Service Delivery Models for Converged Satellite-Terrestrial 5G Network Deployment: A Satellite-Assisted CDN Use-Case

Article *in* IEEE Network · January 2018

DOI: 10.1109/MNET.2018.1800020

4 authors:

Michele Luglio
University of Rome Tor Vergata

Simon Pietro Romano
University of Naples Federico II

Cesare Roseti
University of Rome Tor Vergata

Francesco Zampognaro
University of Rome Tor Vergata

Service delivery models for converged satellite-terrestrial 5G network deployment: a satellite-assisted CDN use-case

Michele Luglio, Simon Pietro Romano, Cesare Roseti and Francesco Zampognaro

Abstract—The current shift towards the virtualization of network infrastructure components enables a dynamic instantiation, deployment and configuration of virtual network functions (VNFs), which can be offered "*as-a-Service*" to multiple tenants, thus enabling 5G architectures. Simultaneously, the recent High Throughput Satellite (HTS) systems can play an important role in the 5G era thanks to their characteristics, such as the large coverage, fast deployment of the ground infrastructure and native broadcast/multicast broadband capabilities. In this context, this paper proposes a review of the satellite service delivery models in order to identify viable alternatives to deploy converged satellite-terrestrial services. This objective is pursued by taking as a reference a satellite-assisted IP streaming service for the enhancement of current Content Delivery Network (CDN) infrastructures, as tackled by the European Space Agency within the SHINE (*"Secure Hybrid In Network caching Environment"*) project. SHINE aims at efficiently extending terrestrial CDN services to satellite-enabled scenarios, by designing innovative mechanisms for the secure distribution of real-time multimedia information across hybrid channels, leveraging both unicast and multicast communication paradigms. The original contribution of the paper is the analysis of satellite architectures and configurations tailored to efficiently support the SHINE solution, together with a high-level applicability assessment taking into account different satellite-enabled service models.

Index Terms—5G, CDN, HTS, NFV, Virtualization

1 Introduction

Slicing and multi-tenancy concepts, which have been successfully applied to cloud computing technologies, are nowadays extended to the networking domain. Basically, slicing allows a network operator to provide dedicated virtual resources with associated functions to a specific service/tenant. As a consequence, multiple virtual networks can be created on top of a shared physical infrastructure. Such a new communication paradigm exploits two main technology enablers: Software Defined Networking (SDN) [1] and Network Function Virtualization (NFV) [2]. SDN allows the de-coupling between data and control planes in network nodes, with a central controller that dynamically establishes the forwarding rules and network nodes just enforcing such rules; NFV envisages the instantiation of network functions as software modules running on shared commodity hardware. Therefore, SDN can be used to dynamically and flexibly enable NFV by properly configuring the forwarding path.

Meanwhile, High Throughput Satellite (HTS) systems have recently become available and offer more than double the total throughput than a traditional Fixed Satellite Service (FSS), for the same amount of allocated frequency band on a geostationary orbit. HTS systems take advantage of high degree frequency reuse among multiple spot beams to increase the total throughput. Furthermore, most HTS systems exploit Ka-band or even Q/V-band, which allows wider bandwidth compared to traditional Ku-Band. Thanks to these characteristics, HTS systems have the potential to provide a high speed broadband Internet access, not only for areas uncovered by terrestrial networks, but also as a complement to the traditional terrestrial systems [3].

The convergence of such satellite platforms in 5G architectures undeniably represents a great opportunity for the improvement of the supported integrated services in a twofold manner: increasing the overall capacity and introducing specific satellite features such as the large geographical coverage and a native broadcast/multicast link [4]. Therefore, satellite platforms can play a fundamental role in supporting B2B applications, such as satellite-assisted local caching, which in turn should represent a key feature of the future 5G Mobile-Access Edge Computing (MEC) functionality [5].

In this paper we propose the delivery of multimedia content across an innovative Content Delivery Network (CDN) involving both satellite and terrestrial trunks. The target CDN leverages state-of-the-art technologies in the core network, whereas the edge component is fed by a satellite broadcast/multicast channel for an efficient local in-network caching at micro-centre locations. Such a hybrid satellite-terrestrial architecture has been designed in the frame of the SHINE *"Secure Hybrid In Network caching Environment"* project [6] that includes, among its objectives, an efficient and secure end-to-end delivery of multimedia contents through a combination of network coding and standard real-time streaming techniques.

- *Author list is in alphabetic order.*
- *M. Luglio, C. Roseti and F. Zampognaro are with University of Rome "Tor Vergata", Department of Electronics Engineering, via del Politecnico 1, 00133, Rome, Italy.*
- *S.P. Romano is with Università degli Studi di Napoli Federico II, Napoli, Italy, Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione.*

2 Satellite service delivery models for CDNs

A CDN is a collaborative set of network elements distributed over the Internet, with the aim to support content replication across several mirrored servers/caches. A fundamental aspect addressed herein is the use of an HTS platform to support satellite-assisted caching services. Indeed, the reduced cost per bit of HTS systems has a positive impact on the value chain economics, which translates into greater opportunities for the satellite-enabled delivery chain. First and foremost, HTS satellite integration in a CDN service needs to clearly address the following open issues:

- the selection of the space segment owned by a satellite operator;
- the identification of a teleport infrastructure, meant as ground station allowing physical broadcast transmissions and providing interconnections to the terrestrial network;
- the definition of the reference Hub technology (which constrains terminal selection);
- the identification of the most suitable satellite network setup and configurations which satisfy service requirements (channels, enabled services, etc);
- the management of IP-based services over the satellite domain.

A survey on possible business models suitable for the integration of HTS in current and future communication services is provided in [7] and can be adapted for the target CDN. The *Vertically Integrated Model* consists in a Business-to-Customer (B2C) application, where the satellite operator manages the whole business chain, from the satellite infrastructure to the service subscriptions. Accordingly, a CDN provider, similarly to a common end-user, subscribes a service agreement by selecting one of the available profiles without any specialized configurations and constrained to specific end-user hardware (vendor lock). From the CDN provider perspective, this model presents as a drawback the limited flexibility in service configuration and operational costs (service subscriptions) not properly favorable to scalability. In the *Managed Service Model*, the satellite operator owns and operates the whole satellite infrastructure and takes in charge network operations, while delegating service management to third-parties. This model can prove useful for terrestrial service providers, such as a CDN one, aiming to enhance the offered services by adding satellite capabilities. By exploiting the NFV architecture, the CDN provider can activate and manage one or more Virtual Network Functions (VNFs) on the satellite operator premises as enablers for the target service. An example of the above mentioned VNFs will be provided in section 3.1. The *Virtual Network Operator (VNO) Model* is an attractive business model allowing the satellite operator to lease a virtual space of its hub resources to service providers. In this case, a CDN provider can purchase a line card in the operator's hub to establish the service and has full control over satellite network operations. Therefore, CDN providers are able to deliver fully tailored services according to the end-user requirements, while being still constrained to adopt satellite technology and end-user hardware imposed by the

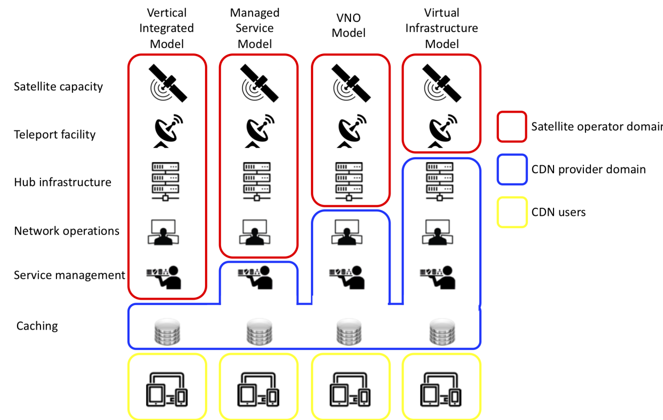

Fig. 1. Satellite delivery models for CDN services.

satellite operator. This model falls in the *Platform as a Service* (PaaS) category, where the satellite network operator is still responsible for the satellite communication platform, while a CDN provider can flexibly develop its own applications and configure the related networking aspects. Finally, the *Virtual Infrastructure Model* further increases CDN providers control over the satellite service provisioning chain, by allowing them to selecting and configure even the ground infrastructure. In more details, the CDN provider opts for a Hub technology and takes care of the maintenance operations (i.e., updates, fail-over and backup). Then, the satellite operator manages only the physical space infrastructure and RF building-blocks, following an *Infrastructure as a Service* (IaaS) approach and lowering its OPEX and CAPEX costs by sharing the overall infrastructure costs.

Fig. 1 summarizes the possible satellite service delivery for CDN. Definitively, the proposed scheme highlights the different business interaction boundaries between a CDN provider and the satellite operator. As a general insight, this service model review reinforces the importance to move towards an NFV architecture for the deployment of new applications to help a synergistic integration with network operators (i.e. satellite), who in turn can provide, *"as a Service"*, functionality ranging from a few *Network Functions* to the whole communication *Infrastructure*. Therefore, the implementation of a caching technique over an NFV architecture has been taken as a pre-requisite for the SHINE solution (see Section 3). In this perspective, caching can be part of a more complex and effective virtual instantiation orchestrated by a CDN Provider.

3 A converged satellite-terrestrial architecture to support multimedia content delivery

The SHINE project has two main distinctive features: a broadcast-enabled satellite-based core network and the edge distribution networks. Within the core of the network, we rely on network coding in order to define a coded multicast technique allowing us to improve both performance and security of communications. At the edges of the distribution network, which also act as in-network caches, we leverage cutting-edge streaming technologies (namely, MPEG-DASH [8] and/or WebRTC [9]) in order to optimize content distribution towards the end users of the content delivery service.

The starting goal behind the introduction of coded multicast in SHINE is the improvement of scalability thanks to the effective exploitation of the available broadcast communication channel interconnecting the transmitting station and the underlying receiving edge cache stations. Such edge caches are supposed to have no device-to-device capabilities (i.e., we do not envisage the presence of any cooperative caching scheme.). Indeed, SHINE does not only leverage Network Coding to improve performance, but also proposes to slightly modify the basic algorithm in order to provide the broadband-enabled part of the network with advanced security capabilities. With the improved NC-based transmission scheme we propose, we hence achieve the twofold objective of optimizing the use of the shared communication channel and increasing its robustness to malicious activities thanks to the transmission of properly obfuscated frames. The security aspects of SHINE are out of the scope of this paper and will not be discussed in detail. The interested reader is referred to the work in [10] for the coded multicast performance improvements, as well as enhanced security features. We will instead herein focus on the use of NC as a virtualised network function, framing this novel approach into the SHINE architecture.

A high-level view of the SHINE architecture is reported in Fig. 2. The picture highlights the main logical components of the architecture, in terms of macro-blocks and related functionality. Namely, we identify the following elements:

1) *a source encoder block*, taking on the responsibility of properly encoding the original content in order to allow for the subsequent coded multicast transmission over the satellite network;
2) *core satellite-enabled communication infrastructure*, looking after DVB-enabled transmission [11] of coded multicast frames from the content provider to the edge caches, both during the cache population phase and during the steady-state operation of the CDN;
3) two different "flavors" of edge access networks: (i) a *WebRTC-enabled access network*, included in the architecture in order to demonstrate SHINE's operation in the presence of this novel real-time communication infrastructure at the edges of the overall content delivery architecture; (ii) an *MPEG-DASH enabled access network*, included in the architecture in order to demonstrate SHINE's capability of leveraging such a well-assessed web-based distribution approach.

Satellite multicast support for Network Coding (NC) is of uttermost importance for the overall SHINE caching. Specifically, the source encoder is a software module implementing the main logic behind the proposed coded multicast technique. It is in charge of transforming the original content and applying the required transformations in order to arrive at a representation format that is suitable for the subsequent coded multicast transmission. The component in question has indeed to look after both the cache population phase and the actual content delivery phase. The cache population phase envisages that the edge caches pre-fetch some content, based on appropriate functions of the content library, as well as on information about estimated future users' demand for content. During the delivery phase, on the other hand, the source forms a multicast "codeword" to be transmitted over the shared link in order to meet the actual users' content demands. As already stated, we envisage that the cache population phase is carried out through transmission (over the satellite core network connecting source node with edge caches) of content chunks. As to the content delivery phase, it takes place through DVB-encapsulated transmission, over the satellite network, of coded multicast frames.

3.1 SHINE and the Virtual Network Coding Function

Network Coding (NC) is a novel technology that can be seen as the generalization of classic point to point coding, since it introduces the concept of coding for network flows. This paradigm shift is a fundamental aspect, since it brings in the concept of "service intent" to the provisioning of both functional and non-functional Quality of Service (QoS) requirements.

Accordingly, NC-based multicast represents a quite disruptive approach for local CDN feeding, when compared to traditional popularity-based methods. It allows to serve incoming requests in a combined fashion in order to reduce the overall delivery load (and hence the requested bandwidth), by exploiting the satellite multicast capability, instead of trying to maximize cache hits, as pursued by popularity-based approaches. Thus, the SHINE approach is convenient when the cache population increases (higher number of requests) and the content popularity is not easily assessable, so that it can be adopted as a complement to traditional caching methods.

For its operation, NC relies on having access to network, computation and storage resources throughout the network. This obviously entails that novel design approaches are made available to network engineers in order to allow them to truly disclose the potential capabilities of NC [12]. Network, computation and storage resources are also the subject of the Network Function Virtualization (NFV) research field. NFV and NC can indeed be seen as complementary

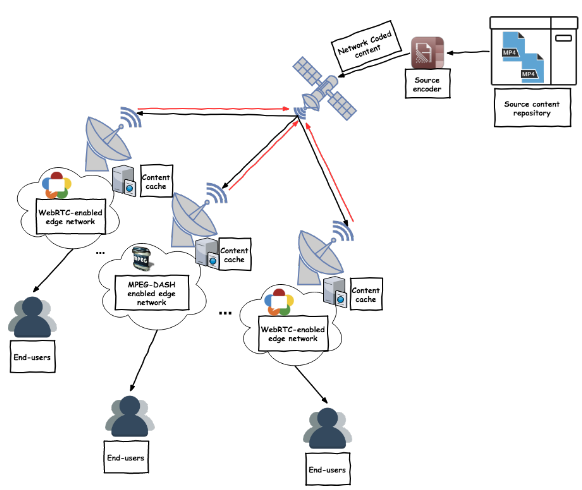

Fig. 2. SHINE architecture: high-level view

ways to address related challenges in the design of next generation network architectures and technologies.

NC departs from traditional networking technologies in that it looks at network flows as streams of physical packets, which are logically grouped from the network coding perspective. Such physical flows are then associated with proper mathematical models and can thus be the subject of computational manipulations. The above mentioned features of NC make it a perfect candidate for the integration within a virtualized framework of abstract entities/resources such as a virtual network or a so-called "network slice". Indeed, in the NC case we add "streams of packets" to the set of entities (namely, "network" and "resources") that can be abstracted and offered in a virtualized environment.

Digging a bit deeper into the details, if we look at NC as a functionality provided to the network, its virtualization consists of integrating ad hoc defined NC functional toolboxes into existing Network Function Virtualization frameworks. The relevant NC toolboxes can be summarized as follows:

- *Coding/Re-encoding/Decoding Functionality* (CRDF), which is associated with the application of NC-based in-network computations to the set of identified streams of packets;
- *Flow Engineering Functionality* (FEF), which deals with the dynamic adaptation of the available network resources in order to allow NC-based functions to be performed in an optimized way;
- *Physical Abstraction Functionality* (PAF), which is in charge of properly interacting with the abstracted view of the underlying physical (i.e., storage and computation) resources, as provided by the companion toolboxes.

3.1.1 Integration with ETSI NFV architecture

Fig. 3 shows our proposed Virtual NC Function (VNCF), integrated within the ETSI NFV architecture on top of the abstracted set of underlying physical system/network resources as part of the well-known Network Function Virtualization Infrastructure (NFVI).

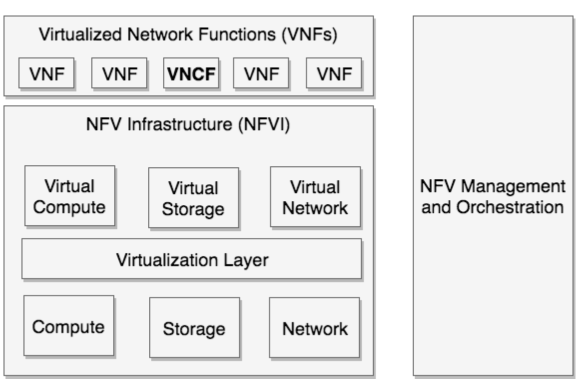

Fig. 3. ETSI NFV with VNCF

Clearly, the previously described FEF (Flow Engineering Functionality) toolbox needs to interact with NFV Management and Orchestration (MANO), VNF Manager (VNFM), and Virtualized Infrastructure Manager (VIM) components. The MANO's responsibility of orchestration of resources across VIMs, as well as management of network services life-cycle, perfectly fits the needs of the FEF and PAF toolboxes. The FEF can in fact obtain available network, connectivity and computation resources, as well as statistical information about the current network status (e.g., congestion, link failures, etc.) in order to dynamically optimize and fine-tune NC operation. Optimization may in turn result into proactive requests for the manipulation of the flows, as well as for the activation of relevant flow engineering policies. On the other hand, the FEF can also interact with the VIM in order to obtain the allocation, upgrade and release of NFVI resources.

For the sake of brevity, we are not delving here into the details of how we implement the above-mentioned concepts within SHINE. The interested reader is referred to [13] for an exhaustive description of the VNCF on a real virtualization platform with details on the software architecture and signaling model.

4 SHINE CACHING STRATEGIES

Caching approaches can be classified as either pull-based or push-based. Pull-based is usually associated with client-initiated caching. More in general, pull-based caching can be intended as the in-network caching during the actual transfer of a content from a server to the client. The push-based approach consists of a caching generally uncorrelated with the content delivery process. A given content is cached on a local server in the view of a future content request. Such a caching is compliant with the server-initiated approaches, but can be adopted also in some client-initiated scenarios.

4.1 Cooperative Pull-Based SHINE caching

Fig. 4 shows the overall SHINE network architecture, indicating the sequence of operations for a client to retrieve a multimedia content in case of a *cache miss*. For the sake of generality, we envisage three different SHINE client clusters, each physically connected to a different access point and a different SHINE edge cache. Two SHINE edge caches are connected to *Telco 1* access network, while the third one is connected to a different *Telco* network. This configuration highlights the SHINE potentiality to seamlessly work on top of different telco domains in a general configuration. We consider a content request generated by one of the clients from Cluster 1: **1. GET request**. This request is forwarded to the local DNS server, which is co-located with the SHINE edge node. Of course, if the requested content is cached in the SHINE local cache, the delivery process quickly and efficiently results into a local delivery. On the contrary, in case of cache miss, the SHINE local DNS server queries the local CDN name server (in the Telco 1 premises) to know the IP address of a CDN server (or of the origin server) having the requested content: **2. Request for hostname resolution**. The local CDN server takes in charge the request and finds out the most suitable CDN server capable to satisfy the request, by leveraging protocols and algorithms out of any SHINE operation. As soon as it is done, the CDN name server communicates to the SHINE local node the IP address of the selected server: **3. Reply: CDN server IP address**. Successively, the SHINE edge node sends the pending content request to the identified server: **4. GET request**. To

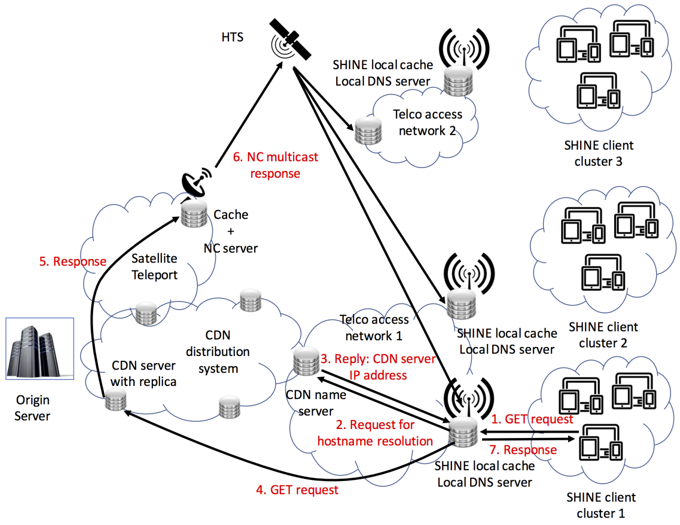

Fig. 4. Cooperative SHINE pull-based CDN architecture.

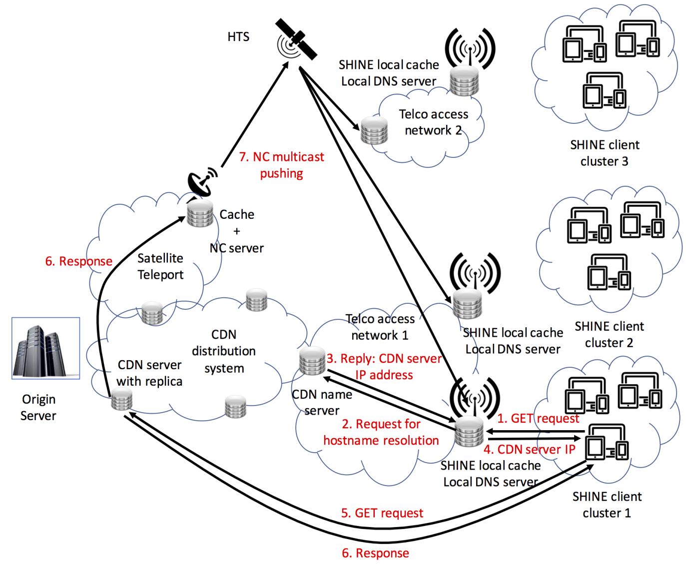

Fig. 5. Cooperative SHINE push-based CDN architecture.

fully implement SHINE concepts, the CDN server must be aware that the request is coming from a SHINE node and it must force the corresponding response towards the NC server in the satellite premises: **5. Response**. This feature can be implemented at the application layer in the CDN servers aimed at supporting the SHINE architecture. It is indeed the only actual modification required to allow for the adoption of the SHINE approach from CDN providers. Once content arrives at the NC server, this latter performs coding operations and forwards via multicast the coded response to all of the connected SHINE edge nodes: **6. NC multicast response**. The cooperative nature of the SHINE caching approach relies on the fact that content requests from a client cluster are served jointly with other requests coming from other SHINE edge caches, by leveraging NC techniques. Finally, the SHINE local cache decodes multicasted content and eventually satisfy the client's request: **7. Response**.

The main advantage of this approach relies on a significant offloading of terrestrial core networks by fully exploiting satellite multicast transfers and local caching. The drawback is the potential increase in the response delay in case of cache-miss. In fact, the overall latency experienced by the client includes two additional contributions with respect to a "traditional" terrestrial only CDN delivery: a GEO satellite one-way delay and NC processing time at the NC server.

4.2 Cooperative Push-Based SHINE caching

Push-based caching envisages a pro-active transfer of contents towards local caches, uncorrelated with the actual content delivery process. Assuming the same circumstances of the previous example (SHINE local cache miss), Fig. 5 shows the operations performed with the push-based approach. It is important to remark that the overall CDN configuration and architecture is actually unchanged. In addition, the first 3 steps are the same as with the previous caching strategy.

Differently from the pull-based approach, the SHINE local node forwards the IP address of the CDN server storing the requested content directly to the client: **4. CDN server IP**. Once obtained the server IP address, client contacts it to require the target content over the ISP connection: **5. GET request**. The server replies by directly sending the content to the client: **6. Response**. Simultaneously, it also forwards a replica of the content to the NC server to start content pushing in the SHINE local cache: **7. NC multicast pushing**. In fact, the response directly sent to the client through the terrestrial connection does not "pass through" the SHINE local cache. On the contrary, the NC server prepares a coded version of the content to be pushed via satellite to the SHINE local cache. Most likely, the content will be successfully cached on the SHINE local cache after its actual reception at the client side. With this approach, terrestrial traffic offloading depends only on possible SHINE local cache hits, while cache misses are still addressed via terrestrial connections. On the other hand, the push-based approach can still bring the benefits of multicast network coding, while minimizing the latency experienced by the client in case of cache misses.

5 SHINE-enabling Integration Business Models

The actual implementation of SHINE on current communication infrastructures requires a correct integration among one of the applicable satellite configurations (Section 2), the VNCF component (Section 3.1) and the overall network architecture required for caching (Section 4). At the system level, SHINE involves a number of stakeholders, including of course a satellite operator aiming at fully exploiting the performance gains deriving from the adoption of native multicast in a high capacity network, as well as the flexibility allowed by HTS multi-beams satellite platforms.

SHINE edge caching can be either integrated in a mobile Telco provider access network or provided as a direct-to-home (DTH) service. In the former configuration, users subscribe a commercial agreement (B2C) with a Telco provider implementing the SHINE caching solution in its access network. In case of a SHINE DTH service, the SHINE local cache must be integrated in the user terminal, which needs to be enhanced with a satellite interface. In any case, SHINE DTH caching or Telco-CDN provider must interwork with a satellite operator (B2B relationship), in order to set up an integrated satellite multicast downstream. Satellite integration level, technology selection and configuration flexibility strictly depend on the tailored application of one of the service delivery models introduced in Section 2. Furthermore, SHINE caching can be integrated with terrestrial CDNs for a more efficient content retrieval in case of local cache miss. In this case, the only requirement to enable SHINE operations is to allow redirection of the content to the NC server, as envisaged in the caching schemes proposed in Section 4.

The NC server lies on a NFVI deployed either within the Satellite Network Operator (SNO) network or in an Internet-based caching reseller's premises, depending on the applied satellite delivery model (see sub-section 5.1). Last but not least, a number of independent SHINE services can be supported by multiple virtual server instances dynamically configured and deployed to account for different caching services, each exploiting the global coverage with regional granularity provided by the HTS multi-beam architecture.

5.1 Evaluation of the satellite delivery models

SHINE integration with a satellite platform is of paramount importance if one wants to exploit network coding multicast, which is indeed the main pillar of the proposed caching solution. The selection of the most suitable satellite delivery model is a key aspect for the practical configuration of the overall caching service because of its impact on several networking and architectural aspects. Hereafter, we go back to some of the models presented in Section 2, with the aim to identify pros and cons in supporting SHINE operations.

A first option foresees the adoption of the *Vertical Integrated Model*, where commercial satellite solutions are integrated to connect the *source encoder block* with SHINE-enabled *access networks*. Routing and multicast operations are then fully managed at IP and upper layers and the SHINE NFVI is out of the satellite domain. A potential drawback resides in the fact that both IP multicast support and satellite Network Address Translation (NAT) need a return link via satellite for both client-initiated multicast registrations and content requests (i.e., cache miss) notifications. This represents a significant architectural constraint imposing the adoption of a bi-directional satellite technology and leading to a double-hop over satellite in case of cache misses. Indeed, multicast support and multicast-friendly firewalls may not be activated over satellite and unmanaged connections over public networks do not allow control over QoS. This makes SHINE performance configuration-sensitive and ineffective when IP multicast is disabled. Thus, the resulting delivery model could be suitable only when SHINE local caches are behind a single satellite terminal (satellite point-to-point transmissions) and scalability is not required (few users). The advantage here is a fast service deployment time with no CAPEX (with regard to the satellite domain) for CDN providers. With the *Managed Service Model*, the SHINE VNCF, as well as other VNFs (i.e. NAT, firewall, multicast router), are implemented within the SNO premises and can be properly configured by the SHINE caching provider. Then, this model guarantees to have all the required network services activated and properly configured to enable satellite multicast and to control the target transmission performance, up to the edge networks. On the other hand, working at the network layer, the satellite return link is still necessary to support operations to dynamically join target multicast groups. The *VNO model* further extends the flexibility of the previous one by allowing a deeper control over the satellite network configuration, with the possibility to act on lower layer settings, such as enabling ad-hoc channels for multicast streaming or multicast management linked to user terminals. This aspect specifically allows to leverage a broadcast downstream-only satellite component. The only constraint is related to the satellite communication technology, which is in this case dictated by the SNO. Last, the *Virtual Infrastructure Model* enables the maximum degree of flexibility in the integration and configuration of the satellite component, by adding to the VNO model the possibility to select, customize and deploy the Hub infrastructure and, accordingly, the satellite terminal technology. The cost of such an approach concerns high upfront CAPEX costs to build up an ad-hoc satellite communication platform, making this approach viable for large-scale services involving a very high number of users.

6 Conclusions

The deployment of the SHINE solution brings innovative mechanisms for the secure distribution of real-time multimedia information across a combination of unicast and multicast communication networks. The main novelty resides on the application of a "network coding" technique over satellite multicast-enabled transmissions. In addition, SHINE caching is deployed as a VNF in compliance to the ETSI NFV architecture fostered in 5G networks. In this context, an efficient integration with the satellite communication infrastructure is of paramount importance, in order to fully exploit both multicast and large coverage features. The analysis carried out in the paper highlights that SHINE can be in principle enabled atop all the possible satellite delivery models, although with different implications. In particular, a holistic performance and functional evaluation indicates the *Virtual Infrastructure model* as the "optimum" towards 5G deployment because it supports a complete virtualization, orchestration and control of all the requested functions. This implies a migration from the legacy, vendor-constrained satellite platforms to fully virtualized, 5G-compliant component deployments, which translates into consistent CAPEX costs within the satellite domain. In a shorter time frame, and with progressively lower CAPEX costs, the SHINE service can be integrated by adopting other possible satellite service delivery models, still enhancing the overall architecture with its secure and efficient local caching capability. Nonetheless, these approaches offer a limited control over the satellite configurations, with a negative impact on the overall architecture (satellite return link could be mandatory, NC server on public networks), achievable performance (i.e. double hop over satellite, best-effort transfers) and OPEX costs when relying on existing commercial offers. Currently, the SHINE project partners are finalizing a prototype implementation that will allow an exhaustive test campaign aimed to both validate the whole caching process and assess performance improvements with respect to traditional terrestrial-only CDN solutions.

REFERENCES

[1] Open Networking Foundation: *SDN Architecture, Issue 1, June 2014*

[2] ETSI GS NFV 002, v1.1.1: *Network Functions Virtualisation (NFV); Architectural Framework* , 2013.

[3] A. Abdelsalam et al., *Analysis of bandwidth aggregation techniques for combined use of satellite and xDSL broadband links*, (2018) International Journal on Satellite Communications and Networking. pp.1 - 15.

[4] M. Corici et al., *Assessing satellite-terrestrial integration opportunities in the 5G environment*, [On-line] Available on: https://artes.esa.int/sites/default/files/Whitepaper, Sept. 2016.

[5] ETSI GS MEC 003, v1.1.1: *A Mobile Edge Computing (MEC); Technical Requirements*, 2016.

[6] SHINE project: *Secure Hybrid In Network caching Environment*, URL: https://artes.esa.int/projects/shine-secure-hybrid-network-caching-environment

[7] D. Bettinger, "Maximizing the HTS Opportunity: Leveraging New Satellite Architectures and Business Models to Grow," *Asia Pacific Satellite communications council (APSCC)*, APSCC 2013 Q3 newsletter, ISSN 1226-8844.

[8] "MPEG-DASH – Dynamic Adaptive Streaming over HTTP," ISO/IEC Standard, ISO, Standard ISO/IEC 23009.

[9] S. Loreto and S. P. Romano, "How far are we from WebRTC-1.0? an update on standards and a look at what's next," *IEEE Communications Magazine*, vol. 55, no. 7, pp. 200–207, 2017.

[10] S. P. Romano, C. Roseti, A. M. Tulino, "SHINE: Secure Hybrid In Network caching Environment" in Proceedings of The International Symposium on Networks, Computers and Communications (ISNCC), 19-21 June 2018, Rome, Italy

[11] "Digital Video Broadcasting (DVB); second generation DVB interactive satellite system (RCS2); part 2: Lower layers for satellite standard," ETSI EN 300 421 (V.1.1.2), ETSI, European Standard ETSI EN 301 545-2 (V.1.1.1).

[12] M. Ji et al. , "Order-optimal rate of caching and coded multicasting with random demands," IEEE Transactions on Information Theory, vol. 63, no. 6, pp. 3923 - 3949, June 2017.

[13] S. P. Romano and F. Giangrande, "On the use of Network Coding as a Virtual Network Function in satellite-terrestrial CDNs", in Proceedings of The First International Workshop on Resource Slicing for Future Clouds and Networks, 15-19 April 2018, Honolulu, HI, USA