Computer Communications 149 (2020) 232–240

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

On 5G network slice modelling: Service-, resource-, or deployment-driven?

Apostolos Papageorgiou [a,*], Adriana Fernández-Fernández [a], Shuaib Siddiqui [a], Gino Carrozzo [b]

[a] i2CAT Foundation, Software Networks group, Gran Capitá 2, 08034, Barcelona, Spain
[b] Nextworks s.r.l., R&D department, Via Livornese 1027, 56122, Pisa, Italy

ARTICLE INFO

Keywords:
5G
Slicing
NFV
Data modelling

ABSTRACT

Network slicing has been one of the hottest topics in standardization recently, as telecom operators are intensively investigating its usage for operating isolated and independently configurable logical networks, in order to ease and promote the network sharing and neutral hosting business. However, extensive deployments of slice management solutions are often impeded by incompatibilities of the used slice data models, which stem from different definitions and modelling approaches for the slicing concept, e.g., some driven by 3GPP standards, others by other standards or proprietary solutions, and so on. Although various studies on slicing have been performed, none of them has focused on slice data modelling across research and standards. Incompatible slice models do not only limit interoperability but they also reduce the efficiency of network slicing systems. This paper lays a foundation towards more efficient and interoperable network slice modelling by methodically investigating, categorizing, and formally describing core slice modelling approaches, including new modelling suggestions. Subsequently, we analyse their advantages and disadvantages and we propose slice model quality metrics, which we use for performing a case study on our testbed.

1. Introduction

Network slicing was introduced in the Software-Defined Networking (SDN) domain as a way to ensure that "*actions in one slice do not negatively affect other slices*" [1] and spread to the entire telecom architectures domain as "*a set of network functions, and resources to run these network functions, forming a complete instantiated logical network*" [2]. In the meantime, it has been blamed to be many things, ranging from being just a buzzword to being a means for telecom operators to bypass net neutrality. Luckily, telecom companies and researchers, as well as standardization organizations, have been converging to a common understanding of what a slice is, usually defining it as something similar to an isolated and individually manageable and configurable set of network infrastructure resources and (physical or virtual) network functions of standard network architectures that are deployed on them. In 5G, operating diversified slices upon the same physical infrastructure will help to satisfy requirements of different verticals with reduced costs [3].

However, even if all implementers ever adhere to the same network slicing definitions, some business models boosted by network slicing (e.g., network sharing [4], neutral hosting [5]) will not grow dramatically until common models are used. A universal end-to-end network slice data model (i.e., one that covers in detail all parts that can exist in a slice) does not exist and is very unlikely to appear, especially for slices that span across different types of network technologies. This is not only because each organization is concerned with its own part of the huge landscape of telecom technologies, but also because it might be more efficient for implementers to just use selected data models with smaller scope, choosing and combining them depending on the nature of the slice they want to implement. However, in some cases, remarkably at the level of management and orchestration (MANO) in 5G systems, it is important to have a core slice model that connects the main network slice ingredients, even when they span across different types of networks. It is indeed in the context of management and orchestration that 3GPP has defined a template for network slices, IETF groups have worked on YANG modules for network slices, ETSI (European Telecommunication Standards Institute) has tried to map slice models to its Network Function Virtualization (NFV) descriptors, and various research works have proposed different solutions for end-to-end slicing and modelling. While the details of these works will be explored in the next section, it can be safely argued that widely accepted MANO-layer slice data models are needed, but not out there yet.

The problem addressed in this paper is about how to model and inter-relate the ingredients of a network slice and which modelling decisions are preferable depending on the circumstances or the desired features of the slicing solution. By answering these questions, the goal of this paper is to pave the way towards standardized MANO-layer network slice representations, which focus on the slice structure and

* Corresponding author.
E-mail addresses: apostolos.papageorgiou@i2cat.net (A. Papageorgiou), adriana.fernandez@i2cat.net (A. Fernández-Fernández), shuaib.siddiqui@i2cat.net (S. Siddiqui), g.carrozzo@nextworks.it (G. Carrozzo).

https://doi.org/10.1016/j.comcom.2019.10.024
Received 25 June 2019; Received in revised form 30 September 2019; Accepted 22 October 2019
Available online 25 October 2019
0140-3664/© 2019 Elsevier B.V. All rights reserved.

can be combined with lower-level data models for specialized slice ingredients such as specific access slicing models (e.g., for LTE) or VIM (Virtual Infrastructure Manager) models (e.g., OpenStack projects).

To this end, we explore related attempts (cf. Section 2), identify three classes of approaches for slice modelling (including one which is practically our own suggestion, stemming from our development activities for the 5GCity project), describe them all in a uniform UML-based way (cf. Section 3), analyse their features, advantages, and disadvantages (cf. Section 4), and evaluate them using slicing-specific metrics in a Case Study empowered by our network slicing testbed, deployed in three cities (cf. Section 5).

This paper has a threefold contribution, which consists of (i) a systematic classification of slice data modelling approaches, (ii) the design and specification of the deployment-driven modelling approach, and (iii) the derivation of appropriate slice data model metrics, followed by a Case Study which uses these metrics and our own testbed in order to evaluate the main generic slice modelling approaches. Our findings can be summarized in that network slice models can be centred around (i) network services, in order to maximize NFV alignment, (ii) network resources, in order to be Cloud-native, or (iii) deployable components, in order to address a bigger number of deployment technologies. Although each of the approaches has disadvantages, data models of any of the three approaches should be able to capture the information required for slice management, if developed correctly.

2. Background and related work

Along with an understanding of what a network slice is, which was explained in the introduction and will be understood in more detail in the next subsections, it is necessary to understand the basic virtualization and modelling concepts and technologies involved in slicing (cf. 2.1), as well the state of the art related to slice data modelling, which consists primarily of three lines of work, namely *slice definitions*, *slice data models*, and *IT/Network infrastructure models*, each of them having their own implications (cf. 2.2, 2.3, 2.4, respectively).

2.1. Background on slicing enablers

Network Function Virtualization (NFV): This concept, mainly represented by the ETSI NFV standard (and its reference architecture described in [6]), refers to the software-based implementation of telecom network functions that were previously implemented on legacy networking devices or servers. Once virtualized (i.e., implemented as software), these functions are called VNFs (Virtual Network Functions). Typical examples are the elements of the 3GPP packet core, firewalls, Customer Premises Equipment (CPE) such as home gateways, and more. However, once virtualized, these VNFs become part of service chains which may involve any kind of functionality which is directly or indirectly related to networking aspects. This means that NFV has come to incorporate various scenarios in which the VNFs are not anymore exclusively strictly network functions, but potentially also parts of applications that are involved in NFV service chains, which can be more or less anything, varying from a video processing server to an image recognition component (as, for example, in [7]), as long as it is modelled and orchestrated based on NFV specifications. **In the context of 5G slicing**, NFV is a core enabler in the sense that it can be used to create and orchestrate the network functions (and service chains) that will implement the functionality of a network slice. When it comes to modelling, this means that (ETSI) NFV data models for services, functions, and their running instances, will normally be part of the slice model.

Virtual resources and 5G network resources: Virtualization refers to creating a view of the resources of a system (computing, storage, and network) which differs from the actual physical hardware that provides them (i.e., servers/CPUs, memory, and networking equipment, respectively) but allows us to use and manage them in a similar way. The virtualized view is typically much more fine-granular and dynamic than the physical view and is provided by virtualization middleware, which is different for each type of resource. Virtual resources mainly refer to computing and storage, as well as the network interfaces of and the logical networks between the devices that provide these resources, and is often performed in the scope of (inter-networked) data centres [8]. 5G network resources is a broader term, which we use here to refer to the rest of the telecommunication resources, notably the access network resources, be it 3GPP radio, WiFi, or similar, as well as resources of the base stations, physical 5G core functions, and their interconnections. **In the context of 5G slicing**, virtual and 5G resources are necessary elements for the creation of slices, as long as they are "sliceable", i.e., individually partitionable with the ability to be assigned to different slices. For example, a 5G radio resource might be sliceable or not, depending on the existence (or not) of sophisticated RAN (Radio Access Network) controllers behind them. Similarly, virtual resources need to be partitionable, e.g., via the usage of quotas or other types of multi-tenancy. Given this "sliceability", virtual and 5G resources provide the ingredients which – together with the NFV services on top of them – will determine the scope and functionality of a slice.

Resource data models: In order to be managed by higher-layer management entities and orchestrators, the resources mentioned in the previous paragraph are represented using sets and hierarchies of information models and data models. This paper focuses mainly on data models, while a good explanation of the relationship between the two is provided in [9]. Such data models can be found in different domains (e.g., software engineering, telecommunications, computer systems), developed by different standardization bodies (e.g., 3GPP, IETF, IEEE, ETSI), and realized with different technologies (e.g., YANG, XML, JSON, YAML). They usually capture single entities, such as "LTE radio", "virtual compute", and so on. **In the context of 5G slicing**, these data models have to be adjusted and combined in order to represent slice parts. However, a complete representation of a slice, i.e. one that can enable managing all aspects of its lifecycle as defined in related standards documents and research solutions, is expected to require additional data models or data model parts, developed specifically for the slicing landscape.

2.2. Works related to slice definitions

Slice definitions *imply* slice models and end-to-end slicing frameworks *require* slice models. More than this, by defining or describing the slice functionality and lifecycle in a given manner, they often assume the existence of specific slice model parameters. *However*, they do not provide concrete models or modelling solutions that can be used either for interoperability or as "off-the-shelf" slice representation templates.

One of the most prominent slice definitions was contributed by NGMN (Next Generation Mobile Networks) [2]. For example, by specifying that a "Sub-network instance" can belong to more than one slice, this definition dictates that slice-specific characteristics such as concrete VLAN tags that characterize the traffic of a slice or wireless channels that are used exclusively by a slice cannot be modelled within the sub-network entities. Since this definition is very high-level, one could refer to more detailed descriptions of end-to-end slicing approaches in order to derive more such assumptions.

For example, [10] seems to dictate the explicit modelling of physical nodes and physical links (inter-related to virtual nodes and virtual links), while it also requires slice isolation to a degree that would probably forbid sharing certain resources among slices (e.g., again, VLAN tags or concrete wireless channels). On a higher layer, [11] foresees the existence of Virtual Network Application (VNA) descriptors as elements of network slices, in addition to the NFV-specified descriptors. This would turn the components of a VNA into additional ingredients of the overall slice data model.

In a more specific scenario, namely in order to address the resource allocation problem in the context of network slices, [12] introduces some relationships between involved high-level entities (Slice owner, "micro-slices", Cloud controllers etc.), as well as some slice parameters (slice traffic volume, traffic-to-processing ratio, etc.). The surveys of Afolabi et al. [13] and Foukas et al. [14] can also serve as valuable sources for deriving further such modelling directives.

2.3. Works related to slice data models

Most of the works that get down to suggesting concrete data modelling approaches belong to the standardization domain. To date, significant attempts in the direction of standardizing slice models have appeared in 3GPP, ETSI, and IETF. *However*, as it will become clear in Section 3, the respective contributions are based on different principles, they are often biased by the focus of the developing organization, and none of them has dominated the area yet. They often address only parts of what should be an end-to-end slice, while they are also "work in progress", e.g., the 3GPP Network Slice Template (NST).

3GPP recently published its study on network slicing (3GPP TR 28.801 [15]), which was followed-up by the specification of slice provisioning requirements and procedures (3GPP TR 28.531 [16]) and the slicing-related 5G Network Resource Model (3GPP TR 28.541 [17]). While [15] foresees the existence of an NST as the description of the structure, contained components, and configuration of a network slice, [16] leaves the concrete definition of NSTs open. However, [17] specifies some possible ingredients for this by providing models for slicing-related entities. Although these models are very detailed with regard to the attributes of 5G core and radio functions, as well as their relationships with slice instances, they rely upon externally linked models (e.g., ETSI NFV-defined Network Service models) for the (virtualized) infrastructure that hosts and interconnects these functions. For example, (Cloud or edge) compute nodes, or physical connections of non-3GPP technologies (fibre, WiFi), are not covered in [17], although they are core ingredients of a fully-manageable end-to-end slice.

According to the ETSI NFV report on network slicing support (ETSI GR NFV-EVE 012 [18]), network slices are mapped to – potentially nested – Network Services. This means that the slice model is practically implicitly defined by the Network Service model, namely the NSD (Network Service Descriptor) [19], in addition with a Network Slice Instance (NSI), which is just used to inter-relate NSDs that belong to the same slice. Although the NSI and the NSD include sufficient information about all the network functions involved, they do not deal with restricting the resources (e.g., compute resources) that belong to a slice, and they are coupled to the network virtualization vision and less focused on the network access part or on the support of "Cloud-native" concepts such as microservices, which could also run on slices.

More technology-agnostic approaches have been followed by the ETSI NGP (Next Generation Protocols) slicing information model [20] and an IETF draft which defines a network slice YANG module [21]. However, [20] is work-in-progress with few details for the detailed slice ingredients, while [21] was moderately supported and has expired, potentially because of the timing and the limited alignment with 3GPP and ETSI. More details about [21] are discussed in Section 3.

Finally, some research works provide concrete parts of potential slice models for specific purposes. For example, [22] contributes proposed "RAN slice" and "Cell slice" models as extensions of the previously discussed 3GPP Network Resource Model. Again on the RAN level but on an even lower layer, [23] discusses different structures for radio frames, notably a frame structure that enables time-frequency multiplexing of users based on their service requirements. Applying the "tiling" scheme presented in that paper would probably imply an extended, fine-granular radio frame data model which would be of big importance for the overall slice model, because each of the radio tiles captured in that model would probably belong to an end-to-end slice.

2.4. Works related to IT infrastructure models

The investigation of the previously discussed models reveals that slices are composed of computing and networking ingredients which have existed long before the emergence of network slicing. Although research is required to efficiently compile these ingredients into slice models, existing models of IT and networking elements used in the Cloud and network management domains can be leveraged as parts of an emerging standard slice model.

An important candidate in this domain is the CIM (Common Information Model) [24]. Among its list of interconnected class diagrams that represent computing and networking entities, there are models for *ComputerSystem*, *LogicalNetwork*, *LANSegment*, and more. Even a CIM-VNE extension (i.e., for Virtual Network Environments) has been designed in the context of the scholar work of [25], thus making the model even more mature to be used for representing slices. Further models for describing the IT infrastructure part of a slice can be derived from [26].

3. Generic slice modelling approaches

In order to find the most appropriate way to model network slices, it is necessary to derive the basic concepts that lie in the core of the existing slicing solutions and data models. By examining the landscape discussed in Section 2, it can be seen that existing models are built either around the concept of the (network) *services* that the slice is meant to support or around the concept of the IT and telecom infrastructure *resources* that compose this (services-hosting) slice. It is also easy to imagine a hybrid approach, in which the two concepts are inter-linked (without one of the two being "hidden" behind the other, as will be explained in the rest of the paper), thus focusing on how services are *deployed* on resources. In line with these basic concepts, three main generic models are identified, namely *service-driven*, *resource-driven*, and *deployment-driven*

Having identified these generic approaches, this work contributes high-level models (described in UML) that represent the skeleton of the three generic models (Figs. 1, 2, 3, and 4). Specific detailed data models that belong to one of these categories should be possible to map to the respective skeleton. The *core concept*, the *model structure*, and prominent *representative solutions* for each of the designed models are explained and discussed in the following subsections.

3.1. Service-driven models

Core concept: Slices are modelled as sets of nested Network Services. Infrastructure resources (e.g. compute, storage, network) belong to a slice only if they play a role in the runtime instances of the Network Services.

Model structure: As shown in Fig. 1, slices simply consist of one or more NSs (Network Services), which in turn consist of VNFs (Virtual Network Functions), PNFs (Physical Network Functions), VirtualLinks, and VNFFGs (VNF Forwarding Graphs), with their relationships as defined by the ETSI NFV standard. The detailed descriptors of the four mentioned NS ingredients contain fields (e.g., "virtualComputeDesc" of the VNF descriptor) which hold information about the (physical or virtual) "hosts". The collection of these fields implicitly specify the total resources of the network slice.

Representative solutions: The ETSI Report on Network Slicing Support [18] is heavily service-driven, including direct mapping of the slice concept to the NS concept. The 5GTango project interprets this as "*making sense to describe a Network Slice as a set of 1 or more NSs*" [27], and it endorses this model in its slice management framework. Finally, the 5G-Americas White Paper on Network Slicing [28] is also aligned with this view, though it does highlight essential differences between slices and NSs and does go into details of slice modelling.

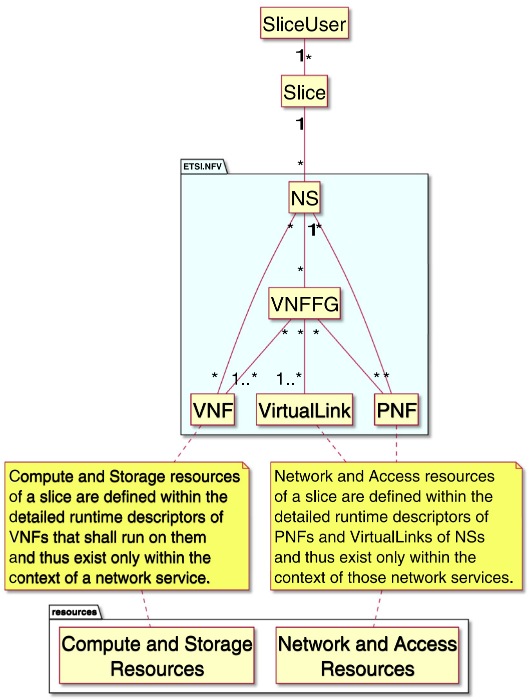

Fig. 1. Service-driven 5G network slice model.

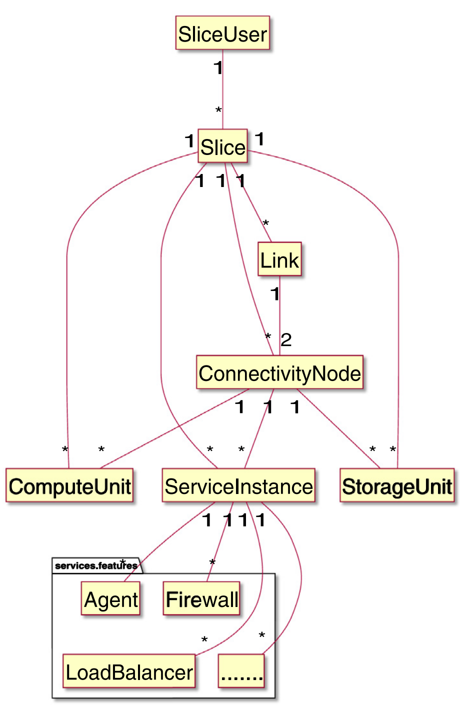

Fig. 2. Resource-driven 5G network slice model.

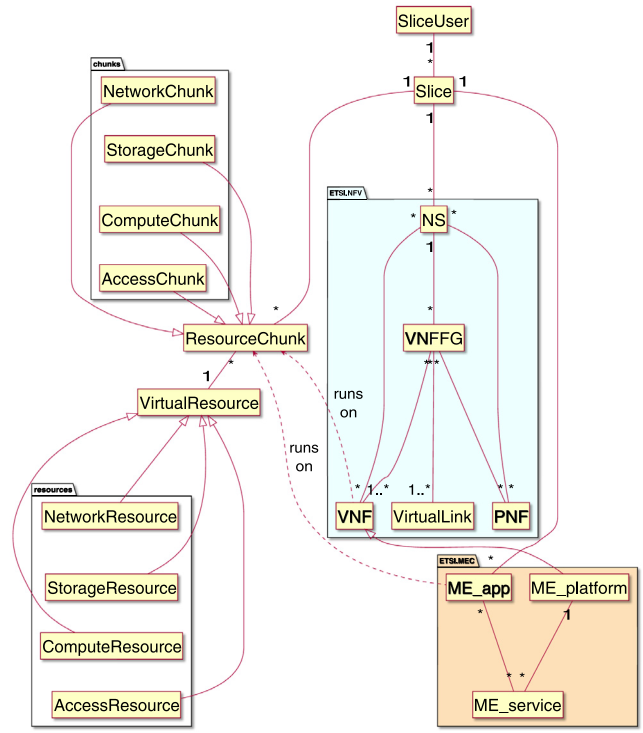

Fig. 3. Deployment-driven 5G network slice model.

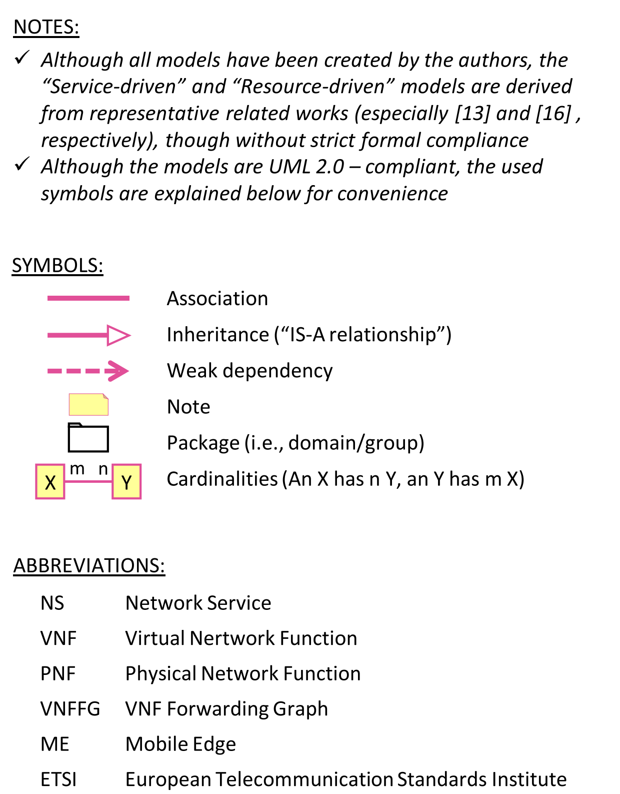

Fig. 4. LEGEND for slice data models.

3.2. Resource-driven models

Core concept: Slices are modelled as sets of infrastructure resources with service elements running on them. This is closer to a Cloud-native approach, since the model can be easily modified or complemented to let the resources host anything other than Network Service elements.

Model structure: As shown in Fig. 2, slices contain connectivity nodes (e.g., switches), which are linked with compute and storage units, as well as service instances. The core structure is very close to "network graphs" that have been typically used in routing and load balancing problems, while also containing service instances that can have various (standard or custom) elements.

Representative solutions: An IETF draft that attempted to develop a standard YANG representation for Network Slices [21] is resource-driven and it practically formed the basis on which we have derived this model. Further, the 5G-Matilda project defines a "Slice Meta-model" [29] that is built around the VIM-controlled elements (compute, storage, network), as well as the components that are placed on them. Further, the mentioned metamodel is explicitly designed to be compatible to Cloud-native applications, which decouples it from ETSI-defined NSs (though it supports them).

3.3. Deployment-driven models

Core concept: Slices are modelled as sets of infrastructure resources with Network Services or other deployables linked to them. Complete Network Service instances, ME application instances, or similar entities, are linked directly as elements of the slice model, while in turn requiring their own ingredients to be mapped to the infrastructure resources of the slice.

Model structure: As shown in Fig. 3, slices consist of a set of chunks, which are parts of compute, storage, core network, or access network resources, and may be paired with each other (e.g., by specifying the VLAN that stitches a compute chunk with an access network chunk). In addition, slices are linked to NSs in a way similar to the service-driven model, with the differences that (i) NSD-contained infrastructure resources must be mapped to the aforementioned chunks, (ii) chunks can exist even without NSs, and (iii) NSs can be replaced with (or accompanied by) similar deployables such as ME applications, which then follow an internal modelling as specified by their own domain.

Representative solutions: This model is a proposal of the current paper, based on related development activities of the 5GCity project [30]. To an extent, it can be understood as a hybrid approach between the two other models, based on the idea of reusing (and being adapted to) standard models of important deployables (such as ETSI-defined NSs) but without being bound to them. Practically, any approach that combines ETSI NFV and ETSI MEC deployments within common, slice-aware management environments (e.g. [31]) could be interpreted as also leaning towards this approach.

4. Analysis of slice modelling approaches

The generic models of Section 3 are here analysed in a way that helps developers and standardization organizations decide how to design specific detailed slice data models.

4.1. Comparison of 5G slice modelling approaches

Each of the approaches has advantages and disadvantages in terms of efficiency, or restrictions that it poses on the systems that use it. These advantages and disadvantages have been investigated with regard to four important aspects that the authors have been faced with during the usage and development of slice models in projects related to network virtualization. These aspects are:

- *Ease of slice lifecycle development*: This aspect refers to the implementation of functionalities such as slice creation, service instantiation on a slice, slice activation and deactivation, and slice deletion. Depending on the used model, implementing such functionalities will require additional logic or depend more on existing orchestrators and tools.
- *Degree of alignment to standards about slicing*: This aspect refers to the compatibility of each model to the specifications related to slicing, which were explored in Section 2.
- *Palette of other related standards supported by the approach*: This aspect refers to the number and variety of specifications that are supported or considered by this model, including important specifications that are not directly related to slicing.
- *Support of network sharing functionality*: This aspect refers to the degree to which the features of each model either support or impede the implementation of ISP (Internet Service Provider) network sharing solutions on top of the slicing system.

The identified advantages and disadvantages of the generic models with regard to these four aspects are discussed in Table 1.

4.2. Towards the evaluation of 5G slice data models

A quantitative evaluation of data models would require the selection of widely accepted data model efficiency metrics, as well as a sufficiently big and representative dataset of slice model instances, ideally for industrially rolled-out and operational network slices. The adoption of slicing, as well as the related research, have not yet reached a state that would meaningfully enable such an evaluation. Therefore, in this paper, the models will be evaluated based on a case study of limited scope, performed upon the slicing testbed of our 5GCity project (cf. Acknowledgement).

However, the first step towards preparing the ground for such case studies (or later even for quantitative slice model evaluations), is the definition of appropriate *slice data model quality metrics*. Such metrics can be derived by examining the generic *data model quality metrics* of [32] and interpreting them in the context of slice data modelling. Similar (or additional) metrics can be found in [33] and [34]. Specifically, 31 metrics (belonging to 8 different categories, i.e., quality factors) identified in [32] could be concretized for the case of slice models.

For example, the "number of violations to data modelling standards" could be defined as the number of slice model parameters that present an incompatibility with a parameter that appears in either the ETSI NSD or the 3GPP NST. However, it makes indeed more sense to focus on the few metrics that the survey of [32] found to be most useful and can be deterministically quantified for the case of slice models. This leads to the three metrics of Table 2. The first two should be considered per Use Case, while the third one can also be measured system-wide. Although these metrics will become more valuable when measured upon sufficient datasets of slice model instances that follow different modelling approaches and implement the same (well-defined) Use Cases, this paper uses them already to provide initial insights about the behaviour of the three models in realistic scenarios.

5. Case study

This section explores the values of the three slice data model quality metrics of Section 4 for the three slice modelling approaches discussed in Sections 3 and 4. The system developed in our 5GCity project includes a slice management solution, which is used on top of 5G infrastructure that has been deployed in three European cities, namely Barcelona, Lucca, and Bristol. The infrastructure includes physical resources for the radio part, the edge servers, and the Cloud data centre, as well as their interconnections with switches and Gigabit ethernet links. On top of those, there is a virtualization layer using Openstack along with proprietary software in order to present a sliceable view of the elements. These are the elements that can be used as slice ingredients, namely compute chunks, network chunks, and access chunks. Further, various Use Cases have been analysed and implemented based on NFV concepts, including a video processing Use Case and a 5G neutral hosting Use Case, which will be used in this case study in order to provision slice data model instances. In the following subsections, we go through the questions and expectations, the methodology, the used datasets, the results, and the observations of our Case Study, in line with the typical structure of Case Studies in software engineering (refer also to [35]).

Table 1
Comparison of 5G slice modelling approaches.

	Service-driven	Resource-driven	Deployment-driven
Slice lifecycle development	+ The entire slice lifecycle management can be implemented via OSS/BSS extensions that interact only with the NFVO. – Modelling everything that needs to run on the set of slice resources as a standard NS might be undesired due to complexity of NSD creation, lack of ETSI NFV expertise, or conceptual distance of the deployables with the ETSI NFV standard.	+ Cloud resource orchestrators can be used for the slice resources management, while services can be potentially modelled with less complexity than ETSI-based solutions. – The absence of linking to NS descriptors can lead to duplications or incompatibilities between the slice lifecycle management implementation and the NS lifecycle management implementation.	+ The implementation of the slice resource management can be done outside of the NFVO (as in the resource-driven approach), while services deployed on the slice during its lifecycle are not restricted to be modelled in a specific way. – The conformance of the models of the deployable instances (right part of Fig. 3) with the resources-related part of the slice model (left part of Fig. 3) might be challenging to achieve in complex systems. This refers mainly to correctly modelling the weak dependencies of Fig. 3.
Standards alignment	+ Intuitively close to the 3GPP and ETSI NFV expectations of how a slice data model should look like. – Heavily dependent on the endorsement of ETSI NFV-based modelling of (network) services.	+ Intuitively close to the Cloud-native way of modelling resources (e.g., note the similarity to OpenStack resource types and hierarchy). – Structurally disconnected from other existing standard models.	+ Aligned to Cloud-native standards for the modelling of resources as well as to 3GPP and ETSI for the modelling of services. – Risks incompatibilities with those standards by having the loosest integration between the “services part” and the “resources part”.
Supported standards	+ 100% support of any ETSI NFV-modelled solution. – Support of the deployment of services modelled based on other standards (e.g. ETSI MEC or Cloud-native standards) is possible either with additional efforts or not at all.	+ Supports the inclusion of any kind of services in the slice instance. – It requires its custom service modelling even for services that are already modelled based on other standards.	+ It supports ETSI NFV and ETSI MEC service descriptors, as well as any other standard descriptor that can be linked to its resources representation. – It does not support NSs that describe their (required) “host” resources in a way that cannot be mapped to any of the slice resource chunks.
Network sharing functionality	+ It can be built on top of existing NFVOs. – All the aspects and the phases of the network sharing functionality are dependent on the NFVO and its way of operation.	+ Slice resource sharing is decoupled from runtime constructs (e.g., service instances), so that it can be implemented at a pre-runtime phase and without having to go into the heavyweight details of NS design. – Limited flexibility with regard to quickly building slices by composing off-the-shelf services from NS catalogues.	+ Slice resource sharing is decoupled from runtime constructs (as in the resource-driven approach), while service creation and instantiation can also be facilitated by an NFVO and other NFV-related modules such as NS catalogues. – Higher complexity during operation due to the heterogeneity of the models of the diverse service instances that are deployed on the slice.

5.1. Questions and expectations

The main question is how the complexity, the duplication, and the reuse of slice model instances increase or decrease depending on the used slice modelling approach and the examined scenario (in terms of number and type of slices and services). According to the analysis of Section 4, the expectation is that the deployment-driven approach decreases complexity and increases reuse, while the resource-driven approach is also expected to lead to lower complexities than the service-driven approach since it has a more compact generic model compared NFV-related descriptors, which form the basis of the service-driven approach.

Table 2
Derived slice model quality metrics.

Metric	Definitions
Slice model complexity $C = \sum_{i=1}^{n} e_i + r_i$ (*cf. metric #22 in [32]*)	Where n is the number of slice model instances created for a Use Case, e_i is the number of objects in slice model instance i and r_i is the number of relationships in the slice model instance i.
Duplicate slice parameters $D = \sum_{i=1}^{n} \sum_{k=1}^{P_i} x_k$ (*cf. metric #26 in [32]*)	Where n is the number of slice model instances created for a Use Case, P_i is the number of parameters of slice model instance i, and $x_k = 1$ if $\exists j \neq k$ for which p_j is a copy of p_k (p_y being the yth parameter of slice model instance i), while $x_k = 0$ otherwise.
Slice model reuse $R = \frac{n - n_d}{n} * 100\%$ (*cf. metric #30 in [32]*)	Where n is the number of slice model instances in the system and n_d is the number of discrete slice model instances in the system.

5.2. Methodology

In order to obtain results of the three metrics for the three approaches, the following steps have been performed:

1. The 5GCity platform has been used in order to create the resources-related part of a slice and the JSON representations of all the components needed to capture the ingredients (aka *chunks*) of the resources of a slice have been compiled together. Further, a *high-level slice object* has been created to group all the chunks together. An example segment of a compute chunk is shown in Snippet 1, while an example segment of a network chunk is shown in Snippet 2.
2. The two previously mentioned Use Cases have been implemented as network services according to the ETSI NFV standards suite. This means that Slice Templates, as well as Network Service and Virtual Network Function descriptors (called *NST*, *NSD*, and *VNFD*, respectively) have been developed for them. An example segment of an NSD for the video processing Use Case is shown in Snippet 3.
3. Runtime instances of the aforementioned descriptors (which are called *NSI*, *NSR*, and *VNFR*, respectively) have been captured after deployment of the services and their functions, because these runtime instances are required in order to specify a slice (including resources that it actually uses) in the case of the service-driven approach. An example segment of a VNFR is shown in Snippet 4.

4. "Service instance" parts for the resource-driven approach have been generated by adapting VNF descriptors to this specific model where the concept of service chaining is not defined and the use of NSDs is not foreseen. Therefore, to capture the service elements of the resource-driven model, we remove the information that is only relevant when VNFs are encapsulated and interconnected inside some NSD (e.g. connection points). The resulting element is denoted as *VNFD**.
5. The set of partial data model instances that are required have been put together for each of the approaches and the resulting slice data model instances have been inspected in order to measure the values of the three metrics. This has been done for three different custom scenarios. What this meant concretely for the "implementation" of each of the three approaches is explained in the following paragraph, along with the details of the three scenarios that were investigated.

Snippet 1: Sample part of slice data model instance containing information of a compute chunk


```
{
   "availability_zone":"omega",
   "available_ext_net":true,
   "compute_id":"5d88d3063947c0633c7db451",
   "description":"Compute_Chunk_Video_Processing_Slice",
   "id":"5d88d3073947c0633c7db455",
   "name":"ComputeChunk",
   "requirements":{
      "cpus":{
         "required":10
      },
      "ram":{
         "required":14,
         "units":"GB"
      },
      "storage":{
         "required":40,
         "units":"GB"
      }
   },
   "user_id":"5d88d3053947c0633c7db44d",
   "username":"SliceUser"
}
```


Snippet 2: Sample part of slice data model instance containing information of a network chunk


```
{
   "name":"NetworkChunk",
   "id":"5d88d30a3947c0633c7db45b",
   "physical_network_id":"5d88d3083947c0633c7db458",
   "openstack_project_id":"5d88d3073947c0633c7db455",
   "user_id":"5d88d3053947c0633c7db44d",
   "tag":1357,
   "cidr":"10.10.201.0/24",
   "os_network_id":"c1080236-49e3-4cc0-b0fb-a7d5790ea2d6",
   "os_subnet_id":"e4a39b33-3213-4fd0-86cd-291e602b7c13",
   "requirements":{
      "bandwidth":{
         "required":1000,
         "units":"MB/s"
      }
   }
}
```


Snippet 3: Sample part of slice data model instance containing an NSD


```
{
   "name":"Video_Processing_NS1",
   "short-name":"Video_Processing_NS1",
   "vendor":"vendorA",
   "description":"Video_Processing_NS1",
   "vld":[
      {
         "short-name":"mgmt_net",
         "name":"mgmt_net",
         "mgmt-network":"true",
         "vnfd-connection-point-ref":[
            {
               "vnfd-connection-point-ref":"eth0",
               "member-vnf-index-ref":1,
               "vnfd-id-ref":"48c58e7f-9230-41ce-a81e-102e769832b1"
            },
            {
               "vnfd-connection-point-ref":"eth0",
               "member-vnf-index-ref":2,
               "vnfd-id-ref":"26029032-9e67-4360-b26b-f9d1426a101e"
            }
         ],
         "type":"ELAN",
         "id":"mgmt_net"
      },
      {
         "short-name":"data_net",
         "vim-network-name":"data_net",
         "name":"data_net",
         "mgmt-network":"false",
         "vnfd-connection-point-ref":[
            {
               "vnfd-connection-point-ref":"eth1",
               "member-vnf-index-ref":1,
               "vnfd-id-ref":"48c58e7f-9230-41ce-a81e-102e769832b1"
            },
            {
               "vnfd-connection-point-ref":"eth1",
               "member-vnf-index-ref":2,
               "vnfd-id-ref":"26029032-9e67-4360-b26b-f9d1426a101e"
            }
         ],
         "type":"ELAN",
         "id":"data_net"
      }
   ],
   "constituent-vnfd":[
      {
         "member-vnf-index":1,
         "vnfd-id-ref":"48c58e7f-9230-41ce-a81e-102e769832b1"
      },
      {
         "member-vnf-index":2,
         "vnfd-id-ref":"26029032-9e67-4360-b26b-f9d1426a101e"
      }
   ],
   "version":"1.0",
   "logo":"logo.png",
   "id":"45637a39-57c2-4f92-b197-b7f351885551"
}
```


Snippet 4: Sample part of slice data model instance containing a VNFR


```
{
   "vnfd-id":"121c4457-6922-4dc1-9171-2db8fbaad0ce",
   "additionalParamsForVnf":null,
   "vdur":[
      {
         "status":"ACTIVE",
         "name":"NetSlice1.slice_nsd_1-1-captive_portal_vnfd-VM-1",
         "internal-connection-point":[],
         "interfaces":[
            {
               "name":"eth0",
               "mgmt-interface":true,
               "ns-vld-id":"captive_portal_nsd_vld1",
               "ip-address":"192.168.232.5",
               "mac-address":"fa:16:3e:1c:24:d1",
               "mgmt-vnf":true
            }
         ],
         "vdu-id-ref":"captive_portal_vnfd-VM",
         "status-detailed":null,
         "vim-id":"597b5ec0-6b00-4b51-b1a4-f3aa9d5c9ed8",
         "count-index":0,
         "ip-address":"192.168.232.5",
         "_id":"4dbedf5f-354f-4156-b249-3eedf933c8b8"
      }
   ],
   "vim-account-id":"40ef21a9-5dac-4b06-bed0-495c74b0af54",
   "connection-point":[
      {
         "connection-point-id":null,
         "id":null,
         "name":"eth0"
      }
   ],
   "_admin":{
      "projects_write":[
         "58bf498e-7621-45e7-b8e8-f8eb462d17f9"
      ],
      "projects_read":[
         "58bf498e-7621-45e7-b8e8-f8eb462d17f9"
      ],
      "modified":1569450119.4247904,
      "created":1569450119.4247904
   },
   "ip-address":"192.168.232.5",
   "nsr-id-ref":"44cfb316-75ca-48c6-b073-df6b61160b5a",
   "vnfd-ref":"captive_portal_vnfd",
   "_id":"28ee10ba-3bc7-455d-8026-4e118ad2b109",
   "member-vnf-index-ref":"1",
   "id":"28ee10ba-3bc7-455d-8026-4e118ad2b109",
   "created-time":1569450119.4222054
}
```


5.3. Dataset

The complete data model instance *for one slice* will include different elements depending on which of the three approaches is implemented. Based on our implementation, the following elements were required in each of the three solutions in order to represent a slice (refer to the previous paragraph for terminology and abbreviations):

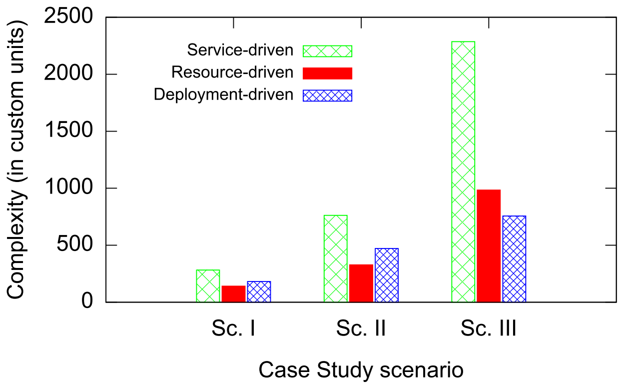

Fig. 5. Complexity of the slice modelling approaches.

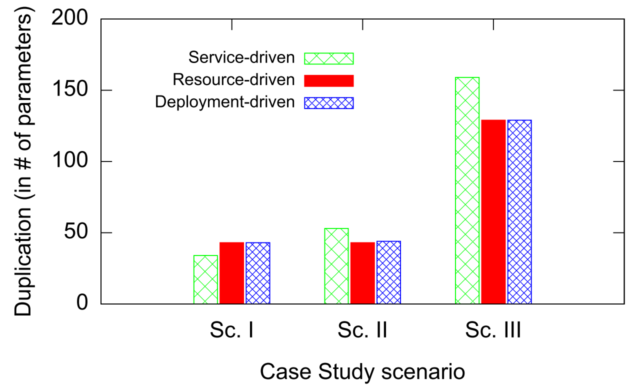

Fig. 6. Duplication of the slice modelling approaches.

- **Service-driven implementation**: The elements required to define the slice were the NSI, the NSR, and the VNFR.
- **Resource-driven implementation**: The elements required to define the slice were the high-level slice object, the chunk objects, and the VNFD*.
- **Deployment-driven implementation**: The elements required to define the slice were the high-level slice object, the chunk objects, the NSD, and the VNFD.

The following three scenarios have been considered:

- **Sc. I**: 1 slice containing compute and network resources as well as 1 network service that has 1 virtual function. This is the simplest scenario and it is meant to provide a basic implementation of the three considered models.
- **Sc. II**: 1 slice containing compute and network resources as well as 2 network services that have 2 virtual functions each, which are interconnected using 2 virtual links. This scenario is based on the implementation and requirements of the video processing Use Case.
- **Sc. III**: 3 slices containing compute and network resources as well as 2 network services per slice that have 2 virtual functions each, which are interconnected using 2 virtual links. This scenario is representative of the 5G neutral hosting Use Case, which allows sharing a common virtualized infrastructure among different service providers by the deployment of multiple isolated slices.

5.4. Results and observations

Using the previously discussed scenarios and datasets, the values for the slice data model quality metrics defined in Table 2 have been computed.

Fig. 5 shows the *Complexity* of each one of the slice data model approaches for the three different scenarios. The Complexity is measured in custom units, namely number of objects and relationships, as defined in 2. As expected, the complexity of the three slice models is directly related to the scenario in terms of number and type of slices and services. For instance, while in Sc. I only one VNF with a single interface is considered in the NS, in Sc. II and Sc. III, the models include more objects and relationships to define and link the two VNFs with two interfaces per NS.

Although the service-driven model does not specify the requirements of resource components that belong to the slice, this approach presents the highest complexity of the three models considered in the performed evaluation. It can be also observed that the resource-driven model has the lowest complexity in the first two scenarios given the absence of linking to NS descriptors and the use of more compact data models to describe the services. Nevertheless, once we increase the number of slices (i.e. Sc. III), the complexity of the resource-driven model exceeds the one of the deployment-driven model. The reason for this is that, while in the resource-driven model the services-related components are within the slice instances, in the deployment-driven model, slices are detached from the services, which allows the reutilization of the same NSD and VNFD for several slice instances.

In Fig. 6, the *Duplication* of each one of the slice data model approaches for the three different scenarios is plotted. For the computation of this metric, we identified the duplicate parameters that are not required for relating different entities within the model. For instance, in the resource-driven and deployment-driven models, in addition to the corresponding identifiers, some other parameters defined as part of the compute and network chunk entities are also included in the slice instance entity. In fact, duplicate parameters for these two models are only present in the resources part of the slice, which explains the equal behaviour of both models across the three different scenarios.

Finally, the *Reuse* metric can be computed only for the Sc. III, where three slices are considered. Moreover, this metric is only achieved by the deployment-driven model, which allows that slices can be defined without being restricted to specific service descriptors. In this way, the NSDs and VNFDs that are included in the deployment-driven slice model can be reused in the three slices considered in this scenario, which accounts for around *36% of the total number of slice data model instances*.

6. Conclusion

In this work, network slice modelling approaches have been categorized and analysed, leading to a series of insights that can help develop more efficient and interoperable slice data models. More concretely, the study of research-driven definitions of slicing and slicing-related standards from 3GPP, ETSI, and IETF has concluded that three main generic approaches for slice modelling can be identified. The first one focuses on the concept of virtual network services and it makes slices an easy fit into the lifecycle of ETSI NFV-compatible systems. The second one focuses more on the IT and networking resources of the slice, thus avoiding restrictions posed by the design and technology of services running on it. Finally, the third one focuses on easing the linking of existing data models of various types of services that can be hosted on a slice.

Furthermore, evaluations of concrete network slice models can be performed either in a custom and qualitative manner based on the comparisons provided in this paper, or they can be designed on the basis of slice model metrics. Such metrics were defined as well, by examining and interpreting important generic data model quality metrics in the context of network slicing. Extensive quantitative slice model evaluations using these metrics are a subject of future work because, firstly, the slice modelling landscape needs to be consolidated and applied in practice and, secondly, sufficient data sets from different slicing Use Cases and management systems need to appear.

However, the proposed metrics have been already used to perform a case study based on models that we developed for our 5GCity testbed and Use Cases. The results give some initial insights about the behaviour of the modelling approaches, namely that the deployment-driven approach tends to increase reuse and reduce complexity, while the results depend also on the size of the Use Cases in terms of slice model instances. Nevertheless, these metrics reflect only a limited set of the advantages and disadvantages of the modelling approaches, which were discussed in the paper. Such advantages and disadvantages should be considered carefully in addition to the metrics before selecting a slice modelling approach. This analysis can be exploited both by implementers of slicing systems and by standardization bodies that work on the definition of slice modelling specifications.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 761508 (5GCity project) and the Spanish national project 5GCity (TEC2016-76795-C6-1-R).

References

[1] R. Sherwood, et al., Carving research slices out of your production networks with OpenFlow, SIGCOMM Comput. Commun. Rev. 40 (1) (2010) 129–130, http://dx.doi.org/10.1145/1672308.1672333.

[2] NGMN, Description of Network Slicing Concept, Tech. Rep., Next Generation Mobile Networks Alliance, 2016, https://www.ngmn.org/fileadmin/user_upload/160113_Network_Slicing_v1_0pdf.

[3] A. Galis, K. Makhijani, Network slicing landscape: A holistic architectural approach, orchestration and management with applicability in mobile and fixed networks and clouds, in: IEEE Network Softwarization (NetSoft), 2018.

[4] K. Samdanis, X. Costa-Perez, V. Sciancalepore, From network sharing to multi-tenancy: The 5G network slice broker, IEEE Commun. Mag. 54 (7) (2016) 32–39, http://dx.doi.org/10.1109/MCOM.2016.7514161.

[5] ATIS, ATIS-I-0000052: Neutral Host Solutions for Multi-Operator Wireless Coverage in Managed Spaces, Tech. rep., Alliance for Telecommunications Industry Solutions, 2016, https://access.atis.org/apps/group_public/download.php/31137/ATIS-I-0000052pdf.

[6] ETSI, Network Functions Virtualisation (NFV) - Architectural Framework, GS NFV 002 v1.2.1, Tech. rep., European Telecommunications Standards Institute, 2014, https://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/010201_60/gs_NFV002v010201p.pdf.

[7] C. Colman-Meixner, H. Khalili, K. Antoniou, M.S. Siddiqui, A. Papageorgiou, A. Albanese, P. Cruschelli, G. Carrozzo, L. Vignaroli, A. Ulisses, P. Santos, J. Colom, I. Neokosmidis, D. Pujals, R. Spada, A. Garcia, S. Figerola, R. Nejabati, D. Simeonidou, Deploying a novel 5G-enabled architecture on city infrastructure for ultra-high definition and immersive media production and broadcasting, IEEE Trans. Broadcast. 65 (2) (2019) 392–403, http://dx.doi.org/10.1109/TBC.2019.2901387.

[8] M.F. Bari, R. Boutaba, R. Esteves, L.Z. Granville, M. Podlesny, M.G. Rabbani, Q. Zhang, M.F. Zhani, Data center network virtualization: A survey, IEEE Commun. Surv. Tutor. 15 (2) (2012) 909–928.

[9] A. Pras, J. Schoenwaelder, RFC 3444: On the Difference Between Information Models and Data Models, Tech. rep., IETF Internet Draft, 2018, https://tools.ietf.org/html/rfc3444.

[10] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. Ramos-Munoz, J. Lorca, J. Folgueira, Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges, IEEE Commun. Mag. 55 (5) (2017) 80–87, http://dx.doi.org/10.1109/MCOM.2017.1600935.

[11] N. Nikaein, E. Schiller, R. Favraud, K. Katsalis, D. Stavropoulos, I. Alyafawi, Z. Zhao, T. Braun, T. Korakis, Network store: Exploring slicing in future 5G networks, in: Proceedings of the 10th International Workshop on Mobility in the Evolving Internet Architecture, ACM, 2015, pp. 8–13.

[12] M. Leconte, G.S. Paschos, P. Mertikopoulos, U.C. Kozat, A resource allocation framework for network slicing, in: IEEE INFOCOM 2018 - IEEE Conference on Computer Communications, 2018, pp. 2177–2185.

[13] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, H. Flinck, Network slicing and softwarization: A survey on principles, enabling technologies, and solutions, IEEE Commun. Surv. Tutor. 20 (3) (2018) 2429–2453, http://dx.doi.org/10.1109/COMST.2018.2815638.

[14] X. Foukas, G. Patounas, A. Elmokashfi, M.K. Marina, Network slicing in 5G: Survey and challenges, IEEE Commun. Mag. 55 (5) (2017) 94–100, http://dx.doi.org/10.1109/MCOM.2017.1600951.

[15] 3GPP, Study on Management and Orchestration of Network Slicing for Next Generation Network, Technical Report (TR) 28.801, 3rd Generation Partnership Project (3GPP), version 15.1.0, 2018, https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3091.

[16] 3GPP, Management and Orchestration; Provisioning, Technical Specification (TS) 28.531, 3rd Generation Partnership Project (3GPP), version 15.0.0, 2018, https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3274.

[17] 3GPP, Management and Orchestration; 5G Network Resource Model (NRM), Technical Specification (TS) 28.541, 3rd Generation Partnership Project (3GPP), version 15.0.1, 2018, https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3400.

[18] ETSI, Network Functions Virtualisation (NFV) - Report on Network Slicing Support with ETSI NFV Architecture Framework, GR NFV-EVE 012 v3.1.1, Tech. rep., European Telecommunications Standards Institute, 2017, https://www.etsi.org/deliver/etsi_gr/NFV-EVE/001_099/012/030101_60/gr_NFV-EVE012v030101p.pdf.

[19] ETSI, Network Functions Virtualisation (NFV) - Network Service Templates Specification, GS NFV-IFA 014 v2.4.1, Tech. rep., European Telecommunications Standards Institute, 2018, https://www.etsi.org/deliver/etsi_gs/NFV-IFA/001_099/014/020401_60/gs_NFV-IFA014v020401p.pdf.

[20] ETSI, Next Generation Protocols (NGP) - E2E Network Slicing Reference Framework and Information Model, GR NGP 011 v1.1.1, Tech. rep., European Telecommunications Standards Institute, 2018, https://www.etsi.org/deliver/etsi_gr/NGP/001_099/hellocheckthecomm{and}11/010101_60/gr_ngp011v010101p.pdf.

[21] L. Qiang, A. Galis, L. Geng, K. Makhijani, P. Martinez-Julia, H. Flinck, X. de Foy, Technology Independent Information Model for Network Slicing, Tech. rep., IETF Internet Draft, 2018, https://datatracker.ietf.org/doc/draft-qiang-coms-netslicing-information-model/.

[22] R. Ferrús, O. Sallent, J. Pérez-Romero, R. Agustí, Management of network slicing in 5G radio access networks: Functional framework and information models, 2018, CoRR journal abs/180301142.

[23] K.I. Pedersen, G. Berardinelli, F. Frederiksen, P. Mogensen, A. Szufarska, A flexible 5G frame structure design for frequency-division duplex cases, IEEE Commun. Mag. 54 (3) (2016) 53–59.

[24] DMTF, CIM (Common Information Model) Schema: Version 2510, Technical specification (ts), Distributed Management Task Force, version 2.5.1.0, 2018, https://www.dmtf.org/st{and}ards/cim/cim_schema_v2510.

[25] W. Fuertes, J.E. López de Vergara Mendez, F. Galán, A generic model for the management of virtual network environments, in: 2010 IEEE Network Operations and Management Symposium - NOMS 2010, 2010, pp. 813–816, http://dx.doi.org/10.1109/NOMS.2010.5488367.

[26] M. Ghijsen, J. van der Ham, P. Grosso, C. de Laat, Towards an infrastructure description language for modeling computing infrastructures, in: 2012 IEEE 10th International Symposium on Parallel and Distributed Processing with Applications, 2012, pp. 207–214, http://dx.doi.org/10.1109/ISPA.2012.35.

[27] EU Project 5GTango, 5G Network Slicing in 5GTANGO, Tech. rep., https://5gtango.eu/blog/36-5g-network-slicing-in-5gtango.html, last visited May 2019.

[28] 5G Americas, Network Slicing for 5G Networks and Services, Tech. rep., 5G Americas White Papers, 2016, http://www.5gamericas.org/files/3214/7975/0104/5G_Americas_Network_Slicing_1121_Final.pdf.

[29] EU Project Matilda, Deliverable D15: Deployment and Runtime Policy Metamodel, Tech. rep., http://www.matilda-5g.eu/index.php/outcomes, last visited May 2019.

[30] EU Project 5GCity, Deliverable 41, 5GCity Orchestration Design, Service Programming and Machine Learning Models, Tech. rep., https://www.5gcity.eu/deliverables/, last visited May 2019.

[31] G. Baldoni, et al., Edge computing enhancements in an NFV-based ecosystem for 5G neutral hosts, in: 5GNetApp Workshop At the IEEE Conference on NFV and SDN, 2018, pp. 1–5.

[32] D.L. Moody, Measuring the Quality of Data Models: An Empirical Evaluation of the Use of Quality Metrics in Practice, in: Proceedings of the 11th European Conference on Information Systems, ECIS '03, 2003, 1337–1352.

[33] R. Maier, Evaluation of data modeling, in: Proceedings of the Third East European Conference on Advances in Databases and Information Systems, ADBIS '99, Springer-Verlag, 1999, pp. 232–246.

[34] P. Mohagheghi, V. Dehlen, Existing model metrics and relations to model quality, in: 2009 ICSE Workshop on Software Quality, IEEE, 2009, pp. 39–45.

[35] P. Runeson, M. Höst, Guidelines for conducting and reporting case study research in software engineering, Empir. Softw. Eng. 14 (2) (2009) 131.