



**HAL**  
open science

## On the Virtualization of the Satellite Segment

Slim Abdellatif, Pascal Berthou, Samir Medjiah, Julien Julien Bernard,  
Patrick Gelard

► **To cite this version:**

Slim Abdellatif, Pascal Berthou, Samir Medjiah, Julien Julien Bernard, Patrick Gelard. On the Virtualization of the Satellite Segment. *The Journal of Engineering*, 2016, 10.1049/joe.2015.0162 . hal-01376205

**HAL Id: hal-01376205**

**<https://hal.science/hal-01376205v1>**

Submitted on 4 Oct 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# On the Virtualization of the Satellite Segment

Slim Abdellatif<sup>b,a</sup>, P. Berthou<sup>a,c</sup>, Samir Medjiah<sup>a,c</sup>, Julien Bernard<sup>e</sup>, Patrick Gelard<sup>d</sup>

<sup>a</sup> CNRS, LAAS, 7 avenue du colonel Roche, F-31400

<sup>b</sup> Univ de Toulouse, INSA, LAAS, F-31400, Toulouse, France

<sup>c</sup> Univ de Toulouse, UPS, LAAS, F-31400, Toulouse, France

<sup>d</sup> CNES, Centre National d'Etudes Spatiales 18 Av. Edouard Belin, 31400, Toulouse, France

<sup>e</sup> Viveris Technologies, 1 Avenue de l'Europe, 31400 Toulouse

**Abstract**— For years, Satellite Virtual Network Operators (VNOs) are a major player in the satellite communication market landscape. Typically, they repackage services leased from Satellite Network Operators (SNOs) to provide their customers with added-value end-to-end services. However, the level of control and visibility that Satellite VNOs have on their purchased services (and underlying resources) is limited mainly because of SNOs' protective/conservative policies and the closed nature of satellite devices. From the Satellite VNO perspective, this refrains the development of novel services and complicates the provision process of the services they offer. This paper proposes and elaborates on the concept of full virtualization of satellite Hubs that enables enhancing the level of control and visibility exposed to satellite VNOs. Analysis of the opportunities brought by this proposal is presented as well as insights on how it can be implemented.

**Keywords** — *Satellite communications; Virtual Network Operator; Network Virtualization*

## I. INTRODUCTION

For cost reasons, satellite Virtual Network Operators (VNO) are common players in the satellite communication landscape. They are typically commercial operators but can also be defense network operators. Both are facing an ever-increasing pressure to deliver new and advanced service offerings. To answer this need, a comprehensive level of visibility and control on the satellite resources that they lease from SNOs is a necessity. Unfortunately, this is not the case. Worse, some control capabilities require human intervention from the SNO to validate or perform the required configuration.

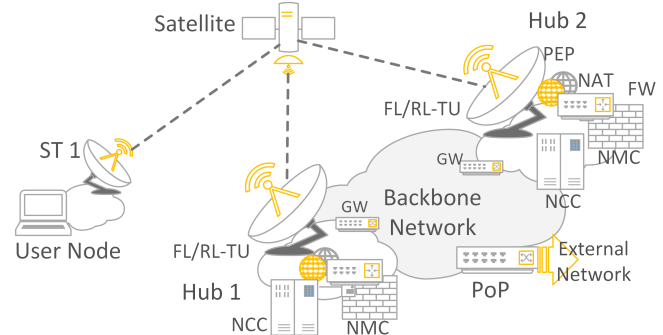
This paper addresses the problem of providing VNOs with a capability-rich interface to their leased satellite resources in order to enable the development of novel services, to achieve better resource utilization and to simplify and automate some of their management tasks.

This paper proposes and develops the idea of full virtualization of satellite Hubs, which was promoted in our previous work [1].

This paper is organized as follows. Sections II and III respectively describe the broadband satellite network under study as well as the motivations of this work. Sections IV and V detail our proposal, analyze their implications on VNO's capabilities and explain how they can be implemented. Section VI deals with our testbed realization. Section VII concludes the paper.

## II. SATELLITE NETWORK ARCHITECTURE

This work considers a typical Broadband Satellite Network (BSN) that provides a multi-beam coverage with forward and return links. The ground segment of the BSN gathers multiple Hubs that are interconnected via a dedicated backbone network with some PoPs (Point of Presence) or gateways to external networks, typically the Internet (see Figure 1).



**Figure 1 – Satellite Communication Architecture.**

Generally, a Hub supports bidirectional traffic on one or many beams. It combines a Forward Link Transmission Unit (FL-TU) and a Return Link Reception Unit (RL-RU) with a Gateway (GW) to terrestrial networks, a Network Control Centre (NCC) and a Network Management Centre (NMC). The FL-TU performs baseband related functions like DVB-S2 coding and modulation with Adaptive coding and modulation (ACM). The NCC provides control functions; it typically performs Satellite Terminals (ST) admission control and resources control/allocation on the forward and return links. The NMC performs all management functions, i.e. network element's (ST, Hub) configuration, as well as fault, performance, accounting and security management. Performance Enhancing Proxy (PEP) designed to improve TCP performance over satellite links may also be co-located at the Hub (or deported at the PoPs or closer to end-users). The successful delivery of satellite communication services to end-users involves one or many real-life business actors, each playing one or many roles.

Referring to [2], three major roles are distinguished:

- *Satellite Operator (SO)*: it owns the satellite and assumes its operation. It leases satellite capacity at the

transponder level (physical layer) to one or several SNOs.

- *Satellite Network Operator (SNO)*: it operates a broadband satellite network with one or more satellite transponders and one or more satellite hubs. It provides satellite forward and return links to second-tier operators by dividing transponder level bandwidth. The NCC controls this bandwidth sharing. Via the NMC, the SNO provides a management interface to the purchased resources.
- *Satellite Virtual Network Operator* (referred below as *VNO*): Based on the satellite links contracted from one or multiple SNOs, it builds and provides end-to-end higher-level added-value services that are made available via a satellite access.

### III. PROBLEM STATEMENT

Satellite VNOs rely on a virtual network infrastructure (with a satellite segment and a terrestrial segment up to SNO's Point of Presence (POP)) leased from one or many SNOs to provide satellite communication services to its customers. Because of the expensive cost of satellite resources, two main business models are in use by VNOs [3]:

- 1) SNO resources are leased and integrated into VNO's network infrastructure as customer subscriptions go along.
- 2) SNO resources are leased beforehand, based on VNO's market predictions. Then, they are exploited to set up customer services. These resources may correspond, in the one hand, to either dedicated or shared physical components of the satellite Hub [4] and, on the other hand, either to guaranteed bandwidth or shared satellite links.

The key difference between these two models lies in the level of isolation between VNOs (guaranteed dedicated resources vs shared) as well as on the range of control and management capabilities that VNOs have on their resources (simply raising some performance indicators (in business model 1) to configuration capabilities on some of the VNO related Hub components and STs (business model 2)).

As part of a VNO offer, an SNO provisions the satellite resources (i.e. satellite communication services) on behalf of the VNO, via provisioning interface ( $PI_{SNO-VNO}$ ). It also makes available a management interface ( $MI_{SNO-VNO}$ ) to these resources that allows the VNO to implement its management functions (e.g. monitoring, configuration and accounting). Generally, it is based on SNMP complemented with vendor specific interfaces.

Figure 2 describes the relationships between the SNO's and VNO's management systems materialized by the  $PI_{SNO-VNO}$  and  $MI_{SNO-VNO}$  interfaces. Figure 2 also presents the data management paths and highlights the fact that many (if not most) management actions go through SNO's NMC. Some, i.e. higher layer related statistics and configurations (e.g. routing, VPN, etc.), go directly from VNO's NMS to STs.

The provisioning of satellite resources and services via the provisioning interface,  $PI_{SNO-VNO}$ , is not automated and implies human intervention. Moreover, the amount of

resources is predefined and does not adapt to varying VNOs needs (unless the above-cited non-automated provisioning process is triggered again).

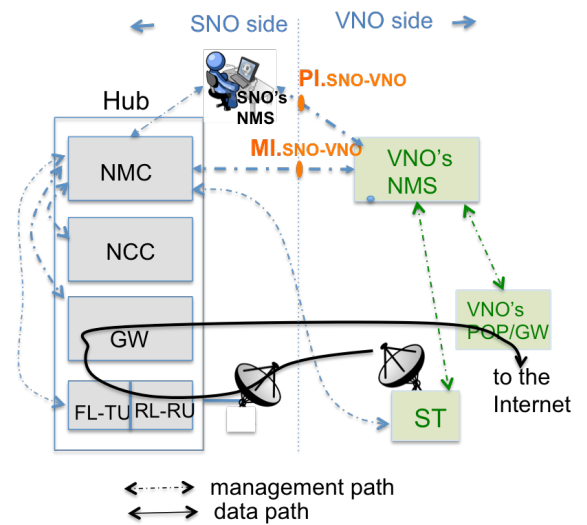


Figure 2 - SNO to VNO interfaces

On the other hand, the range of capabilities of the management interface,  $MI_{SNO-VNO}$ , is quite limited mainly because of the SNO's protective policies that aim at ensuring foolproof isolation between VNOs. Besides, in practice, a resource allocation at the level of physical transmit and receive units (FL-TU and RL-RU) is the condition to provide VNOs with a well-furnished management interface.

Current SNO practices with human intervention in the loop, static and coarse resource allocation coupled with a limited exposed management interface restrain VNOs from (1) automating some of their network and service management functions, (2) developing and building new services, and (3) optimizing their leased resources usage. Network operations from SNO and VNO are intimately linked. In current satellite network architectures, the network monitoring and resource allocations are done inside the satellite network with the NMC and NCC cooperation, under the responsibility of the SNO. Although, a VNO should do the same operations to control the traffic it transits, the interface between the SNO and VNO ( $MI_{VNO}$ ) is designed in this goal.

The  $MI_{VNO}$  interface is generally based on SNMP with vendor dependent management objects, complemented with non-standard vendor specific interfaces. Recent systems are using either ETSI BSM or 3GPP compliant architectures. In the first case, the  $MI_{VNO}$  interface is restricted either to the SI.SAP interface [5] or to protocols such as IMS/COPS in the second [6]. They limit the VNO network operations to what these interfaces allow.

VNOs are asking for an increased level of visibility and control on their leased resources with reduced, if no,

intervention from the SNOs. Hereafter, we elaborate on the application of satellite hub virtualization to address this need.

#### IV. SATELLITE HUB FULL VIRTUALIZATION

##### A. Key idea and implications on VNO capabilities

Satellite Hub virtualization with a full-featured logical satellite hub instance assigned on a VNO basis is the answer to the issues pointed out at the previous section. Each Virtual Hub provides satellite communications access to VNO's clients via virtual forward and return links. Their capacities are in line with the SLA contracted by the VNO and may range from simply sharing the available resources with other VNOs on a best-effort basis to a fine-grained guaranteed bandwidth allocation that may change over time with VNO needs. In addition to its dedicated virtual forward and return links, a virtual hub possesses (1) its own independent NCC that controls the access of its client stations/traffic to its virtual satellite links; (2) its own independent NMC that implements the management functions that apply to the leased resources, virtual hub components and its client stations; and (3) its own virtual gateway that routes VNO related customer traffic from and to the terrestrial backbone (as shown in Figure 3).

Assuming the right level of isolation between virtual hubs, we argue that an interface to the virtual hub can be provided to the VNO with a range of capabilities that is quite equivalent to the interface that the SNO has from its physical Hub. More precisely, such interface, denoted as  $I_{VNO}$  in Figure 3, allows the monitoring of all the satellite communications of the VNO's clients as well as the resources leased by the VNO. It also allows the configuration of virtual hub components such as the Virtual NCC and the virtual GW as well as the STs belonging to the VNO's customers. As a consequence, the provisioning of a VNO customer's satellite communication service can be fully performed and automated by the VNO with no intervention from the SNO, unless extra resources (mainly forward and return link resources) are needed. Indeed, referring back to the business models described in the previous section, resource overprovisioning cannot be used extensively by VNOs. So, when provisioning some customer services, the VNO might require adjusting the amount of leased resources to the SNO in order to afford customer's needs. This requires involving the SNO via a dedicated Resource Provisioning Interface denoted by  $RPI_{VNO}$  in Figure 3. By enabling device virtualization at the Hub level with guaranteed isolation between VNOs (particularly for the forward and return links but also for the other Hub resources (CPU, memory, TCAM, etc.)), this resource provisioning process can be automated at the SNO side with no human intervention in the loop. As a consequence, the provision process of VNO's customer services can be fully automated (with an efficient resource usage from the VNO) despite the possible involvement of the SNO.

The granular and dynamic allocation of satellite resources to the VNO coupled with the VNO's independent and

configurable instance of the NCC (that controls the access to these resources) allows unprecedented efficient and customized resource usage policies; it is the key building block to the emergence of new types of services provided by VNOs, like, for instance, those borrowed from the cloud computing industry "you pay as you go", etc.

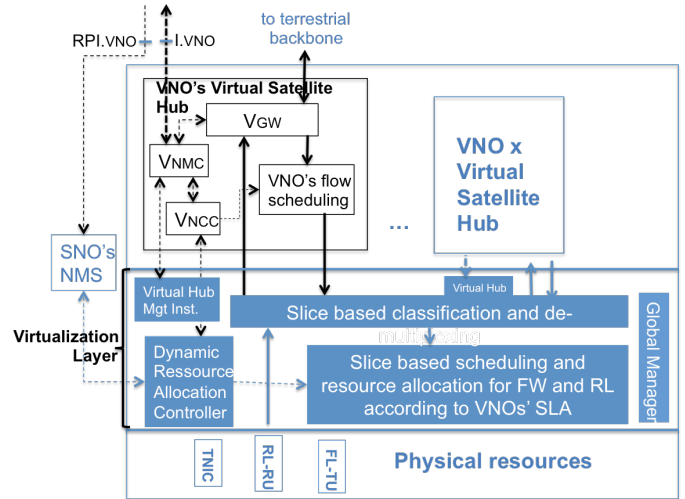


Figure 3 - Functional components of a virtualization satellite Hub

##### B. Virtualization Layer functional requirements

The ultimate goal of the virtualization layer is to provide multiple separate logical instances of a satellite Hub (with its different components) that can be independently managed. Classically, the virtualization layer is in charge of partitioning, between the different instances, the physical resources of the Hub, namely, the CPU, memory, TCAM, terrestrial network interface cards (TNIC) and the forward/return links along with their respective transmit/receive units. A stringent level of isolation must be ensured in order to enable a comprehensive management interface to the virtual hub. This applies to data and control traffic but also to performance and fault isolation in order to mitigate the impact of one misbehaving virtual hub on other virtual hubs.

###### 1) Satellite resources virtualization

Hereafter, the focus is on the functional requirements that relate to satellite resources. The others result from server or OS virtualization and/or router virtualization. Figure 3 shows the main functional components of the virtualization layer. The FW/RL links partitioning or sharing between virtual Hubs is enforced by the *slice based scheduling and resource allocation* component. Different granularity levels must be supported: line card level, carrier level, and most importantly at the time slot level of a carrier. The behavior of this component is controlled by the *Dynamic Resource Allocation*

*Controller* that ensures that the resources that are delivered to each virtual Hub is in conformance with the SLA contracted by the associated VNO. Different SLA models must be considered to suit VNO's business needs. One important requirement is that dynamic SLA must be allowed, that is, as explained above, the VNO is allowed to ask for and change over time the amount of satellite resources. It is up to *Dynamic Resource Allocation Controller* to check whether enough resources are available and proceed with a reallocation or deallocation of resources. This is achieved by tuning the *slice based scheduling and resource allocation* component.

Integrating the slicing technique within the Hub (MAC scheduling) is the condition to achieve high level of isolation across slices and efficient resource usage. This follows the same logic as the Network Virtualization Substrate proposed in [7] for WIMAX networks and differs from other directions that may consider the Hub as a black box and implement the slicing out of the box. Virtual Network Traffic Shaper [8] is such an approach that does not impose any modification on the Hub. The price to pay is a coarse isolation and the lack of slicing on the return link.

### 2) Hub management virtualization

In order to provide each VNO with a dedicated comprehensive management interface, the virtualization layer activates on a per VNO basis an agent module (*Virtual Hub Management instance* in Figure 3) in charge of implementing common management functions (e.g. monitoring status and statistics, configuration, etc.) that apply to VNO related resources and network elements. This module interacts with the virtual Hub's NMC instance (i.e.  $V_{NMC}$ ), which is the front-end module in charge of exposing a personalized interface to the VNO (i.e.  $I_{VNO}$ ). This personalized interface is obtained by filtering out of the range of capabilities of *Virtual Hub Management instance*, the unwanted features. With the use of *Virtual Hub Management instance modules*, the virtualization layer enforces management isolation between virtual hubs, in so far as each VNO perform the overall management activities only on its leased resources and network elements. Another requirement is configuration isolation, which must not restrict VNOs on the IDs or numbers (e.g. overlapping addresses, VLAN-IDs, etc.) to employ.

### 3) Classifier

Finally, the *slice-based classification* module classifies each packet in order to derive the originating or destined virtual Hub. This classification can take different forms and rely on packet addressing fields or an explicit Slice-ID-tag incorporated in the packets.

### C. Virtual Satellite Hub

Instead of using a unique instance of each satellite network components (NCC, NMC, GW...) several virtual instances are setup to give to the VNO the same view the SNO has. Each virtual component is connected to the Virtualization layer that transposes the request to the real satellite system.

As described previously, the virtual NMC ( $V_{NMC}$ ) provides the appropriate interface to configure and monitor VNO related network components and resources. The Virtual NCC ( $V_{NCC}$ ) can be tuned to implement personalized allocation algorithms. The Virtual gateway ( $V_{GW}$ ) can be easily reconfigured according to the VNO terrestrial network and its addressing plane. Finally, the VNOs can perform and customize the scheduling of their customer flows within their slice. This is achieved thanks to the *VNO's flow scheduling* module (in Figure 3), which is controlled by the virtual Hub's instance of the NCC ( $V_{NCC}$ ). This is necessary in order to let the VNOs implement their own policies, which is a prerequisite to building new services and achieving efficient resource utilization.

### D. Related Work

Some vendors are already offering VNO support for some time and thereby implement some form of Hub virtualization. This latter is primary based on a VNO partitioned access to the NMC, known as Network Management Domain, which delimits the access of VNOs to their STs. In order to open the access to other Hub components (GW, NCC) with a richer set of management capabilities, a hardware-based isolation between VNOs is assumed, i.e. each VNO get assigned a physical GW and Forward/Return link transmit/receive units [4].

The idea of Hub sharing is not new but our proposed virtualization scheme goes one step forward by instantiating logical (virtual) components (opposed to physical) per VNO while ensuring the right level of isolation. It allows a more fine-grained resource allocation that can be executed on demand and enables a rich-featured interface to VNOs.

## V. TECHNICAL SOLUTIONS

The virtualization of the satellite network could be deployed even over existing network infrastructures, as it does not necessarily require modifications on air interfaces, but only on the ground infrastructure. This section deals with technics and solutions to implement virtualization on a broadband IP satellite network.

First, the way to virtualize resources control is addressed, then a focus is made on how to virtualize the management plane, and finally we discuss user stream classification.

### 1) Virtualization of satellite resources control

Resource virtualization can be accomplished in many ways. At the physical level, solutions are numerous and can be close to solutions in terrestrial networks such as cloud-RAN [9] by separating the RF and the baseband to virtualize (achieved in software). To the best of our knowledge, these solutions are not yet deployed in satellite systems. These techniques are appropriate to achieve foolproof physical isolation, but come at the cost of significant complexity. An allocation of time slots on existing carriers may be realized simply by configuring the NCC. To this end, the virtualization layer will provide an interface to time-slot reservation on uplink and downlink channels. For the return path, this technique should



be combined with the DAMA algorithm. A solution at the IP level can be performed very simply by the introduction of a traffic shaper. For each VNO, a shaper will control the streams entering on the forward link. On the return link, shapers must be also installed on the terminals and combined with a permissive DAMA algorithm since it is no more accountable to limit traffic. A current technical solution would be to rely on metering tools available in the latest versions of the OpenFlow protocol [10].

### 2) Hub management and classifier

The management of the Hub can be implemented by a software that segments the access to a management base (MIB SNMP). A limited MIB can be exported as the Hub interface of the virtual operator ( $V_{NMC}$ ). The classifier separates the streams of the various operators and directs them to the appropriate virtual hub. Solutions based on the 802.1q tagging are ideal for this. OpenFlow is quite suitable for this with the ability to stack the VLAN tags by PUSH-POP operations. In case OpenFlow would be used in the virtualization layer, the NMC is the place where should be implanted an OpenFlow controller because all configurations requests go through it.

### 3) VNO components - gateway

The VNO components are also implemented in software. However several other options exist for the virtualization of the gateway. The virtual gateway is responsible for the interconnection of the VNO satellite network with the terrestrial one. In fact, the main gateway function is routing and forwarding. Virtualization can be handled by Virtual Routing and Forwarding (VRF) functions. VRF allows multiple instances of a routing table to co-exist within the same router and have to be configured either manually or by using classical routing protocol. VRF are easy to deploy but linked to a legacy router. An Openflow virtual switch can be a good alternative as it also offers the forwarding function. In that case, the flow table should be configured according to the controller's rules. *Open vswitch* is a software implementation of an OpenFlow switch and should be enough to support the forwarding functions of the virtual gateway. The VNOs flow scheduling function could also be implemented using this switch. The OF switch is able to classify traffic according to a rule based on any fields of the packet (for instance the classical socket 4-tuple).

## VI. IMPLEMENTATION BASED ON SDN CONCEPTS

This concept has been partially implemented within the open-source satellite platform OpenSAND. OpenSAND [11] is a user-friendly and efficient tool to emulate satellite communication systems, mainly DVB-S2/RCS. It provides a suitable and simple means for performance evaluation and innovative access and network techniques validation. Its ability to interconnect real equipment with real applications provides excellent demonstration means. A typical emulation platform is composed of the following components, which are deployed on separate machines inter-connected via an OpenSAND LAN: the satellite emulator (SE), the OpenSAND

Gateway (O-GW) and at least one OpenSAND Satellite Terminals (O-ST). An OpenSAND manager running on the SE machine is used to control the platform (configuration, supervision, etc.);

According to the technical advices of the previous section, the implementation has been done using Software Defined Concepts and more precisely using OpenFlow protocol and switches. This choice has been motivated by the flexibility of this new protocol that provides the perfect glue between implemented software components. As shown in Figure 4, the SVNO virtual hub (green) is composed of two virtual Openflow switches (OFsoftswitch) controlled by a single VNO controller. The virtual NCC controls the forward link that allows a per slot on demand allocation and the return link that implements a DAMA over the DAMA slots allowed to the VNO. To simplify the switching function of the gateway, the legacy gateway also implements an OF Switch controlled by the SNO controller (blue).

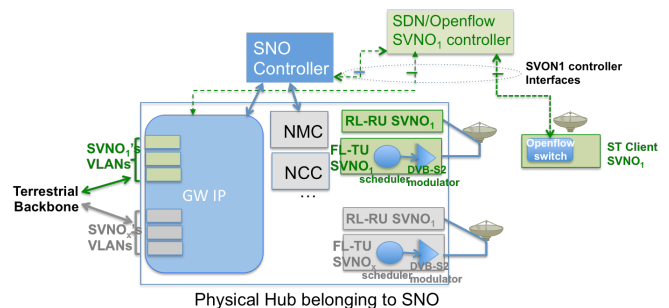


Figure 4 - design of the virtual hub

To install a new SNO, number of OF rules should be installed inside the gateway switches. To add a new ST, rules should be also added to the ST switch and the  $V_{GW}$  switch. In fact, rules are only installed on the controllers and automatically requested by the switches when unknown traffic comes in. The satellite network does not affect the Openflow protocol good working. VNO traffic are marked using different vlan id, thanks to the related OF action. Thanks to the multiple encapsulations, our testbed is configured to offer an end-to-end Ethernet service for the VNOs although it relies on the SNO IP base satellite network. Then, VNOs completely control their addressing plan. Openflow meters are installed inside the VNOs switches to limit uplink and downlink traffics according the agreed SLAs.

The VNOs are perfectly isolated thanks to the allocations made by the *Dynamic Resource Allocation Controller* and the  $V_{NCC}$  that use different slots. VNO can manage the return link resources (and moreover the forward link resources) as a pool of resources shared between the different STs. However, the isolation that must be ensured by OF meters implemented on the VNOs virtual switches suffers from the ofsoftswitch software implementation. The shaping is not stable as it must be. For an operational setup, we recommend to rely on a hardware implementation of OF meters that can be found in Openflow switches.

The testbed is not fully compliant with the above specifications, as the ( $V_{NMC}$ ) has been simplified and does not offer a full SNMP interface. The dynamic resource allocator is also simplified and is implemented as a hack of the existing OpenSAND allocator. However, our test bed proves that the virtualization of an existing satellite network is possible and could partly rely on SDN concepts.

## VII. CONCLUSION

Extending the capabilities of the interface that SNOs expose to VNOs with reduced SNO and human intervention are a prerequisite to let VNOs enrich their satellite related service catalog, automate their service provisioning procedures, and use their leased resources more efficiently. To this end, we propose in this paper the full virtualization of SNO's satellite Hub, which also allows fine-grained dynamic satellite-resource allocations to VNOs. Some aspects of this proposal were implemented and validated on the OpenSAND Platform using SDN technologies. By this way, we intend to prove that the virtualization of an existing satellite network is possible. This will offer new perspectives for the integration of the satellite networks inside a global and hybrid terrestrial offer. The main perspectives to this work are pursuing the implementation of our prototype and investigating which of the functions of the virtual hub can be pushed into the cloud following the NFV approach. This is also the trend followed in terrestrial mobile networks.

## ACKNOWLEDGMENT

This work is partially funded by the French Space Agency (CNES) and by the French National Research Agency (ANR), the

French Defense Agency (DGA) under the project ANR DGA ADN (ANR-13-ASTR-0024).

## REFERENCES

- [1] L. Bertaux, S. Medjiah, P. Berthou, S. Abdellatif, P. Gelard, F. Planchou, M. Bruyere, M., "Software defined networking and virtualization for broadband satellite networks," *Communications Magazine, IEEE*, vol.53, no.3, pp.54,60, March 2015
- [2] ETSI TS 101 545-1 V1.1.1, "Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System (DVB-RCS2), Part 1: Overview and System Level specification," May 2012.
- [3] Virtual Network Operator (VNO) Program Technology Brief, IDirect, 2014
- [4] Deliver World-Class Services without Major Capital expenditures, Hughes, available at: <http://global.hughes.com/global-solutions/vno-capabilities>
- [5] ETSI TS 102 357 V1.1.1, "Satellite Earth Stations and Systems (SES); Broadband Satellite Independent Service Access Point (SI-SAP)", May 2005.
- [6] ETSI TS 102 855 V1.1.1, "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Interworking and Integration of BSM in Next Generation Networks (NGNs)", March 2011.
- [7] R. Kokku, R. Mahindra, H. Zhang, S. Rangarajan, "NVS: A Virtualization Substrate for WiMAX Networks," in *Proceedings of the 16th Annual International Conference on Mobile Computing and Networking*, Sept. 2010
- [8] G. Bhanage, R. Daya, I. Seskar, D. Raychaudhuri, "VNTS: A Virtual Network Traffic Shaper for Air Time Fairness in 802.16e Systems" in *IEEE International Conference on Communications*, May 2010
- [9] A. Checko, H.L. Christiansen, & al., "Cloud RAN for Mobile Networks—A Technology Overview", *Communications Surveys & Tutorials, IEEE*, Volume:17, Issue: 1, page 405-426, sept. 2014.
- [10] Open Networking Foundation, "OpenFlow Switch Specification – version 1.4.0 (Wire Protocol 0x05)", Oct. 2013
- [11] OpenSNAD Platform, available at : <http://opensand.org>