SDN/NFV based Caching Solution for Future Mobile Network (5G)

Yaning LIU, Jean Charles Point
JCP-Connect, Rennes, France
{yaning.liu, pointjc}@jcp-connect.com

Konstantinos V. Katsaros, Vasilis Glykantzis
Intracom Telecom, Athens, Greece
{konkat, vasgl}@intracom-telecom.com

Muhammad Shuaib Siddiqui, Eduard Escalona
Fundacio I2CAT, Barcelona, Spain
{shuaib.siddiqui, eduard.escalona}@i2cat.net

[1]***Abstract*—Caching content locally at the edge of the network and managing these caches by SDN/NFV based technologies are able to satisfy increasing data traffic demand in 5G. The virtualization of caching functionalities allows network operators to deploy caching services with advanced features like flexibility, dynamicity and auto-scalability, and provide caching services to service providers or virtualized network operators over the same common infrastructure. The caching management system provides two levels of operations: i) at the global level managed by the Infrastructure Provider applying to all the tenants or virtualized network operators; or ii) at the tenant-specific level managed by the specific tenants. Our caching solution opens an entirely new space of business opportunities for network operators, service providers and content providers.**

Keywords—SDN/NFV, Virtulization of Caching Service, Multi-tenancy.

I. Introduction

Internet traffic keeps growing at pace as a consequence of the steadily increasing number of users and the adoption of new bandwidth-intensive services (such as video services) by end users. According to recent studies [1], global IP traffic has increased more than fourfold in the past five years, and is expected to increase a further threefold over the next 5 years.

To meet the growing bandwidth demands of mobile users, commercial Long-Term Evolution (LTE) networks are being deployed to significantly increase the bandwidth and reduce the latency experienced in the mobile network. Though LTE improves the network performance between end users and the mobile network, the service latency still highly depends on the distance between the data centre where content is usually stored and the exchange point where the mobile network connects to the Internet. Ensuring low service latency is crucial to satisfying the QoS of current applications, especially for latency-sensitive applications such as audio and video services. Moreover, LTE network backhaul bandwidth will be heavily consumed by duplicated data streams when content (such as high-popular videos) is requested simultaneously and frequently. Caching has been proved to be an efficient solution to reduce this duplication, improve the efficiency of network utilization and the data retrieval performance. The data access delay is significantly reduced since data access requests can be served from the local cache. The deployment of distributed metro/access caches further enables a more scalable architecture, since the content is replicated in several locations of metro/access networks The authors in [2] analysed the gains of Hypertext Transfer Protocol (HTTP) content caching at the location of Serving Gateway (SGW) in an LTE Wireless network, and reported that 73% of the data volume and around 30% of the responses are cacheable. Mobile network caching could be a cost-effective solution to improve the service latency and reduce the mobile backhaul traffic by replicating popular and frequently demanded content in Internet Protocol (IP)-based 3G/LTE network elements closer to mobile users.

In this paper, we revisit caching in the context of future mobile (5G) networks, focusing on the emerging capabilities for software defined networking (SDN), and the support of network functions virtualization (NFV). We build on and extend these capabilities in a caching solution aimed at providing an open accessible, highly configurable, and transparent in-network caching service. Our target is to improve the content distribution efficiency by enabling the opportunistic caching and/or pre-fetching of content across the different points in the virtualized network infrastructure. Virtualization of the caching functionality allows the dynamic deployment of network caches, elasticity, flexible cache management for subscribers according to their requirements and network status, while it facilitates the establishment of new business relationships, such as cache peering agreements between virtual network operators (VNOs).

The proposed caching solution is developed as part of CHARISMA project [3]. The description of the caching solution is explained based on the CHARISMA network architecture briefed in Section III. The paper is organized as follows. Section II presents the requirements and opportunities of a next generation caching solution. Section III describes the CHARISMA caching solution design and operations, and finally paper concludes in Section IV with some insights regarding future work.

[1] This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 671704.

II. Requirements and Opportunities

A. Technical Requirements

In this section, we discuss few technical requirements, for both conventional caching as well as for virtualized caching solutions which serve as the main drivers for the solution proposed in this paper.

Resource Elasticity: As the vast majority of current caching infrastructure relies on dedicated hardware, the cache placement problem (i.e., which network locations caches should be deployed at) has been traditionally addressed as a one-off long-term decision pertaining to network planning and dimensioning. The current state of the art therefore only foresees the formulation of problems related to the static placement of the cache locations, subject to long term observations of the demand, usually in the form of a facility location problem [4]. However, virtualization introduces the aspect of time, decoupling the deployment of a cache instance from any physical deployment constraints. This opens new challenges in also identifying when a cache is mostly needed in a certain network, subject to workload dynamics and estimations of latency and traffic savings. The elasticity of the virtualised resources enables the efficient management of the available resources, allowing Network Operators (NOs) to utilize the leased resources when there is actually a well-justified need for it.

Reduce Cache Miss Delays: The flexibility offered by the virtualization enables a new opportunity regarding the handling of traffic in the context of virtual caching, with a particular focus on achieving low delays. This opportunity stems from the observations that not all traffic can benefit from caching (e.g., conference calls), and that steering traffic through a cache results in delay overheads related to the protocol stack traversal and the cache index lookup operation. While such delays are acceptable in the case of a cache hit, where the latency to retrieve the content from its origin is avoided, the same does not hold for a cache miss, since in this case, latency increases as compared to the case of not employing a cache. An important challenge then relates to the identification of the traffic amenable to caching and the corresponding steering of the traffic through the cache or not i.e., bypassing caches when necessary.

B. Business Oppotunities

In the 5G paradigm, caching plays a significant role as it allows an optimized network usage and a low latency delivery of content. Beyond allowing acceleration in the network for content delivery and edge applications, the virtualized and decentralized caching solution opens the door to new business opportunities both for operators and content providers.

The current business models in content delivery have been characterized by CDN networks and free installation of cache servers. CDN network provide a series of mirrors with global coverage to content providers. They are paid to ensure low latency. Since 2011 OTT providers like Google and Netflix have started to deploy edge servers for free in telecom operators networks offering a comprise solution – they pay for servers (H/W and S/W) and remotely manage them. The telecom operators have in return provided Google and others with rack spaces, power and GE ports for free [5]. The 5G era offers a new set of possibilities in which operators have the chance to strengthen their position versus content providers. Possible business models include: 1) Freemium: the current symbiosis between network operators and OTT/content providers can continue by operators offering free access to 5G network to major OTT providers as this can drive a faster adoption of 5G. 2) Paid access to network resources: Features as tenancy open a new set of OTT/content providers that can arise fed by the existence of multiple VNOs. In a future when 5G connectivity is ubiquitous and relatively cheap, a VNO can assemble quickly needed resources and activate NFV-based caching service according to the needs of new OTT Providers.

CDN providers can also easily migrate into the position of VNOs, managing the distribution and activation of caching resources. An entirely new category of OTT content providers can appear targeting specifically 5G mobile users. Using NFV-based caching solutions offers to content providers the ability to use agile caching of personalized and secure content which will be isolated from other content providers keeping all sensitive data insulated from other participants. The increased competition and potential entry of new players will make the "content delivery" space much more dynamic while operators can keep the control and have a chance to increase their earnings rather than being largely sidestepped.

At the same time, transparent opportunistic caching is also considered as an acceleration technique to be widely adopted by VNOs in their effort to reduce traffic overheads and improve QoE. The NFV capabilities of flexible placement and auto-scaling facilitate a simple business model, where content is cached at zero cost to content providers, subject to user demand. Standardization activities have already engaged in the development of mechanisms overcoming the emerging considerations related to encrypted traffic (HTTPS) and intellectual property rights (IPR), by allowing content provider provide explicit consent for the content to be cached[10][11]. In this context, as VNOs share the same infrastructure, new opportunities arise for the establishment of cache peering relationships, where VNOs exchange transparently cached content (see also Section III-C). The emerging business model then follows established practices in the domain of CDN interconnection [14]. Peering VNOs can exchange content for free in a mutual beneficial manner, while accounting mechanisms can ensure volume-based payments when the use of the peering link is asymmetric.

III. CHARISMA Caching Solution Design

A key architectural innovation to the CHARISMA project is the use of a hierarchical and distributed approach to the network architecture. That is to push out network intelligence, processing and routing towards the end-users. This has various advantages, including low latency (i.e. traffic is only routed via the lowest common aggregation point in the network, rather than always via the CO), lower access times for data (i.e. data is cached at various caching locations, closer to the end-user), and also the ability to distribute traffic more evenly across the network (i.e. load balancing of the network is more easily performed for such a hierarchical and distributed network architecture.) The CHARISMA architecture has been divided up into 4 hierarchical levels, with each active node at each level called a Converged Aggregation Level (CAL). Located at each CAL is an Intelligent Management Unit (IMU), which can either contain physical networking functions (PNFs) of computing, networking and storage resources, or alternatively, these resources can be virtualized. Each of the CAL is controlled, managed and orchestrated by a control, management and orchestration (CMO) platform that allows the network infrastructure owner to offer infrastructure slices to virtual network operators (VNOs), which in turn can offer multiple applications, services toward their customers. CHARISMA caching solution, including virtualized caching resources, is provisioned, managed and orchestrated via the CMO platform. The interested readers can refer to [12] and [13] for more details on CHARISMA network architecture and the CMO platform, respectively.

A. *CHARISMA Caching System Architecture*

In order to meet the requirements of multi-tenancy and low latency of the CHARISMA approach, the CHARISMA caching system implements virtualization of the caching functionality to enable the deployment of in-network caching as software functions running directly over the network commodity hardware. Virtual Caches (vCaches) running as a VNF is able to be dynamically initialised and managed by the orchestrator according to the needs of the network or customers. The system provides SDN/NFV based virtual caching services to service providers (SPs) or Virtualized Network Operators (VNOs) over the same common infrastructure. A virtual Cache Controller (vCC) running as a VNF is created by the orchestrator along with the initialization of caching services for a VNO. It allows the VNO autonomously managing and configuring vCaches allocated to them. In our system, each VNO is assigned by one vCC and several affected vCaches with required resources like network bandwidth, CPU, memory and hard-disk storage. The vCaches with caching and prefetching functionalities can be provided in the different levels of the network accroding to the requirements of the VNO. A vCC is created and allocated to a VNO that is able to configure and manage its vCaches without intervention from the whole CHRISMA platform.

The CHARISMA management system for caching services provides two levels of operations: i) at the global level managed by the Infrastructure Provider applying to all the tenants/VNOs; or ii) at the tenant-specific level managed by the specific VNO. In the global management, the CMO is responsible for the creation/removal/configuration of virtual instances for each VNO with policies defined in Cache Policy Manager (CPM), including a virtual Cache Controller (vCC) along with one or more virtual Caches (vCaches). The deployment of the vCaches such as when/where/how to create vCaches will be intelligently decided by the policy defined in CPM through considering VNOs' requirements, network performance and the status of caching resources. The tenant-specific level management is operated by the vCC assigned specifically to a VNO. The vCC is able to configure the caching and prefetching functionalities running in the vCaches allocated to the VNO.

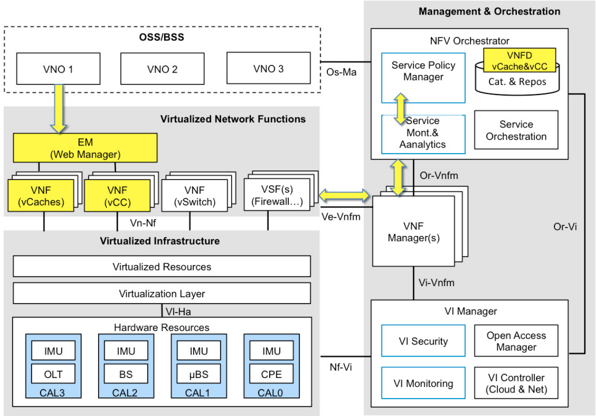

Figure III-1: CHARISMA Caching System Architecture

B. *CHARISMA Caching/Prefetching Service*

The in-network caching system can be managed in both proactive and reactive ways. The proactive approach manages the in-network caching by means of pushing content into the caching devices. This pushing operation could be particularly useful in live audio/video streaming delivery since it is easy to anticipate the users' request. The pushing method also allows for prefetching of content that a user could potentially request onto a cache located in the network at a location that a mobile user will move to. The popularity of the content and the users' social behaviour can also be used to improve the accuracy of predicting the prefetched content. The CHARISMA caching system provides to VNOs both caching and prefetching functionalities that are implemented on vCaches and managed by the vCC. The vCC is able to configure the caching and prefetching programs running in the vCaches like service port, caching/prefetching algorithms. Hence, the vCC is able to help improve user QoS by configuring caching/prefetching settings.

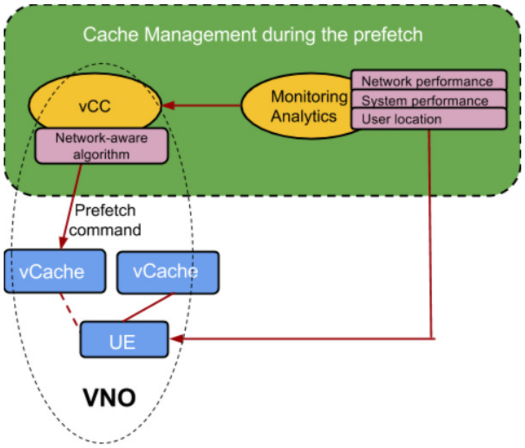

Figure III-2: Network-aware Prefetching Scenario

Figure III-2 describes a network-aware prefetching scenario where a mobile user switches from one vCache hosted network device to another one. a network-aware prefetching policy can improve QoS by downloading the content that the user will most probably request such as user current requested content or popular content, etc., closer to the user, even as the user's network performance is deteriorating. The prefetching is performed as the vCC detects a network handover on the User Equipment (UE) that is an end user of a VNO. The vCaches and vCC have been dedicatedly assigned to this VNO. The Service M&A module collects the information of the network performance (such as wireless signal strength, throughput and loss rate, etc.) at the UE. The vCC periodically communicates with the Service M&A module. The UE connecting to a network device close to the vCache on the right will switch to another network where the left vCache has been deployed. The vCC is able to detect this network switch by the information of the networks status at the UE as provided by the Service M&A module. Subsequently, the vCC triggers a prefetch that is performed on the left vCache. As soon as the UE makes the switch, the UE's request can be directly served by the left vCaches.

C. *CHARISMA Cache Peering in Multi-tenancy Scenarios*

In CHARISMA, the virtualised character of resources and the potential collocation of vCaches of different VNOs on the same IMU, incentivize the establishment of peering relationships between collocated vCaches.In cache peering, a cache miss results in a potential request of the requested item from peering caches. Whether a request is actually issued depends on content availability information exchanged by the peering caches either proactively (e.g., [8]) or reactively i.e., upon the cache miss (e.g., [9]). The co-location of vCaches facilitates the exchange of both content availability information, and of the content itself (in terms of both low latency and/or high bandwidth).

Figure III-3 shows a simplified vCache peering setup. The vCC is omitted for simplicity. The figure shows a typical OpenStack[2] setup : a Network Node forwards traffic to the various VNFs hosted at the Compute Hosts. There, internal (virtual) switches deliver traffic to VNFs, ensuring traffic isolation between tenants. In the example, vCaches of VNOs #1 and #2 are instantiated on the same Compute Host. OpenStack-enabled traffic isolation forbids communication between the two vCaches. The following steps are then followed to support cache peering:

1. A VNO creates a *shared* OpenStack network and attaches its local vCache to it.
2. The VNO that created the shared network employs the Role-Based Access Control feature of OpenStack[3] to grant its peer VNO(s) with access to the shared network.
3. The peering VNO(s) attaches its (their) vCaches to the shared network.
4. The application-level components of the vCache instances (e.g., Squid[4]), are configured (by either the EM or the vCC) as cache siblings.
5. The application-level components of the vCache instances are configured (by either the EM or the vCC) to apply certain rate limits so as to avoid the overconsumption of local resources due to peering requests.
6. The application-level components of the vCache instances are also configured with authentication credentials to enable peering interactions only with authenticated vCaches.

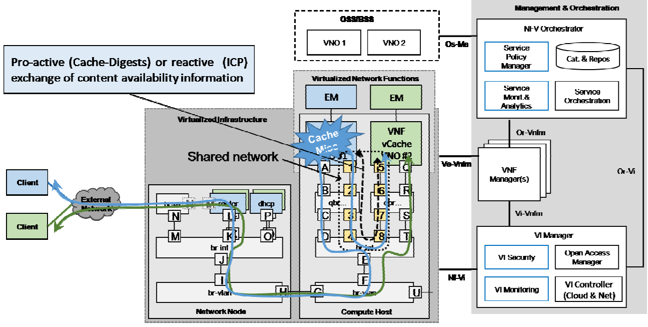

Figure III-3: vCache peering in multi-tenancy (open-access scenarios)

1) Controlling resource sharing. vCache peering relationship neccesitates the firm control of the resources devoted to the exchange of cached content, so as to minimize the impact of the peering link on caching service to local subscribers. Figure III-4 shows a more detailed setup, towards this end. Since load on caches may substantially vary through time, multiple vCache instances may exist per VNO, realizing elasticity. A load balancer (LB) VNF balances the load across the available instances. A separate *peering* instance is dedicated to the peering link, not participating the local caching service, thus isolating the two service domains i.e., local and peering. The content required for the peering relationship is asynchronously prefetched from the local instances to the peering instance(s) as instracted by the vCC (not shown for simplicity). Prefetching can take place during low load periods, thus minimizing effects on the local service. The prefetching mechanism further allows for the intelligent selection of content i.e., VNOs can exchange content availability information that allows them to prefetch content usefull for their peers e.g., content not available at the peer.

D. *Traffic Optimization for vCaching*

CHARISMA further enables the optimised handling of traffic passing through virtualized caches. The objective is to take advantage of SDN capabilities and the availability of virtualized caches, so as to decide which traffic flows get through to the vCaches. The rationale of the designed solution builds on the observation that not all traffic leads to a cache hit and in this case, traffic unnecessarily suffers a delay overhead owing to the traversal of the vCache. Cache misses result in request packets traversing the protocol stack of the network node up to the vCache, only to return back to the network node on their way to the original content server. The content then traverses the network node and compute host's protocol

[2] https://www.openstack.org/

[3] http://docs.openstack.org/newton/networking-guide/config-rbac.html

[4] http://www.squid-cache.org/

stacks once more in order to reach the vCache, before it follows the reverse route towards the requesting client. This requires the traversal of the virtualised infrastructure by 4 times per cache miss. If we consider that the majority of content items is typically of a low popularity (i.e., Zipf-law), it follows that this delay overhead is suffered for a substantial proportion of content requests.

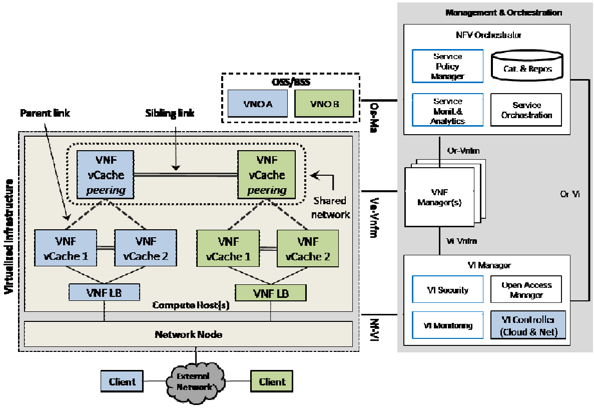

Figure III-4: Advanced vCache peering setup

The CHARISMA solution aims to identify those traffic flows that are likely to lead to a cache hit or miss, and therefore to subsequently decide on bypassing the vCache (or not, as the case may be). The procedure comprises of the following steps:

1. The vCC inspects the cache access logs of the vCaches. One entry is created per received request including: 1) origin server's URL; 2) result code = HIT/MISS. A HIT entry shows that the URL is related to cached content; consequently the corresponding IP address is a network destination likely to lead to a cache hit. A MISS entry shows that the requested item was not cached, which subsequently triggers the content to be fetched from the content origin server. If this URL does not appear in subsequent HIT entries, the item is considered unpopular. The corresponding IP address is therefore unlikely to lead to a cache hit, so future traffic flows to this IP destination will bypass the vCache. All identified IP destinations are ranked in order of cache hit/miss likelihood, based on the number of HIT/MISS entries they associate with.
2. The identified IP destinations are delivered by the vCC to the VNFM, annotated to indicate if the corresponding flows should bypass the vCache or not.
3. The VNFM delivers the annotated IP destinations to the SDN controller of the VIM.
4. The SDN controller transforms the received information into flow rules and corresponding OpenFlow control messages, applying the rules to the `br-int` switch. As the capacity of the `br-int` switch is finite, and shared between the various tenants, our design envisions intelligent rule placement policies, taking into account both the ranking of the destination IPs (step 2) and the fair sharing of the switch memory.

IV. Conclusion

In this paper, a SDN/NFV based caching solution has been proposed in the concept of 5G future mobile networks. Through dynamically deploying the virtualized caching functionalities including vCache and vCC, a configurable and efficient caching service can be dynamically and flexibly provided to the network operator's customer like end users, service providers and VNOs, which opens the door to new business opportunities. Technical requirements and business opportunities have been investigated and analyzed. We further present our caching system architecture with caching and prefetching services, the cache peering feature and traffic optimization that are managed under SDN/NFV environment. The caching system is currently under the development. In our future work, we plan to improve the system by exploring and adding intelligence on caching/efficient algorithms, CPM, and caching peering strategy, etc.

References

[1] Cisco VNI report, "Cisco Visual Networking Index: Forecast and Methodology, 2014–2019" 27, May 2015.

[2] B. A. Ramanan, L. M. Drabeck, M. Haner, N. Nithi, T. E. Klein and C. Sawkar, "Cacheability analysis of HTTP traffic in an operational LTE network," Wireless Telecommunications Symposium (WTS), 2013, Phoenix, AZ, 2013

[3] CHARISMA 5G. Online. http://www.charisma5g.eu/

[4] Qiu, Lili, Venkata N. Padmanabhan, and Geoffrey M. Voelker. "On the placement of web server replicas." INFOCOM 2001.

[5] Limbach, F., "Cooperative service provisioning with OTT players – An explorative analysis of telecommunication business models". 25th European Regional ITS Conference Brussels, Belgium, 2014.

[6] Georgopoulos et al. "Cache as a Service: Leveraging SDN to Efficiently and Transparently Support Video-on-Demand on the Last Mile" 23rd International Conf. on Computer Communication and Networks, 2014

[7] A. Chanda, C. Westphal and D. Raychaudhuri, "Content based traffic engineering in software defined information centric networks," IEEE Conference on Computer Communications Workshops, Turin, 2013,

[8] Li Fan, Pei Cao, Jussara Almeida, and Andrei Z. Broder. "Summary cache: a scalable wide-area web cache sharing protocol". *IEEE/ACM Trans. Netw.* 8, 3 (June 2000), 281-293.

[9] D. Wessels and K. Claffy, "ICP and the Squid web cache," in *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 3, pp. 345-357, Apr 1998

[10] M. Thomson, G. Eriksson, C. Holmberg, "An Architecture for Secure Content Delegation using HTTP," Internet Draft, draft-thomson-http-scd-00, December. 2016

[11] G. Eriksson, J. Mattsson, N. Mitra, Z. Sarker, "Blind cache: a solution to content delivery challenges in an all-encrypted web," White Paper, Ericsson, 2016

[12] CHARISMA Deliverable: D1.2: Refined architecture definitions and specifications. Online. [http://www.charisma5g.eu/wp-content/uploads/2017/01/CHARISMA-D1.2_final_v2.pdf]

[13] CHARISMA Deliverable: D3.2: Initial 5G multi-provider v-security realization: Orchestration and Management. Online. [http://www.charisma5g.eu/wp-content/uploads/2015/08/CHARISMA-D3.2_v1.0.pdf]

[14] Bertrand, G., Ed., Stephan, E., Burbridge, T., Eardley, P., Ma, K., and G. Watson, "Use Cases for Content Delivery Network Interconnection", RFC 6770, November 2012. http://www.ietf.org/rfc/rfc6770.txt