various segments, including the mobile edge, the core networks, and the data centers, as well as what layer of the existing MANO architecture should implement it. In particular, we identified two possible answers to that question: namely, (i) changing the existing orchestration and management software elements to deal with slices, or (ii) considering slicing at a lower layer and presenting a slice to the upper layers. If we choose the former approach, then all the main software elements would need to be updated and adjusted to know about slices, together with all their related APIs. We realised that if slicing is implemented at a lower layer, then the upper layers, such as VIMs and Orchestrators, would not need to know anything about slicing [13].

The following sections describe the slicing model based on the second approach and making use of the new concept of Data Center slicing and VIM on-demand. We present how a Data Center can be sliced and how a VIM on-demand can be allocated for the slice.

A. VIM on-Demand Slices

A Slice is an aggregated set of resources that can be used in the context of an end-to-end networked service comprised of virtual network functions. Slices are composed of multiple resources which are isolated from other slices and allows logically isolated network partitions, with a slice being considered the basic unit of programmability using network, computation and storage. When considering the wide variety of applications to be supported by 5G networks, it is necessary to extend the concept of slicing to cover a wider range of use-cases than those targeted by the current SDN/NFV technologies.

If we have slicing everywhere, including networks and DCs, we observe the following attributes: (i) there is a separation of physical resources; (ii) there is isolation of services as no customers share physical resources; and (iii) it is secure as only specified customer can access host, no sharing or cross VM issues.

In order to support service provisioning over these slices, it is necessary to have mechanisms to support the slicing of the network resources and the Data Center compute and storage resources across the different segments of a Provider's infrastructure. To manifest this slice approach, we have designed a built a DC Slice Controller which is able to allocate a slice of a DC and create a per-slice VIM in an on-demand fashion. The DC slice and the VIM are provisioned solely for use with the service. Each slice and its associated VIM are independent of the other slices and VIMs. In this way, customers will never share servers, and the worry of VMs of one customer interacting or spying on another customer will be eliminated. Also, the issue of one customer's VM consuming all the resources and starving other customer's VMs is also ameliorated to some extent.

Each of these slices will be allocated and de-allocated in an on-demand fashion under software control. A customer can interact with a Slice Controller, and request a new slice. The resulting slice will be isolated from the other slices. Furthermore, each slice will get its own VIM or WIM, and not have management as part of shared one. As each VIM instance is independent of the others, the best type of VIM can be selected for deployment according to the type of resources available in each part of the segmented end-to-end infrastructure. Figure 1 presents how the resources of a DC are isolated from each other, and how a Slice Controller is involved in such a process.

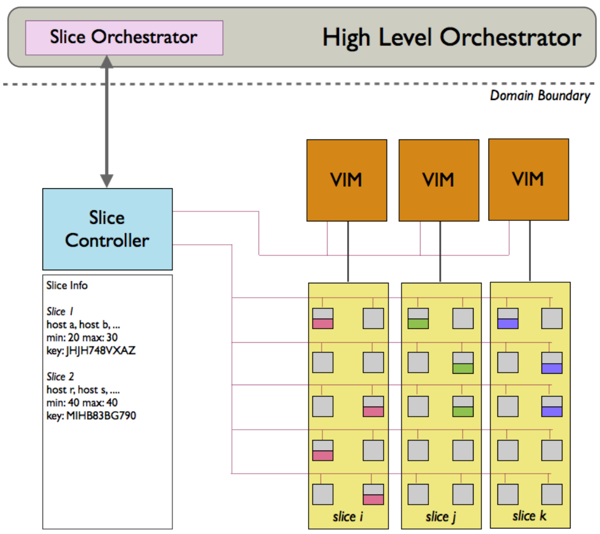

Figure 1. DC Slicing via the Slice Controller

B. UCL Slice Controller

Here we describe the UCL Slice Controller which supports Data Center slicing and the VIM on-demand model. We present each of the main components and describe their functions. The Slice Controller has the following elements for its operation:

- a *Resource Manager* — which manages all of the resources in the DC and keeps a track of which resources have been allocated to which slice.
- a *Slice Information Store* — which database lists all of the slices and all of the resources in the slice.
- a *Slice Creator* — which is responsible for handling requests for slices and interacting with the Resource Manager to determine if it is possible to create a new slice. If the slice creation is possible it interacts with the VIM Factory.
- the *VIM Factory* — which is able to allocate a VIM and configure it to use the resources which have been picked by the Slice Creator. Once the VIM is allocated and deployed, the REST entry point is returned.
- the *VIM Placement Manager* — which is responsible for determining which host should be used to execute a newly created VIM.

IV. Multi MANO HSP Architecture Elements

This section describes the architectural elements that have been utilised for the design of the multi-MANO Hierarchically structured Service Provider (HSP) scenario on a sliceable NFVI. For this multi-MANO HSP scenario, we utilize a higher level 5GEx MANO to interact with a lower level SONATA MANO to form the multi-domain end-to-end slice. We therefore present the 5GEx MANO and the SONATA MANO to highlight their relevant functions and APIs that are of interested for the implementation of the considered scenario.

A. 5GEx overview

The 5GEx project architecture framework [14] depicted in Figure 2 identifies the main components and the inter-working interfaces involved in the multi-domain orchestration. The top of the figure shows the key 5GEx MANO component – the Multi-provider Multi-domain Orchestrator, which we refer to as Multi-domain Orchestrator (MdO) for simplicity. The middle part shows the Domain Orchestrators, responsible for performing Virtualization Service Orchestration and / or Resource Orchestration exploiting the abstractions exposed by the lower Resource Domains. The bottom part of the figure shows various Resource Domains, hosting the actual resources.

The MdO handles the orchestration of resources and services from different providers, coordinating resources and service orchestration at the multi-domain level. Multi-domain refers to multi-technology (orchestrating resources and services using multiple Domain Orchestrators); or multi-provider (orchestrating resources and services using Domain Orchestrators belonging to multiple administrative domains).

The 5GEx I3 interface (from Management to Domain Orchestrator) was designed and implemented to allow the interaction between a MdO and the underlying Domain Orchestrators. In particular, the sub-interface I3-RC is responsible for retrieving the local resource availability from the technological Domains and for enabling their programmability. I3-RC is based on the 5GEx common domain abstraction API [14]. A model called the *BiS-BiS node model* is considered to

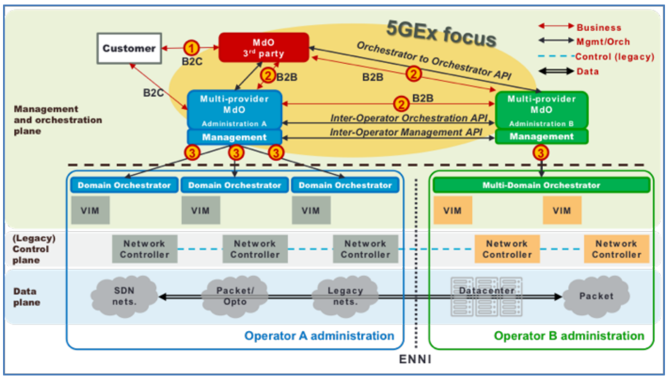

Figure 2. 5GEx Framework

represent the resources and capabilities of each domain. The I3-RC interface and BiS-BiS abstractions play a key role in the implementation of the north-south interface interconnecting the MANO systems across domains, which are considered in the HSP scenario (as further described in Section V).

B. SONATA overview

The SONATA project [6] is based on a customisable Service Platform with a NFV Orchestrator that supports Network Service Software Development Kit (SDK) for developers and specialised DevOps workflows. One major component of the SONATA architecture is the SDK, that provides service developers with both a programming model and a set of software tools. The SDK allows developers to define complex services consisting of multiple VNFs. A service provider (which can also be the service developer) can then deploy and manage the created services on one or more SONATA service platforms through the corresponding Gatekeeper components. A second major component of the system is the SONATA Service Platform (SP). Due to the modular design of its management and orchestration (MANO) framework, the SP offers customisation opportunities by replacing components of the loosely coupled MANO framework (MANO plugins). SONATA's functional elements are represented on Figure 3.

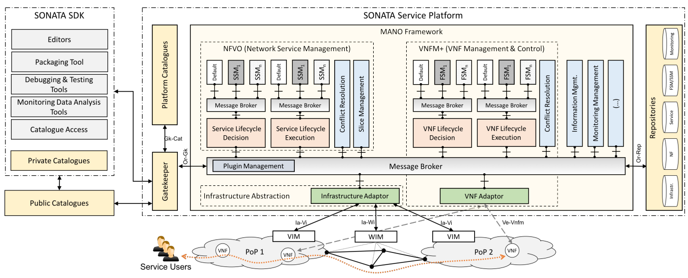

Figure 3. SONATA Framework

The Gatekeeper module is the main entry point of the Service Platform and implements its Northbound Interface. It exposes API endpoints for SDK tools and provides the following functionalities: (i) accepting new developers' to be part of the contributors of new developed services; (ii) accepting new services, in the package format, to be deployed on the platform; (iii) validating submitted packages, both in terms of file format and developer submitting the package; (iv) accepting new service instance requests from customers interested in instantiating a service; and (v) following a service performance through automatically monitoring each on-boarded service or function. The Gatekeeper directly interacts with the platform-specific Catalogue, which is the entity storing service artefacts uploaded to the platform.

The Gatekeeper plays an important and active role in the implementation of the communication interface in our inter-MANO HSP scenario, as it provides the API methods that are used by the higher level 5GEx MANO to interact with the lower level SONATA MANO. Further details are reported in the following section where we show how these elements are composed to form the full hierarchy.

V. End-to-End Multi-MANO HSP System

This section provides an overview of how the different MANO architectural elements described in Section IV have been combined in order to build a novel hierarchical slice-enabled multi-MANO environment for the deployment and management of end-to-end 5G network services.

A. 5GEx-SONATA Multi-MANO System

The implementation of the above inter-MANO scenario is based on the 5GEx MANO system at the top level of the hierarchy, playing the role of the highest level orchestrator. This choice represents the straightforward design approach as the 5GEx system was originally conceived and built to be a multi-domain orchestrator. At the lower level of the orchestration hierarchy, an instance of the SONATA Service Platform is deployed to operate on the resources of its own segment of the NFVI, and behaving as a Domain Orchestrator, accessed via a SONATA Domain Adapter.

The 5GEx MANO operates over its own segment of the NVFI using the Data Center slicing approach discussed in Section III: the on-demand allocation of lightweight VLSP [5] VIM instances is triggered by the VLSP Domain Adapter interacting with the local Slice Controller. The VLSP Domain Adapter in this scenario plays the same role played by the Slice Orchestrator component in Figure 1. Figure 4 depicts the architectural view of this inter-MANO scenario.

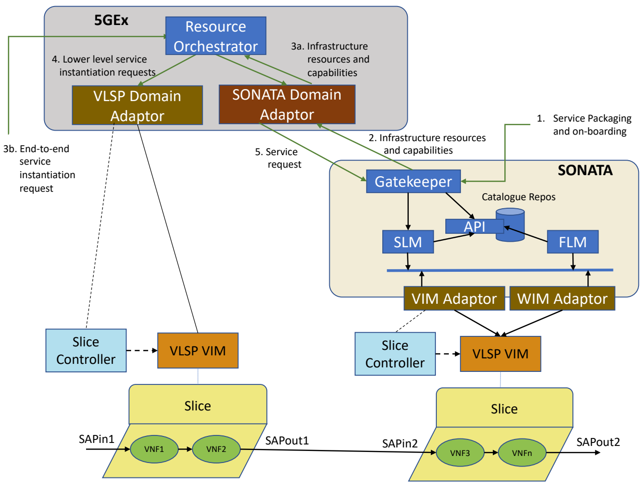

Figure 4. 5GEx-SONATA detailed multi-MANO interaction

In a similar way, the SONATA SP operates on top of another segment of the NFVI, again using the Data Center slicing capabilities and on-demand instances of the VLSP VIM. In this case, the SONATA's own Infrastructure Adapter drives the creation of slices by interacting with the local Slice Controller which offers to the SONATA MANO the interface to interact with the on-demand allocated VIM.

The multi-MANO interworking happens via the interaction between the 5GEx MANO SONATA Domain Adapter component and the Gatekeeper component in the SONATA SP. The SONATA Domain Adapter utilises the REST API exposed by the Gatekeeper, and provides the 5GEx Resource Orchestrator with an abstract view of the resources and services of the underlying slice in the SONATA domain, via the I3-RC interface introduced in Section IV-A. More specifically, the services and the required VNFs that have previously been on-boarded to the SONATA Service Platform are exposed by the SONATA Domain Adapter as Domain Capabilities to the 5GEx Resource Orchestrator, together with an abstracted view of the resources available from the DC slice.

When a request for instantiating an end-to-end service is received at the top-layer MANO, the 5GEx Resource Orchestrator will deploy the requested service elements in the different allocated DC slice parts according to the resources and capabilities reported by the Domain Adapters. Virtual links will be created between the allocated service elements (e.g., VNF1, VNF2 and VNF3 in Figure 4) for the implementation of the requested end-to-end Service Function Chain.

The following section provides further details on the considered multi-MANO interaction, highlighting the detailed interworking of the systems, based on the data models, abstractions and interfaces that have been used for its implementation. Figure 4 shows the different steps of the interaction between the relevant components of the 5GEx and SONATA MANO frameworks and how have been used to build the HSP scenario.

B. Experimental Implementation and Evaluation

Imagine a service developer wishes to deploy a service on a 5G Provider infrastructure, where one branch of the provider is managed by the SONATA MANO and the another branch uses the 5GEx MANO system. It is important that all service elements and VNFs are on-boarded before any deployment can occur. For 5GEx, the VNFs and their descriptors are on-boarded when the MANO starts, and services get comoposed from VNFs. On the SONATA platform, a packaged service is on-boarded to the SONATA SP (Step 1 on Figure 4).

On initialization, in the branch that makes use of the 5GEx MANO system, the VLSP Domain Adaptor creates a slice and then collects local information. Also, the SONATA Domain Adaptor interacts with the SONATA Gatekeeper API (Step 2 on Figure 4) to also create a slice, and gather both resources available from the external SONATA NFVI and a list of services and functions from the SONATA Catalogue. The slice creation is driven by interacting with the Slice Controllers. On-demand instances of the lightweight VLSP VIM are allocated as part of the DC slice parts and reference to them are returned to the adapters.

The 5GEx Resource Orchestrator retrieves the available DC slice resources, functions, and services from the connected underlying Domains via the I3-RC interface (Step 3a on Figure 4). The Domain Adapters expose a BiS-BiS [14] abstracted view of the resources coming from both the VLSP Domain and the SONATA Domain. The available collected DC slice resources are added to the BiS-BiS view as computational characteristics (CPU, memory and storage). The available services and functions are included as domain capabilities. The end-points of the on-demand VIMs instantiated in the sliced NFVI are also collected and mapped as Service Access Points, [14] of that sliced resource domain.

A service instantiation request is issued to the 5GEx Resource Orchestrator, which will split the required end-to-end service elements on the lower-layer resource domains according to the capabilities and resource availability reported by the corresponding domains (Step 3b on Figure 4). Next, embedding occurs where lower level service instantiation requests are sent to the individual Domain Adapters according to the embedding decisions (step 4 on Figure 4).

The end-to-end service request includes service elements available from the SONATA domain, and so the SONATA platform will receive a service instantiation request from the SONATA Domain Adapter on the relevant endpoint of the Gatekeeper API (step 5 on Figure 4). From the SONATA SP perspective, the request will look like a normal service request, such that the normal SONATA service deployment flow will follow the request, instantiating the entities as described in the package on-boarded by the service developer in Step 1.

In our scenario, the WAN configuration between the two NFVI segments is performed via the functionality of the VLSP VIMs at the service deployment time. More specifically, the upper 5GEx Resource Orchestrator takes care of mapping the Service Access Points to the service elements deployed at the edge of each slice part of Figure 4. It requests the SONATA SP to do the same process. This results in creating a per service overlay tunnel on the physical network that inter-connects the different segments.

For validation, we evaluated this setup on the UCL server testbed. We implemented a service as an orchestrated deployment of VLSP lightweight VNFs across the sliced NVFI. After successful instantiation of the requested end-to-end service, network traffic was generated and injected in the service function chain using the iperf network bandwidth measurement tool. More specifically, an iperf server was attached to the SAPout2 of Figure 4 and was able to receive the packets generated by an iperf client instance attached to SAPin1.

We successfully demonstrated an end-to-end service instantiation, using multiple MANOs, which had service elements executing on the different resource domains, that were all interconnected to create a single end-to-end service function chain. Moreover, the service elements were running on a slice of each domain, where each slice was allocated on-demand.

VI. Conclusions

This paper considered emerging 5G scenarios where different segments of a Service Provider's infrastructure – from the edge cloud to the central DC – are administered by separate organisational divisions / departments, which possibly leverage on heterogeneous MANO systems, in each division, for the deployment of the end-to-end services on the softwarised distributed resource infrastructure. Whilst each domain has its own MANO, the MANOs are configured in a north-south way creating a hierarchy of service provision capabilities, called the Hierarchically structured Service Provider (HSP). This approach works particularly well where each domain, from the mobile edge, to the core DC, can be managed independently of the others, but needs to be combined to form slices.

We discussed how the end-to-end slicing approach can provide a more effective resource management for the deployment of customer services. We demonstrated why having a multi-MANO hierarchical orchestration system with slicing support can be a viable approach to address the requirements of that scenario, and we presented this unique and new configuration where services are deployed across distributed domains.

We believe that our HSP experiment is still exploratory work on the very wide subject of MANO platforms interworking. In this paper, we created a first design and implementation for the problem, by providing a solution for an inter-MANO interaction based on a hierarchical approach that includes the 5GEx and SONATA MANO frameworks. We demonstrated how the rich data models, abstractions, and flexible APIs provided by those systems could be considered as reference, in future work, for the design and implementation of more generic HSP MANO-to-MANO scenarios.

Acknowledgements

This work was supported by the EU project NECOS – "Novel Enablers for Cloud Slicing" (777067).

References

[1] 5G-TRANSFORMER. 5G Mobile Transport Platform for Verticals, 2017. http://5g-transformer.eu.

[2] 5G!PAGODA. EU-Japan - A network slice for every service, 2017. https://5g-pagoda.aalto.fi.

[3] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck. Network Slicing and Softwarization: A Survey on Principles, Enabling Technologies, and Solutions. *IEEE Communications Surveys and Tutorials*, 20(3):2429–2453.

[4] NGMN Alliance. Description of Network Slicing Concept", NGMN 5G P1 Requirements & Architecture, Work Stream End-to-End Architecture, 2016.

[5] Stuart Clayman, Lefteris Mamatas, and Alex Galis. Efficient Management Solutions for Software-Defined Infrastructures. In *First IFIP/IEEE International Workshop on Management of 5G Networks (5GMan 2016) in conjunction with the IFIP/IEEE Network Operations and Management Symposium (NOMS 2016)*. IEEE/IFIP, 2016.

[6] S. Dräxler, H. Karl, M. Peuster, H. R. Kouchaksaraei, M. Bredel, J. Lessmann, T. Soenen, W. Tavernier, S. Mendel-Brin, and G. Xilouris. Sonata: Service programming and orchestration for virtualized software networks. In *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 973–978, May 2017.

[7] ETSI. ETSI GS NFV-MAN 001 V1.1.1. Network Functions Virtualisation (NFV); Management and Orchestration.

[8] ETSI. ETSI Report on Net Slicing Support with ETSI NFV Architecture Framework.

[9] Leandro A. Freitas, Viníıcius G. Braga, Sand L. Correˆa, Lefteris Mamatas, Christian E. Rothenberg, Stuart Clayman, and Kleber V. Cardoso. Slicing and Allocation of Transformable Resources for the Deployment of Multiple Virtualized Infrastructure Managers (VIMs). In *Workshop on Advances in Slicing for Softwarized Infrastructures (S4SI)*, Montreal, Canada, 2018.

[10] A. Galis and K. Makhijani. Infrastructure Slicing Landscape – Tutorial. In *IEEE NetSoft 2018*, Montreal, Canada, 2018.

[11] IETF. Network slicing - revised problem statement, draft-galis-netslices-revised- problem-statement-03, 2018.

[12] Slicenet. End-to-End Cognitive Network Slicing and Slice Management Framework in Virtualised Multi-Domain, Multi-Tenant 5G Networks, 2017. https://slicenet.eu.

[13] Stuart Clayman. Network Slicing Supported by Dynamic VIM Instantiation. In *IETF 100, Singapore*, 2017.

[14] I. Vaishnavi, J. Czentye, M. Gharbaoui, G. Giuliani, D. Haja, J. Harmatos, D. Jocha, J. Kim, B. Martini, J. Melián, P. Monti, B. Nemeth, W. Y. Poe, A. Ramos, A. Sgambelluri, B. Sonkoly, L. Toka, F. Tusa, C. J. Bernardos, and R. Szabo. Realizing services and slices across multiple operator domains. In *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, pages 1–7, April 2018.