On Orchestration Interactions for Preparation of Slice Creation in Multi-domain 5G Networking Environment

Eugen Borcoci, Serban Georgica Obreja
University POLITEHNICA of Bucharest
Bucharest, Romania
eugen.borcoci@elcom.pub.ro, serban@radio.pub.ro

***Abstract*— Network slicing is a major concept in 5G networking technology, offering multi-tenant, multi-domain, multi-operator and end-to-end (E2E) capabilities. 5G slicing allows tenants to share resources while customizing their own services via slice isolation. Integrated management and control based on ETSI Management and Orchestration (MANO) is applied, but enriched, in order to cope with multi-domain slicing. Several architectures have been proposed in the literature, with different views on the complex orchestration functions. This paper is oriented to some open architectural issues. It presents a short analysis of the orchestration interactions, focusing to some aspects of initial slice design and preparation, in a multi-domain and multi-operator environment. A specific paper contribution is on inter-domain topology discovery and collection of information by a multi-domain capable slice orchestrator, especially on specifying the design and preparation phase of a slice construction.**

Keywords-5G slicing; Multi-domain; Multi-tenant; Management and orchestration; Software Defined Networking; Network Function Virtualization

I. INTRODUCTION

The emergent 5G mobile network technologies aim to answer the increasing demand and challenges addressed to communication systems and Internet [1]. 5G can support dedicated use-cases and provide specific types of services to satisfy simultaneously various customer/tenant demands in a multi-tenant fashion [2]-[4].

The 5G network slicing concept (based on virtualization and softwarization) enables programmability and modularity for network resources provisioning, adapted to different vertical service requirements (in terms of bandwidth, latency, etc.) [2]-[9]. A *Network Slice* (NSL) is a managed logical group of subsets of resources, Physical/Virtual network functions (PNFs/VNFs) in the architectural Data Plane (DPl), Control Plane (CPl) and Management Plane (MPl). The slice is programmable and has the ability to expose its capabilities to the users.

The NSL behavior is realized via *Network Slice* Instance(s) (NSLI), which are created (based on NSL templates) at request of a tenant or at Slice Provider initiative (the word "tenant" defines a user or group of users with specific access rights and privileges over a shared set of resources). A *blueprint/template* is a logical representation of network function(s) and the associated resource requirements. It describes the structure, configuration and work flows for instantiating and controlling an NSLI; it includes certain network characteristics (e.g., bandwidth, latency, reliability) and refers to the required physical and logical resources, and the sub-networks. An NSLI may be dedicated or shared across multiple *Service Instances*.

The verticals may use a common infrastructure, with appropriate levels of isolation and Quality of Services (QoS) provisioning. The slicing allows 5G to create an eco-system, multi-tenant, multi-domain, multi-operator and end-to-end (E2E) - capable.

Software Defined Networks (SDN) and Network Function Virtualization (NFV) technologies, combined with cloud/edge computing are used in 5G slicing architectures and implementations [10]-[13].

In a multi-domain, multi-tenant, multi-operator, and E2E context, the slice construction includes a design and preparation phase. This phase requires a lot of orchestration/inter-domain interactions in order to construct the network services (NS) and NSL catalogues to be used later for actual slice creation in the subsequent phases. Information on available resources (network, compute, storage) in several domains should be collected by an orchestration entity charged with the task of creation of a multi-domain slice.

The specific contribution of this paper is to propose a mechanism for multi-domain interconnection topology information collection, by the highest-level orchestration entity. The mechanism is essentially usable in the design and preparation phase of an NSLI. This work is still preliminary, limited for the time being, to architectural and high level interactions solutions.

The structure of the paper is described here. Section II recalls some major features of the 5G management and orchestration framework. Section III provides few relevant examples of related architectural work on orchestration. Then, based on a generic relevant architecture, Section IV details the design and preparation phase interactions, providing inputs to the Section V. The latter is focused on collection of topological information in a multi-domain context. Section VI presents conclusions and a future work outline.

II. NETWORK SLICE MANAGEMENT AND ORCHESTRATION

The objective of this section is to shortly introduce the slice management and orchestration framework, in order to help the identification of the work area for Section V

The 5G networks need efficient services and resource *orchestration* and programmable management systems. It is

 ISBN: 978-1-61208-721-4

necessary to model the E2E services and to abstract and automate the control of physical and virtual resources. Orchestration extends the traditional management: it consists in a coordinated set of activities to automatically select and control multiple resources, services and systems, aiming to meet certain objectives (e.g., serve a tenant requesting a specific network service) [14].

At services level, the term *Network Service Orchestration* (NSO) can be defined, [15]-[17], as an automated management and control (M&C) process for services deployment and operations, performed mainly by telecommunication operators and service providers. The NSO involves different types of resources and potentially multiple providers; it can decouple the high-level service layer from the underlying management and resources layers (e.g., NFV functional components like Virtualized Infrastructure Manager (VIM), Element Management Systems (EMS), SDN controllers, etc.). The NSO defines the interaction with (chains of) network functions (NFs) of the underlying technologies and infrastructures through adequate abstractions. In a multi-domain (technological or administrative) context, the NSO should have an overall high-level view of all domains, *including topology information*, in order to be able to orchestrate E2E slices and associated services, independently of geographical location. The NSO cooperates with NFV MANO functional blocks.

The orchestration activities can appear at several architectural levels (e.g., slice, resources and NFV MANO - levels). An orchestration entity interacts, vertically and horizontally, with other M&C entities in the same or peer domains. Currently, there is not yet a standard for information exchange process in multi-domain technological/administrative environments. There are many multi-domain orchestration architecture variants and candidates, proposed (see [2]-[6] for examples). This paper is working in the information exchange area.

The *life-cycle management* (LCM) of an NSLI comprises several phases performed by the Slice Provider: (1) *instantiation*, *configuration*, and *activation*, (2) *run-time* and (3) *decommissioning* [6]. The phase (1) is split into the instantiation/ configuration sub-phase (the necessary shared/dedicated resources, including NFs, are configured and instantiated, but not yet used) and the activation sub-phase (the NSLI becomes active for handling network traffic). The run-time phase focuses on data traffic transport, reporting the network service performance and possible NSLI re-configurations or scaling, if dynamic conditions impose that. The phase (3) includes the deactivation and termination of the NSLI and release of the allocated resources.

The LCM is preceded by a *design and preparation phase* (0) for the future instantiation and support of an NSLI. The functional architecture, steps and interactions within this preliminary phase are still open research issues. The paper is focused on design and preparation phase, treated in Section V.

III. EXAMPLES OF RELEVANT ARCHITECTURES

This section presents few examples of relevant 5G slicing architectures in order to emphasize the orchestration entities roles.

A. *ETSI slicing architecture (high level)*

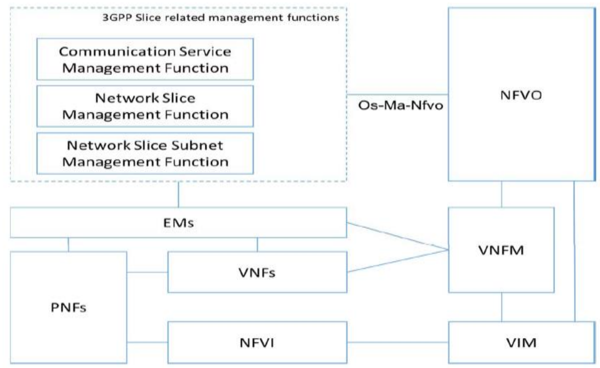

Figure 1. Network slice management in an NFV framework (ETSI GR NFV-EVE 012 V3.1.1, [6])

NFV -Network Function Virtualization; EM - Element Manager; MANO - Management and Orchestration (NFVO – NFV Orchestration; VNFM – VNF Manager; VIM Virtual Infrastructure Manager); VNF/PNF – Virtual/Physical Network Function; NFVI -NFV Infrastructure; NS-Network Service; OSS-Operations Support System.

Figure 1 shows the ETSI NFV architecture [10], to which several new functional blocks are added in order to support the network slicing (ETSI-NFV EVE 012 [6]).

The 3GPP TR 28.801 document [7] identifies three new management functions: *Communication Service Management Function* (CSMF) – it translates the communication service requirements to NSL requirements; *Network Slice Management Function* (NSMF) - responsible for the management (including lifecycle) of NSLIs (it derives network slice subnet requirements from the network slice related requirements); *orchestration* system.

Network Slice Subnet Management Function (NSSMF) - responsible for the management (including lifecycle) of *Network Slice Subnet Instances* (NSSIs). The Os-Ma-NFVO Reference Point (RP) is the interface with ETSI NFV-MANO. To interface properly with NFV-MANO, the NSMF and/or NSSMF need to determine the type of NS or set of NSs, VNF and PNF that can support the resource requirements for a NSLI or NSSI, and whether new instances of these NSs, VNFs, and the connectivity to the PNFs, need to be created, or existing instances can be re-used.

B. *5GPPP slicing architecture*

The 5GPPP Working Group proposes in [1] a slicing architecture having four planes: *Service, Management and Orchestration, Control and Data plane*. The Service plane comprises the *Business Support Systems* (BSSs) and business-level Policy and Decision functions, as well as applications and services operated by the tenant. Note that this plane includes an *end-to-end orchestration* system.

 ISBN: 978-1-61208-721-4

The *Management* and *Orchestration plane* includes several functional blocks: *Service Management*, *Software-Defined Mobile Network Orchestrator* (SDMO), and *NFV managers* like VIM and VNFM.

The SDMO is composed of a *domain specific application management*, an *Inter-slice Resource Broker*, and *NFV-NFVO*. The SDMO performs the E2E management of network services; it can set up slices by using the network slice templates and merge them properly at the described multiplexing point. The Service Management intermediates between the service layer and the Inter-slice Broker; it transforms consumer-facing service descriptions into resource-facing service descriptions and vice versa. The Inter-slice Broker handles cross-slice resource allocation. The domain-specific application management functions could be, e.g., for 3GPP: Element Managers (EM) and Network Management (NM) functions, including Network (Sub-) Slice Management Function.

The *Control plane* includes two types of SDN –type controllers: *Software-Defined Mobile Network Coordinator* (SDM-X) and *Software-Defined Mobile Network Controller* (SDM-C), as well as other control applications. The SDM-C and SDM-X take care of dedicated and shared NFs respectively. Following the SDN principles, they translate decisions of the control applications into commands to VNFs and PNFs. Note that SDM-X and SDM-C, as well as other control applications, can be implemented as VNFs or PNFs.

The *Data plane* comprises the VNFs and PNFs needed to carry and process the user data traffic.

The architecture also includes a *Multi-Domain Network Operating System* containing different adaptors and network abstractions above the networks and clouds heterogeneous fabrics. It is responsible for allocation of (virtual) network resources and maintains network state to ensure network reliability in a multi domain environment.

C. ETSI Multi-domain Multi-tenant slicing architecture – example 1

Figure 2 shows (adapted from ETSI GR NFV-EVE 012 [6] and J.Ordonez-Lucena et. al. [3][18]) a multi-domain slicing architecture, viewed at run-time phase. A given slice instance can span several *Infrastructure Providers* (InP) and/or administrative domains.

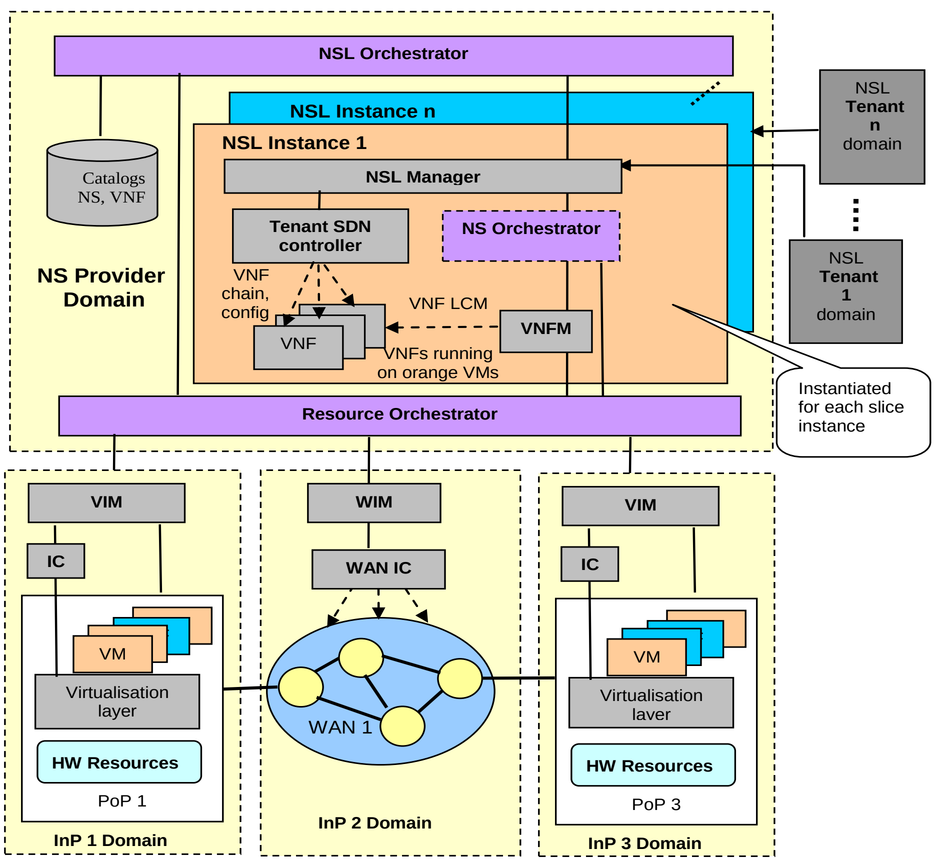

Figure 2. Run-time image of a multi-domain slicing architecture - example 1 (adapted from ETSI GR NFV-EVE 012 [6] and Ordonez-Lucena [3][18])

NS – Network Service; NSL - Network Slice; VNF – Virtualized Network Function; VNFM – VNF Manager; SDN Software Defined Networking; LCM –Life Cycle Management; VIM – Virtual Infrastructure Manager; WIM – WAN Infrastructure Manager; IC- Infrastructure SDN controller; HW- Hardware; WAN – Wide Area Network

The architecture is still high–level depicted, e.g., the NSL orchestrator and Resource Orchestrator are multi-domain capable, but not detailed. Note also that (for simplicity sake), this architectural picture focuses on the transport and core network domains, omitting the RAN (Radio Access Network) domain.

The NSL provider can simultaneously operate multiple NSLIs, which run on top of a common infrastructure, spanning across multiple administrative domains and each belonging to a different infrastructure provider. The NSL provider, taking the role of an infrastructure tenant, rents the infrastructure resources owned by the underlying InPs and uses them to provision the NSLIs.

The NSL provider has a *Resource Orchestration* (RO) functional block. The (RO) uses the finite set of resources that are at its disposal (the resources are supplied by the underlying VIMs/WIMs) and dispatches them to the NSLIs in an optimal way. All the NSLIs receive the resources needed to satisfy their (potentially different) requirements, while preserving their performance isolation. Note that RO should have information on resource availability in each domain, and this supposes a set of inter-domain interactions. The work [18] does not specify the implementation of the RO (distributed or centralized).

Each NSLI has its own management plane, (to assure isolation across NSLIs), consisting of four functional blocks: *NSL Manager, NS Orchestrator (NSO), Tenant SDN Controller*, and *VNF Manager (VNFM).* The VNFM(s) and the NSO perform the required life cycle operations (e.g., instantiation, scaling, termination, etc.) over the instances of the VNFs and NS(s), respectively. Interactions between these functional blocks and the RO are necessary.

The NSL Manager coordinates the operations and management data from Tenant SDN Controller and the NS Orchestrator, performing the fault, configuration, accounting, performance, and security management within the NSLI. Each tenant consumes its NSLI and operates it at its convenience (within the limits agreed with the NSL provider) through the NSL Manager.

The *NSL Orchestrator* (NSLO) is the highest layer of the architecture, *having a key role in the creation phase* and in the run-time phase. In the creation phase, NSLO receives the order to deploy an NSL instance for a tenant (or the Slice Provider decides to construct a slice). The NSLO has enough information (including on multi-domain) as to check the order feasibility and, if feasible, then triggers the instantiation of the NSL. To accomplish this, it interacts with RO, and accesses the VNF and NS Catalogues. These catalogues contain VNF and NS descriptors, exposing the capabilities of all the VNFs and services that an NSL provider can select for the NSLs.

At run-time, the NSLO performs policy-based inter-slice operations, e.g., it analyses the performance and fault management data, received from the operative NSL instances, to manage their Service Level Agreements (SLAs). In case of SLA violations, the NSLO decides which NSL instances need to be modified, and sends corrective management actions (e.g., scaling, healing, etc.) to their NSL Managers.

D. Multi-domain Multi-tenant Architecture -example 2

Creation of slices across a federated environment is a complex task, in terms of slice decomposition (per-domain), both in the construction phase and (to assure the performance maintenance) in running phase. T.Taleb, I.Afolabi et.al., recently proposed in [17] a hierarchical multi-domain orchestration architecture (see adapted Figure 3). They introduced a *Multi-domain Service Conductor* (MSC) stratum, to perform service management across federated domains. The MSC analyses and maps the service requirements of incoming slice requests onto appropriate administrative domains and maintains the desired service performance during service lifecycle. Below MSC, a *Cross-domain Slice Coordinator* is defined for each slice, which aligns cloud and networking resources across federated domains and carries out the (LCM) operations of a multi-domain slice. It also establishes and controls inter-domain transport layer connectivity, assuring the desired performance.

In [17], a multi-domain NSLI can combine several *Fully-Fledged NSIs* that belong to distinct administrative domains, to get an E2E multi-domain (i.e., federated) NSLI. The constituent Fully-Fledged NSIs, instantiated in different administrative domains, can be called NSSI of the multi-domain NSLI.

When a slice request is received from a 3rd party, a sequence of actions will be performed [17]: 1) mapping of the service requirements onto capability requirements; 2) translating the capability requirements into: a. NSLI resource requirements (compute, storage, networking); b. NSLI topology and connectivity type, policy, isolation and security requirements; 3) identifying the infrastructure-domains with the required resources, able to assure the E2E NSLI functional and operational requirements; 4) instantiating NSSIs in each infrastructure domain and then "stitching" them to create the federated NSLI; 5) run-time coordination management operations across different domains for maintaining the E2E NSLI service integrity.

The architecture introduces (at top level) a novel plane - *Service Broker* (SB) to handle incoming slice requests from verticals, *Mobile Virtual Network Operators* (MVNO), and application providers.

The main SB operation are: NS *admission control* and negotiation, considering service aspects; management of slice user/owner relationship enabling a direct tenant interface with the MSC plane; billing and charging; NSLI scheduling, i.e., start and termination time related with slice composition and decommission.

The SB collects abstracted service capability information regarding different administrative domains, creating a global service support repository. The SB interacts with the Operating/Business Support System (OSS/BSS) in order to collect business, policy, and administrative information, when handling slice requests. Each domain will have a dedicated Orchestration plane, involving the NFV style architecture for the management and control.

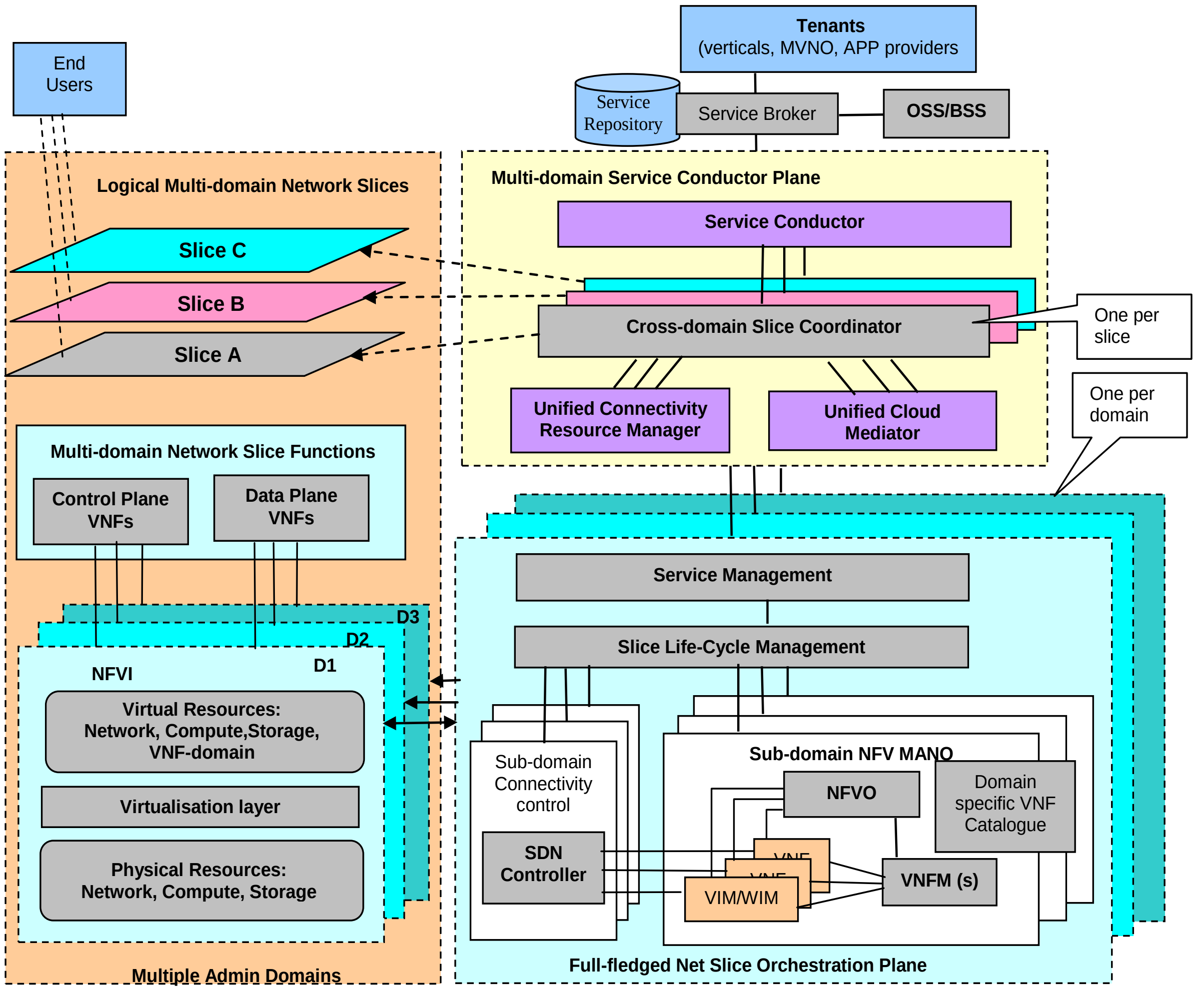

Figure 3. Multi-domain Multi-tenant slicing architecture example 2 (adapted from [17])

IV. NETWORK SLICING DESIGN AND PREPARATION

The design and preparation activities are very important in slicing technology. Among others, catalogues of available services and resources must be constructed in advance to slice instantiation, usable by the tenants in order to select a slice model fitted to their needs.

The general steps performed by the management and orchestration (i.e., higher layers in the architecture examples 1 and 2: NSLO and RO in Example 1; Service Broker and Service Conductor in Example 2) for a slice instance creation are [18]: a. Service ordering; b. Network slice resource description; c. Admission control; d. Optimization and Resource Reservation; e. Network slice preparation.

Service ordering: the NSL provider should construct a *Service Catalogue* (business-driven), containing for each service a *service template*, i.e., a framework to specify the service offering. The Catalogue contains NSLs specifications optimized for different usage scenarios: general 5G services like enhanced Mobile Broadband (eMBB), massive Machine Type Communications (mMTC), and ultra Reliable Low Latency Communications (uRLLC), or vertical-specific applications. A service template includes all information required to drive the deployment of an NSL, e.g.: the NSL (technology-agnostic) *topology*, NSL network requirements (functional, performance, security), temporal, *geolocation* and other operational requirements [18]. The NSL provider offers APIs to tenants, to express their needs and giving them access to the Service Catalogue, where the tenant can select the service template that best matches its requirements. Some parameters and attributes can be customized by the tenant. The result of this dialogue is a catalogue driven NSL *service order* containing information to be mapped on RAN, transport, and core network domains). The Slice Orchestrator entity of the system should process such information.

Network Slice Resource Description: this step creates a resource-centric view of the ordered NSL. Different levels of implementations (NSL-IL) can be defined (for a higher flexibility and better adaptation to the tenant needs) [18]. The NSLO extracts the relevant content from a resource

 ISBN: 978-1-61208-721-4

viewpoint (e.g., the *NSL topology*, NSL network requirements) and constructs an NSL-IL for the NSL instance, i.e.: the NSL topology serves to identify which NS(s) need to be deployed for the NSL, retrieving the corresponding NS descriptor(s) from the NS Catalogue; the deployment option is selected for each descriptor (*NS descriptor ID, NS FlavorID, NS-IL ID*), that best matches the features and the performance level required for the NSL; a NSL-IL is constructed by referencing the selected triplet(s).

Admission Control: the target NSL-IL specifies the resources needed for the tenant's demands. Now, an *admission control* (AC) will be enforced on the ordered NSL-IL, from a resource viewpoint, to decide acceptance /rejection for deployment. Several types of information are needed [18] in this process: (1) the NSL instance resource requirements (resources to be allocated for each VNF instance and virtual link, affinity/anti-affinity rules applicable between VNF instances, reliability requirements for each VNF instance and virtual link; (2) the geographical region(s) where each VNF is needed; (3) the time intervals for activation of the NSL instance; (4) information of the PoPs (Points of Presence) (and the WAN network(s) connecting them) to which the NSL provider is subscribed. Such information is available partially at the NSL orchestrator and partially at RO; therefore, these two functional blocks need to cooperate within the AC acting.

Optimization and *Resource Reservation*: if several variants of NSL-ILs are found feasible by the AC, then RO can run an algorithm to select an optimal solution (note that this is a multi-criteria optimization problem). Afterwards, RO may proceed with resource reservation; it sends resource reservation requests to the underlying VIM(s)/WIM(s). The hard and soft nature of this reservation depends on the use case and NSL provider's policies.

Network Slice Preparation: this is the last step prior to get an operational NSL. It consists of setting up all that is required to manage the NSLI throughout its life cycle, i.e., from commissioning (instantiation, configuration, and activation) to decommissioning (de-activation and termination) (see 3GPP TS 28.801 V.15.1.0 [7]). It comprises preparation of the network environment; designing and on-boarding the NSL descriptor.

For the preparation of the network environment, the NSL Orchestrator performs (see Figure 2) the following tasks:

- negotiation with RO a priority level for the NSLI this allows the RO to manage the cases when the NSL instances compete for the same resources, or the case of lack of enough resources.
- instantiation of the management plane of the NSLI (NSL Manager, Tenant SDN Controller, NS Orchestrator, VNFM(s)); it configures these functional blocks, making them ready for the run-time phase.

In parallel to the network environment preparation, the NSL Orchestrator builds up the NSL descriptor, which is a deployment template used by the NSL Manager to operate the NSLI during its life cycle. This descriptor includes the following parts: a set of policy-based workflows; the set of NSL-ILs available for use, constructed in the Network Slice Resource Description phase; VNF configuration primitives at application level and VNF chaining management instructions; information about management data, used for performance management.

V. Collecting Topology Information in a Multi-domain Environment

The focus and contribution of this section is on inter-domain topology discovery and collection of information by a multi-domain capable slice orchestrator (MDSO), in the design and preparation phase of a slice. Given the large number of variants of multi-domain slicing architecture, we consider, in this section, a generic MDSO; it could be equivalent to the NSL Orchestrator in Figure 2 or Multi-domain Service Conductor in Figure 3. The MDSO could belong to a separate third-party business entity (call it Slice Provider), or each administrative domain can be a slice-provider capable (i.e., each domain owns a MDSO). In the last case, peering relationships may exist between the different orchestrators.

The (MDSO) needs to obtain (among others) topological inter-domain information, in order to be able to perform all steps of the design and preparation phase, including the admission control and split of the multi-domain NSL among the respective participating domains.

Usually, the administrative independent domains are not willing to disclose their detailed topology and resource information to third parties. Therefore, a solution based on an abstract summary *overlay network topology* (ONT) [19] could solve the problem. Inside MDSO, a functional *Inter-domain Peering* block can be defined, to provide an *ONT-service* (ONTS), i.e., deliver information on inter-domain topology graph, inter-domain link capacities, etc.

It is supposed here that a given MDSO is the initiator, which plans a new target NSL template (at a specific tenant request or following the Slice Provider initiative). The initiator MDSO obtains (from ONTS) the ONT information. The ONT should be sufficiently rich to span the geographical area of the required NSL.

The objectives of the MDSO, with respect to network topology, are: to get ONT; determine the domains candidates to participate to the required NSL; get information on inter-domain (links) resources; possibly - apply a constrained inter-domain routing algorithm (upon the ONT acquired from ONTS), with an appropriate metric depending on the target NSL characteristics. Using this information, the MDSO can split the multi-domain NSL among the domains.

Figure 4 shows a generic example of a tentative multi-domain NSL (for simplicity the NSL covers only the core network and not the access networks (AN). The required NSL-0 should contain some domains, e.g., D1, D2, D7, D8. The identities of these domains result from the edge specification of the NSL wanted. However, to get a contiguous topology, possible some other transit domains should be involved (e.g., D4, D5), so the final slice will be NSL-1.

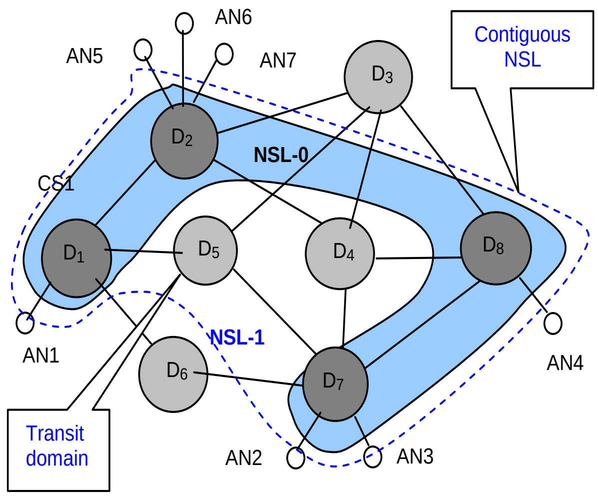

Figure 4. Multi-domain topology of a tentative slice

Depending on the characteristics of the tentative NSL, appropriate routes can be computed (with a specific metric on inter-domain links). For instance, between AN1and AN4, one may have the paths D1, D2, D4, D8 or D1, D5, D7, D8.

If some QoS requirements are imposed in this slice, then resource reservation (see section III) can be done on the inter-domain links.

The sequence of MDSO actions related to collection of information on topology is shortly described below.

The tenant/Slice Provider issues to an initiator MDSO an *NSL_request* (this could be mapped onto a given QoS class) for a target NSL-0.

The initiator MDSO:

1. obtains from ONTS the inter-domain level ONT (topology graph, inter-domain link capacities, etc.). The ONT is sufficiently rich to cover the required NSL.
2. Determines the involved domains in NSL-0 by using the border ingress-egress points knowledge (actually border routers addresses) indicated in the *NSL_request.*
3. Determines a contiguous inter-domain connectivity graph (each domain is abstracted as a node) resulting in an extended NSL-1 (represented by dotted line in Figure 4). In NSL-1 graph, some additional transit core network domains need to be included, e.g., D_4, D_5. Therefore, a contiguous new NSL-1 is defined. Optimization techniques can be applied in this phase.
4. Can run a constrained routing algorithm to determine inter-domain best paths.
5. Can make the first split of the initial NSL among core network domains. This means to produce a set of NSL parameters valid to be requested to each individual domain.
6. Negotiates with each MSDO of a domain, concerning the availability for that part of slice.
7. Run admission control for the overall multi-domain slice.
8. Prepare the networking environment and on–boards the slice template in the catalogue.

VI. CONCLUSIONS AND FUTURE WORK

This paper analyzed several relevant 5G slicing architectures, from the point of view of the management and orchestration functions. It is observed that no unique vision about the scope of orchestration and its hierarchical spilt onto architectural layer exist today.

The second part of the paper has been dedicated to the design and preparation phase of slices. A solution is proposed by this paper related to the problem of inter-domain topology information acquisition and associated actions in a multi-domain context. Note that the solution discussed here only cover a part of information needed by the MDSO initiator in order to split the multi-domain NSL. More complete set of information should be considered (see Section II and III) in order to construct the NSL catalogues. A negotiation dialogue between the initiator MDSO and other MDSOs of peering domain must be performed to provide more information on resources of each domain involved. This will be for future study.

REFERENCES

[1] 5GPPP Architecture Working Group, “View on 5G Architecture”, Version 2.0, December 2017.

[2] NGMN Alliance, “Description of Network Slicing Concept, NGMN 5G P1 Requirements & Architecture, Work Stream End-to-End Architecture”, Version 1.0, Jan. 2016.

[3] J. Ordonez-Lucena, et al., “Network Slicing for 5G with SDN/NFV: Concepts, Architectures and Challenges”, IEEE Communications Magazine, 2017, Citation information: DOI 10.1109/MCOM.2017.1600935.

[4] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, “Network Slicing in 5G: Survey and Challenges”, IEEE Communications Magazine, May 2017, pp.94-100

[5] P. Rost, et al., "Network Slicing to Enable Scalability and Flexibility in 5G Mobile Networks", IEEE Communications Magazine, Vol.55, May 2017, pp.72-79.

[6] ETSI GR NFV-EVE 012 V3.1.1 (2017-12), Release 3 “NFV Evolution and Ecosystem; Report on Network Slicing Support with ETSI NFV Architecture Framework”.

[7] 3GPP TR 28.801 (V15.0.0) (09-2017): "Telecommunication management; Study on management and orchestration of network slicing for next generation network".

[8] 3GPP TS 28.530, “Management of 5G networks and network slicing; Concepts, use cases and requirements”, Rel.15, April2018.

[9] A. Galis “Network Slicing- A holistic architectural approach, orchestration and management with applicability in mobile and fixed networks and clouds”, http://discovery.ucl.ac.uk/10051374/ [retrieved: April, 2019].

[10] ETSI GS NFV 002 v1.2.1 2014-12, “NFV Architectural Framework”.

[11] ETSI GS NFV-IFA 009 V1.1.1 (2016-07) “Network Functions Virtualisation (NFV); Management and Orchestration; Report on Architectural Options”. Technical Report.

[12] ETSI GR NFV-IFA 028 V3.1.1 (2018-01): “Network Functions Virtualisation (NFV) Release 3; Management and

 ISBN: 978-1-61208-721-4

Orchestration; Report on architecture options to support multiple administrative domains” Technical Report.

[13] ONF TR-526, “Applying SDN Architecture to 5G Slicing”, April 2016.

[14] K.Katsalis, N.Nikaein, and A.Edmonds, "Multi-Domain Orchestration for NFV: Challenges and Research Directions", 2016 15th Int'l Conf. on Ubiquitous Computing and Communications and International Symposium on Cyberspace and Security (IUCC-CSS), DOI: 10.1109/IUCC-CSS.2016.034, ttps://ieeexplore.ieee.org/document/7828601

[15] N. F. Saraiva de Sousa, D. A. Lachos Perez, R. V. Rosa, M. A. S. Santos, and C. E. Rothenberg, "Network Service Orchestration: A Survey", 2018, https://arxiv.org/abs/1803.06596 [retrieved: April, 2019].

[16] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck, “Network Slicing & Softwarization: A Survey on Principles, Enabling Technologies & Solutions”, IEEE Communications Surveys & Tutorials, Mar. 2018.

[17] T. Taleb, I. Afolabi, K. Samdanis, and F. Z. Yousaf, "On Multi-domain Network Slicing Orchestration Architecture & Federated Resource Control", http://mosaic-lab.org/uploads/papers/3f772f2d-9e0f-4329-9298-aae4ef8ded65.pdf [retrieved: April, 2019]

[18] J. Ordonez-Lucena, et al., "The Creation Phase in Network Slicing: From a Service Order to an Operative Network Slice", European Conference on Networks and Communications (EuCNC), 2018, https://arxiv.org/abs/1804.09642 [retrieved: April, 2019]

[19] F. Verdi, M. F. Magalhaes, “Using Virtualization to Provide Interdomain QoS-enabled Routing”, Journal of Networks, April 2007, pp. 23-32.