Follow Me Cloud: Interworking Federated Clouds and Distributed Mobile Networks

Tarik Taleb, NEC Europe
Adlen Ksentini, IRISA/University of Rennes 1

Abstract

This article introduces the Follow-Me Cloud concept and proposes its framework. The proposed framework is aimed at smooth migration of all or only a required portion of an ongoing IP service between a data center and user equipment of a 3GPP mobile network to another optimal DC with no service disruption. The service migration and continuity is supported by replacing IP addressing with service identification. Indeed, an FMC service/application is identified, upon establishment, by a session/service ID, dynamically changing along with the service being delivered over the session; it consists of a unique identifier of UE within the 3GPP mobile network, an identifier of the cloud service, and dynamically changing characteristics of the cloud service. Service migration in FMC is triggered by change in the IP address of the UE due to a change of data anchor gateway in the mobile network, in turn due to UE mobility and/or for load balancing. An optimal DC is then selected based on the features of the new data anchor gateway. Smooth service migration and continuity are supported thanks to logic installed at UE and DCs that maps features of IP flows to the session/service ID.

Mobile traffic is increasing at a tremendous pace, exceeding far beyond the original capacities of mobile operator networks. This huge amount of mobile traffic is associated with a wide plethora of emerging bandwidth-intensive mobile applications popular among an ever growing community of mobile users. The challenge presented by all of this mobile traffic stems particularly from the fact that current mobile networks are highly centralized, leading to high demand on central locations due to backhauling of all data traffic, to dramatic increases in bandwidth requirements and processing load resulting in undesirable bottlenecks, and, last but not least, to long communication paths between users and servers. The effects are wasting core network resources, leading to undesirable delays, and ultimately resulting in poor quality of experience (QoE) for users.

A straightforward solution to these issues may consist of having operators invest in speed or upgrade their core network nodes to comfortably accommodate traffic peak hours of these emerging bandwidth-intensive mobile applications. While this is technically and technologically possible, it economically represents a significant challenge for operators, particularly due to the fact that the average revenue per user (ARPU) is not growing as quickly as traffic demands, particularly given the trend toward flat rate business models. There has thus been a need for cost-effective solutions that can help operators accommodate such huge amounts of mobile network traffic while keeping additional investment in the mobile infrastructure minimal. In addition to application-type-based traffic admission control techniques (e.g., throttling video traffic), an important solution consists in selective IP traffic offload (SIPTO) as close to the radio access network (RAN) as possible [1]. The key enabler of efficient SIPTO is to place data anchors and mobility gateways close to RANs, essentially leading to a relatively decentralized mobile network deployment [2].

On the other hand, cloud computing is gaining great momentum. Its market is expanding at a high speed, thanks to the multiple features it supports (e.g., multi-tenancy support, pay as you go, elasticity, and cost-efficient scalability) and the new business models it provides based on infrastructure sharing (infrastructure, platform, software as a service — IaaS, PaaS, and SaaS). In the telecommunications area, cloud computing has been gaining lots of attention. Indeed, there are already many telcos and carrier providers deploying cloud-based services; some deployments are only for internal use, whereas others are being sold as a service. The fast growing business of clouding computing is calling for distributed regional data centers (DCs) [3, 4], forming so-called federated clouds.

Putting these two observations together, cloud providers are distributing their DCs due to growing business. As for mobile operators, they need to decentralize their networks to cope with the growing number of smart phones and associated bandwidth-intensive services. The expected outcome network architecture is depicted in Fig. 1. Indeed, the figure shows the case of a decentralized mobile network architecture whereby core network gateways such as packet data network gateways (PDN-GWs) and serving GWs (S-GWs), in the context of the Evolved Packet System (EPS), are geographically distributed. Also shown is a federated cloud consisting of multiple regional DCs, geographically distributed and interconnected.

In such decentralized mobile networks, the main objective of any mobile operator behind SIPTO is to ensure an optimal mobile connectivity service; that is, a user equipment (UE) device shall always be connected to the optimal data anchor

0890-8044/13/$25.00 © 2013 IEEE

and mobility gateways such as PDN-GWs and S-GWs. However, it is very likely to have a UE device connected to an optimal data anchor gateway (as per its current location) but accessing a mobile service from a distant DC in a distant location (e.g., UE being in location 2, having its data anchored at P-GW2 but receiving service from DC 1). This intuitively results in inefficient mobile connectivity service given the absence of optimal end-to-end (E2E) connectivity. The objective of this article is to enable a user to always be connected to the optimal data anchor and mobility gateway, and to access its data and/or service from the optimal DC, that is, geographically/topologically nearest (or in any other metric, e.g., load and processing speed) DC. Furthermore, this optimal E2E mobile connectivity shall be ensured during the entire movement of the user. It shall be noted that when the notion of "optimal" or "better" PDN-GW/PDN connectivity is used, this is always meant in comparison to a PDN-GW to the same cloud network to which the UE is already connected. The detailed criterion for optimality is defined by operator policy, but typically may be derived from geographical proximity (to the UE) or load.

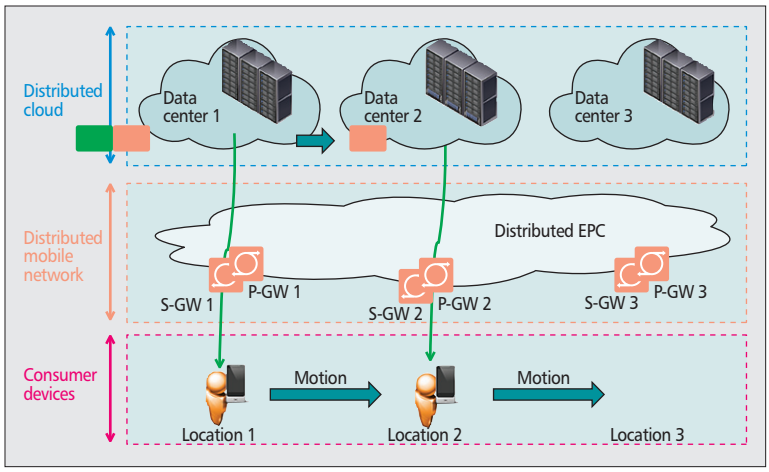

Figure 1. *Distributed mobile networks and distributed clouds.*

In this article, we describe how to achieve the above-mentioned objectives through the Follow-Me Cloud (FMC[1]) concept, described below. An important restriction on which we base our work consists of the fact that we shall introduce neither additional cost nor complexity to the network. The usage of software defined networking (SDN) technologies such as OpenFlow and the like is thus not considered. Changes to Third Generation Partnership Project (3GPP) standards, including those relevant to the nodes and interfaces of the EPS architecture or the underlying protocols, are not an option either.

The remainder of this article is structured as follows. We give an overview on some related research work. The proposed FMC concept is described. We give preliminary performance results. The article is then concluded.

Related Work

Session/Service Identification

Generally speaking, migration of an IP service, due to movement of the receiving UE followed by a change in its IP address, would result in the breakdown of the session and the need to reestablish a new one. This is intuitively due to the fact that IP addresses are, in practice, used for identifying both an endpoint and a network location. Session identifiers should therefore be separated from location identifiers. Methods for such separation have been devised before. Domain Name Service (DNS) does realize such a separation, but it was not designed to provide constant updates of current location. It is rather used only once at session establishment time. The Locator/Identifier Separation Protocol (LISP) [6] makes such separation explicit, but does not natively support endpoint mobility. There are some efforts to include mobility support in LISP, but most of these approaches rely on using a centralized mechanism based on the map server (MS), which makes LISP deployment in architecture like 3GPP networks very complicated. Serval [7] caters for user and service mobility and provides identifier/location separation by introducing an additional layer in the networking stack. It makes use of service identifiers, which require changes to applications using the system. To avoid the breakdown of an IP session between two peers when the IP address of any of the two peers changes during the course of a session, Network Address Translation (NAT) can be also used. In the context of mobile networks, the support of NAT would require changes to nodes of the mobile network operator; also, many operators are not in favor of NAT, mainly due to the foreseen expansion of IPv6. Other research work has considered the usage of OpenFlow to hide, through its rules, any changes to the IP addresses [19]. For OpenFlow-based solutions, scalability represents the main challenge. Indeed, there are various dimensions for scalability, including the number of flows, the number of rules, the flow setup rate, number of packets, and the bandwidth of the control channel. Some ideas have been proposed to deal with this issue. DevoFlow [8] reduces the number of control packets by moving some of the flow creation work from controllers to switches. In [9], the scalability of OpenFlow rules in an FMC scenario is assessed, and an approach to distribute control plane functions is proposed to enhance the system scalability. As mentioned earlier, usage of OpenFlow and other SDN technologies is not an option in this article to avoid any additional complexity in mobile networks.

Information-centric network (ICN) architecture natively supports the separation between the user location and the content identifiers. Indeed, ICN shifts away from a host-centric model toward an information-centric one, where content is retrieved according to its name instead of its storage location (host address). Several ICN approaches have been proposed, such as data-oriented network architecture (DONA) [10] and content-centric networking (CCN) [11]. They share the same concepts: contents belonging to a service have a unique name and are cached at different locations in the network, where:

- The name is independent from the location.
- The communication is driven by a publish/subscribe model.

These solutions differ in the way the contents are named/identified. Naming in ICN could be hierarchical as in CCN, or a flat namespace as in DONA. In CCN the names are rooted in a prefix, unique for each publisher. The granularity of the names is at the chunk level. The content name has several components delimited by a character (e.g., /FMC/Content1/chunk1.extension). Behind using hierarchical addressing is the facility to achieve better routing scalability within name prefix aggregation. In fact, CCN names are used for both naming and transport. Meanwhile, names in DONA are in the form P:L, where P is the hash of the owner's public key and L is the owner assigned label. DONA requires another entity (resolution handler) to perform name resolution by using a route-

[1] *While FMC is widely used to stand for fixed mobile convergence, this abbreviation stands for Follow-Me Cloud throughout this article.*

by-name paradigm. Service centric networking (SCN) [12] extends the principle of ICN to support service request in addition to content request. Services are software elements located in the network infrastructure, hosted on dedicated hardware placed alongside the routing infrastructure (as is done with cloud services). Service naming is a mix between flat and hierarchical naming. The service name is composed of two parts, <service_owner, service_name>, and wild cards are used (<*, service_name>) if users do not know the service provider in advance. It is worth mentioning that ICN naming is very relevant for FMC context, where the aim is to achieve a clear separation between service location/mobility and UE mobility (layers 2 and 3, L2 and L3).

Service Location/Migration in Federated Clouds

Federated clouds refer to the connection of geographically distributed DCs together into a common resource pool to deliver a variety of cloud services. Upon reception of a service request, one of these DCs is chosen to deliver the requested service over the network to the end user. The distribution of cloud computing resources over different locations in the network is beneficial for different reasons such as increasing availability, reducing bandwidth cost, and reducing latency by locating resource near to users. To efficiently handle user requests, there is a need to define a cloud management procedure. This procedure directs the user's service request to the optimal DC, which satisfies user constraints (cost), optimizes network use (load balancing), and ensures application quality of service (QoS)/QoE. Furthermore, this cloud management procedure must be able to migrate all or portions of services between DCs if one of the selected criteria is no longer satisfied (QoS degradation). Obviously, redirecting a user request to the geographically nearest DC seems to be the most efficient solution. However, for successful services (in a certain region), redirecting all requests to the geographically nearest DC can overload it, causing degradation of QoS/QoE. Therefore, more sophisticated solutions need to be used for cloud management.

In [13], a cloud management middleware is proposed to migrate part of user service (constituted by a set of virtual machines, VMs) between DC sites in response to workload change at the DC. Based on workload monitoring at each DC, the middleware initiates VM migration in order to move application components (geographically) closer to the client. Volley [14] is an automatic service placement for geographically distributed DCs based on iterative optimization algorithms. Volley migrates services to new DCs if the capacity of a DC changes or the user changes location (chooses a DC near the new location). The authors of [15] propose a DC selection algorithm for placing the requested VM by a user such that it minimizes the maximum distance between any two DCs. The DC selection problem was formulated as a subgraph selection problem. The demonstrator described in [16] shows how services can be placed according to information retrieved from an application-layer traffic optimization (ALTO) network server. This work can be used to find optimal service locations. Note that most of these research teachings are orthogonal to the FMC framework described herein.

It is worth noting that there are technical issues to consider when migrating services (typically VMs) between two DCs. These issues pertain to the time needed to transfer a VM between DCs, which can disturb the service continuity. This time depends on:

- The time required for converting a VM, particularly if DCs are not using the same hypervisor
- The time required for transferring the service (VM) over the network

The latter intuitively depends on the object size, the connection speed, and the round-trip time (RTT) between the DCs. It is of high importance as VMs are transferred using FTP/TCP-like applications, the performance of which largely depends on RTT. To fix this issue, solutions such as file data transfer (FDT) [17] can be used.

Follow Me Cloud

Problem Statement

Referring again to Fig. 1, a user may be receiving an application/service from a server in DC 1 in location 1 via P-GW1 at a particular time instant. Later on, the user moves to a different location (i.e., location 2), and receives the remaining part of the service from a server in nearby DC 2 via an optimal anchor point, P-GW2. In this regard, two mobility scenarios can be envisioned:

- Connect-freeze-reconnect mobility: In this scenario, the user temporarily pauses/freezes the cloud service when moving from location 1 to location 2 (e.g., a thin client user accessing data from an office, then getting offline while returning home, and then accessing data from home).
- Always connected mobility: In this scenario, the user changes P-GW and DC while being on the move and with no interruption in the service (e.g., a thin client user accessing data while being onboard a high-speed train during a long journey).

The issues we aim to solve in this article are the following:

- In Fig. 1, when the UE moves from location 1 to location 2, both the IP address of the UE and the IP address of the server may change. As discussed earlier, with current networking solutions, an IP session between two peers will simply be torn down if the IP address of any of the two peers changes during the course of the session.
- The second issue pertains to the fact that for the sake of system scalability, the system does not need to migrate the whole service to the new location of the user, only the required portion of service.
- The third issue pertains to when, how, and to which DC the service migration shall be triggered, as well as how the UE shall become aware of the availability of optimal DCs and/or data anchor gateways.

This article proposes a number of solutions that address all these issues, defining a general framework that interworks between a distributed mobile operator network and a network of regional DCs, a federated cloud, to enable the vision of FMC whereby a cloud service follows the user along her movement. As described herein, the key features of the proposed solutions are the following:

- Replacing data anchoring at the network layer by service anchoring
- Replacing IP addressing by service/data identification
- Decoupling session/service[2] mobility from layers 2 ad 3 mobility

Network Architecture

In this article, we consider a network topology as shown in Fig. 1, with additional components, namely FMC controller and DC/GW (data center/gateway) mapping entity as illustrated in Fig. 2. It shall be noted that, while in Fig. 2, these nodes are shown as two independent architecture components, they can be functional entities collocated with existing nodes or run

[2] *Throughout this article, the terms service and session are used interchangeably to refer to the same thing: a service being delivered over a session.*

as software on any DC of the underlying cloud. In both figures, both the cloud network and the mobile operator network are decentralized/distributed.

We first propose that a mobile network operator maps access point names (APNs) to specific geographical locations (or alternatively to P-GWs' identifiers). These geographical locations could be S-GW service areas, mobility management entities (MMEs) pool areas, P-GWs' geographical locations, and so on. The corresponding localized APNs would look like APN1="Internet@location_1," APN2="Internet@location_2," and so on. Admittedly, this is somehow against the original principle of the APN, that is, to achieve location independence of access to a PDN. Indeed, the concept of APNs was designed for General Packet Radio Service (GPRS) (and carried over to Universal Mobile Telecommunications Systems — UMTS and EPS) as a scheme to separate logical from physical points of interconnection between a 3GPP operator's IP network and externally connected PDNs.

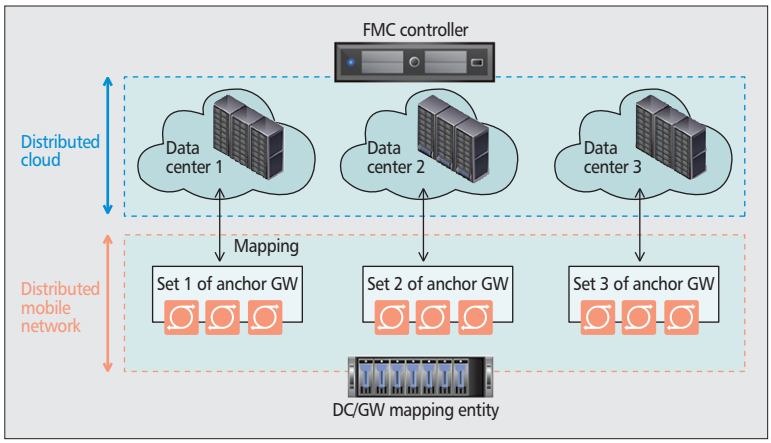

Figure 2. *Interworked cloud/mobile networks architecture.*

The APN allows one logical name to be associated with a particular type of traffic and maps it flexibly (but constant for the duration of an IP/PDN connection) to a route and point of interconnection. The mapping is done by the network based on DNS, and while the UE may be aware of it, it is not concerned with details of the backend connectivity. This was suitable for typical highly centralized network deployments; however, with new traffic and load scenarios coming into play (especially traffic offload at decentralized points as mentioned earlier), this is no longer sufficient. The UE, not necessarily the user, may become involved (partially) with network topology for the sake of its optimal backend connectivity (i.e., minimal network resource consumption, cost, and latency) even with active data transmission over relatively long durations and with larger scale mobility.

In the envisioned network architecture, we also consider that DCs are mapped to a set of P-GWs (i.e., data anchor points in EPS) based on some metric, e.g., location or hop count. This mapping may be static or dynamic. In case of the latter, it could be that the topology information is being exchanged between an FMC service provider and a mobile network operator (MNO). Alternatively, an MNO entity/function could be in charge of updating the FMC service provider with such information in either a reactive or proactive manner. Additionally, we assume that an FMC controller entity exists for managing distributed DC instances; alternatively, distributed DCs coordinate among themselves in a self-organizing network (SON) manner.

It shall be noted that the cloud infrastructure and the mobile network could belong to the same operator (i.e., MNO = FMC service operator) or be administered by two independent operators. The FMC controller and DC/GW mapping entity could be either in the premises of the MNO and/or FMC service provider, or owned and operated by a third party.

In the envisioned FMC service, similar in spirit to CCN, content served by the FMC service has some predefined hierarchy; for example, **content ID = FMCService/Application-Name.DataName.Characteristics**. In the case of the movie *Titanic*, it could be that the content ID = Video.Titanic.30min; this means that this content is video content from *Titanic*, and the frames to be played back are those from the 30th minute since the beginning of the movie.

In this article, we mainly focus on the case of UE devices in EPS connection management (ECM)-active mode. The focus on UE devices in ECM-active mode is important due to the fact that most Long Term Evolution (LTE) UE devices, such as tablets and PCs equipped with an LTE modem, and even devices similar to currently available 3G smart phones will have ongoing background traffic due to many applications (Skype, Foursquare, etc.) that involve the frequent signaling of updates and keep alive messages, ultimately keeping UE devices always actively connected to the network.

FMC Session/Service Identification

To replace IP addressing by service/data identification, a specific application logic/plugin is installed at the UE and the DC servers. Indeed, requests from UE devices for an application or a service available in the cloud are mapped to a unique session/service identifier. In other words, any IP session between a UE device and a cloud server is identified as follows:

Session/Service ID = ***Function***(UE_ID ; Content_ID)

This session/service ID is generated by the end host (e.g., UE) that issues the service request and is communicated to the receiving end host, which is the cloud server.

It shall be noted that the above proposed structure of the session/service identification ensures that all sessions used by the same UE or all sessions used by all UE devices belonging to any mobile network will be uniquely identified, and there shall be no conflict in the session/service ID. Indeed, the usage of the UE ID (which is supposed to be unique within and across different mobile operator networks) in the session/service ID serves to avoid any conflict in session/service ID among UE devices, whereas the usage of content ID in the session/service ID helps to differentiate sessions received by the same UE device. As explained later, the latter also facilitates a smooth migration of the session/service from a DC to another one, achieving the concept of FMC. It is also important to note that since UE devices already have unique IDs, there is no need for a particular server to set up a session ID (e.g., as in the case of the Session Initiation Protocol, SIP). Indeed, in the context of EPS, a number of UE identifiers can be used. The mobile subscriber integrated services digital network number (MSISDN, i.e., the phone number attached to the subscriber identity module, SIM, card), international mobile subscriber identity (IMSI), international mobile phone equipment identifier (IMEI), temporary mobile subscriber identity (TMSI), and integrated circuit card ID (ICCID) are all potential alternatives. While it is outside the scope of this article to decide which identifier to use, the following compares between MSISDN, IMSI, TMSI, ICCID, and IMEI. First of all, the main concern with MSISDN is the fact that an only-packet-switched (PS) UE device does not need to have an MSISDN. As for IMSI, it is highly confidential, and mobile

operators prefer not to expose it outside the mobile network domain. Regarding TMSI, as the name infers, it is a temporary identifier that may change during the course of a session/service, mainly in case the UE goes idle for a while. It is thus not preferred for supporting "connect-freeze-reconnect" mobility scenarios. Using ICCID may be an interesting solution as an ICCID with the minimum sized individual account identification number (IIN) (11 digits) provides approximately 10^{11} or 100 billion unique identifiers per IIN. This amount per issuer (e.g., per MNO) would appear to be more than adequate to provide a new unique subscription identifier for FMC service/sessions from different UE types, including, say, machine-to-nachine/machine type communications (M2M/MTC) devices. It should also be noted that similar to MSISDN, the composition of ICCID contains enough routing information to be used to identify the home subscriber server/home location register (HSS/HLR) of the UE/mobile station (MS) in case the FMC controller needs to contact the HSS/HLR. The IMEI and IMEISV (IMEI software version) are used to identify individual mobile devices. The total number of devices that can be uniquely identified with an IMEI is 10^{14}, which seems to also be adequate for supporting FMC service requests from different UE/MS types.

Triggering FMC Session/Service Migration

The possible need for FMC service migration can be intuitively noticed when a UE device changes its data anchor gateway (i.e., P-GW relocation); that is, changes its IP address. A change of the IP address of the UE device can certainly be noticed by the corresponding DC. A preliminary decision has to first be made by the UE and/or current DC on whether a service migration is worthwhile or not. This decision may be based on the service type (e.g., an ongoing video service with strict QoS requirements may be migrated) [12], content size (e.g., when a user has been watching a movie and the movie is about to finish at the time of P-GW relocation, the UE may decide, at the FMC application layer, not to initiate the service migration), task type of the service (e.g., in case of MTC, in a session of emergency warning services, delay-sensitive measurement reporting services always have to be migrated to the nearest DC), and/or user class. It is worth noting that the service migration decision (to migrate or not) relies on several attributes/criteria (could be conflicting) that depend on the user's expectation on the service (QoS/QoE, cost) and network/cloud provider policies (at each P-GW relocation, load balancing, maximize use of DC resources). Accordingly, to migrate a service or not can be defined as a multi-attribute decision making (MADM) issue, and solved by any relevant algorithm in this area.

Once it is deemed appropriate, by either UE or current DC, to migrate the service, the FMC plugin available at the DC may request the FMC controller to select the optimal DC with the right service and right content to serve the UE in its new location, and to initiate the service migration. As a service may consist of multiple cooperating sessions and pieces, the decision has to be made whether the service has to be fully or partially migrated, while considering the service migration cost, such as the cost associated with the initiation of a new virtual machine at the target DC, the cost (if any) associated with the release of resources at the source DC, and the cost associated with bandwidth consumption due to traffic to be exchanged between the DCs as well as the FMC controller. An estimate of the cost/overhead possibly incurred shall be compared against benefits to the cloud in terms of traffic distribution and to end users in terms of QoE. It shall be noted that there are different forms (e.g., state, data, images), different technologies (e.g., VMware), and different approaches (e.g., SaaS, PaaS, or IaaS) for service migration. The latter decides the former.

Awareness of the Need for Data Anchor Gateway Relocation

As mentioned earlier, service migration may be triggered following data anchor gateway relocation. Such relocation is feasible for UE devices in ECM-idle mode [1] and also for UE devices in ECM-active mode [2]. These solutions work under the assumption that a UE device is aware when an optimal P-GW becomes available and subsequently establishes a new IP session via this optimal P-GW. An important question is how a UE device becomes aware of the availability of an optimal data anchor gateway, so it will trigger relocation from the current data anchor gateway to the optimal one. In this subsection, we provide a number of solutions that render a UE device aware of these things. Indeed, a UE device may use the S-GW change or MME change within existing handover procedures as a trigger. It should be noted that S-GW change and MME change could potentially indicate a change in the S-GW service area and MME pool area, respectively. In the case of an S-GW change for the cause of load balancing, this change shall indicate that the current S-GW is no longer optimal, and that another better S-GW has become available. Additionally, and especially in a distributed mobile operator network where S-GWs could be potentially collocated with P-GWs, a change in S-GW could be an indication that a change of P-GW may be desired; even with non-collocated S-GW and P-GW, the same indication of non-optimality of the current P-GW can be utilized. It should be noted that according to current 3GPP specifications [5], a UE device is aware of an MME change, but not of an S-GW change. Knowing of an MME change does not necessarily make a UE device aware of the distributed network topology; the same can be said when the UE becomes aware of an S-GW change. Indeed, a UE device needs to know only about the optimality of the currently serving P-GW, not the distributed network topology in full.

As mentioned earlier, while MME change is noticed by the UE, as it holds relevant context at its information storage, with the current standards, an S-GW change cannot be noticed by the UE. For this purpose, we propose that when an S-GW changes as part of a tracking area update (TAU) procedure (which in turn occurs within an X2or S1-based handover procedure [5]), MME sends a corresponding flag in the TAU accept message to the UE. The UE shall interpret this flag as an indication that the current P-GW may no longer be optimal and that another optimal P-GW may have become available. Alternatively, the MME sends the optimal APN in the TAU accept message to the UE so that the UE will use it to request PDN connectivity whenever it desires to initiate a new IP session to the same PDN.

Alternatively, a UE device may request APN information from a configuration server (e.g., access network discovery selection function, ANDSF) and subsequently requests PDN connectivity indicating the "localized" APN. For the sake of comparison, Fig. 3 depicts the existing APN resolution mechanism (full lines and steps numbered from 1 to 4) and the proposed mechanism (dashed lines and steps numbered from A to D). This option assumes that the ANDSF acquires localized APN information. The advantage is that existing non-access stratum (NAS) signaling can be kept unchanged (only the ANDSF information element is used differently). It should be noted that while ANDSF was initially designed to prioritize for a UE device a list of currently available non-3GPP accesses, there is recent work in 3GPP that aims to enable ANDSF to provide UE devices with policies on which

PDN connection to select [7]. Based on an indication from the MME that an S-GW has changed or an MME change notification, UE consults ANDSF or DNS or any other network node with defined policies. ANDSF (or the like) is assumed to maintain a table, mapping APNs for each location. Upon receiving the current location of the UE from the UE, ANDSF provides the UE with policies, based on which UE establishes new IP sessions to the same PDN via a new optimal P-GW using the relevant APN indicated by the ANDSF. Using this indicated APN, the UE issues a PDN connectivity request to the MME [3]. MME uses the P-GW selection function to select the optimal P-GW for the UE to connect to the same PDN [16]. After the setup of the new PDN connection, the UE stores the relevant APN into its information storage and maps the relevant IP flows to the relevant PDN connection and APN. The added signaling steps between UE and ANDSF and the different use of APN (now as a "localized" APN) is shown in Fig. 3, with dashed lines and steps numbered from A to D. The last two steps are identical to steps 1 and 2 of the existing procedure. It is also assumed that ANDSF has its configuration data aligned with the DNS data; this is indicated by the double arrow between the two entities. When IP sessions being delivered on top of a given PDN connection are all off (e.g., if the time of the last received/transmitted packet on the PDN connection is older than a certain threshold), the relevant APNs are deleted from the UE's information storage.

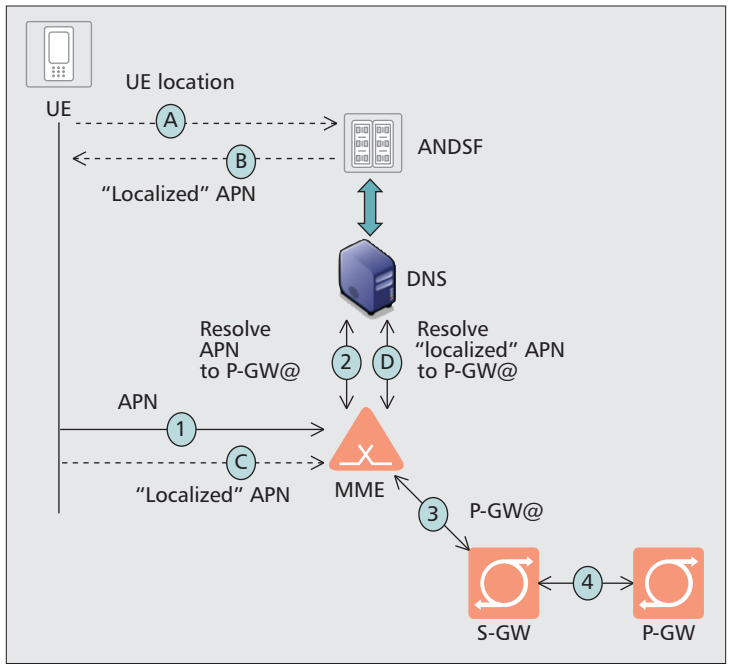

Figure 3. *Overview of APN resolution mechanism (full lines: existing 3GPP mechanism; dashed lines: proposed enhancement).*

FMC Session Establishment and Migration

Figure 4 shows the flow of signaling messages and procedures carried out to establish an FMC session and migrate it to a different server via a different anchor point. In step 1, the network layer of the UE establishes PDN connectivity with an adequate P-GW, PGW1. The UE is then assigned an IP address, IP1, from within the range of IP addresses of PGW1. Later on, at step 2, the user of the UE decides to initiate a session/service to view content available at the cloud. For example, the user indicates the data she desires to view to the FMC controller (or to another appropriate node in the cloud domain) via a web portal or web interface. Based on the current IP address of the UE and DC/GW mapping information available at the FMC controller (or another appropriate node in the cloud domain), the FMC controller selects the appropriate DC and issues a request for establishing the relevant session in step 3. In step 4, the FMC controller indicates the content ID (i.e., the content name and relevant features), the UE identifier to identify the session, and the IP address of the UE, IP1. Afterward, the session is established and identified as a function of the content ID and the UE Identifier (step 5). In step 6, during the mobility of the user, the UE becomes aware of the availability of an optimal anchor gateway as described above. In step 7, the UE establishes a new PDN connection and receives a new IP address, IP2. Being aware of the change in the IP address, and once it deems that a service migration is worthwhile, the FMC application logic at the UE issues a service migration request to the FMC controller (or to another appropriate node in the cloud domain), indicating the session/service ID with new characteristics regarding the content/service (last played frame of a video content, last viewed page of an electronic book, etc). In step 9, once the FMC controller decides that it is worthwhile to enforce the migration of the service to a different DC (i.e., comparing incurred overhead/cost vs. benefit), it carries out DC selection based on the DC/GW mapping information and the new IP address of the UE, IP2. In step 10a, if the content (e.g., code, data, state) is not available at the newly selected DC, the FMC controller issues a content migration request message to the source DC requesting that it forward required content portion to the newly selected DC. In response, in step (10a′), the source DC forwards required content and/or exchanges adequate state information with the newly selected DC. It should be noted that depending on the data size, data migration can be performed using one or more suitable robust and fast data delivery technologies. In step 10b, the FMC controller issues a session migration request message to the selected target DC (DC2 in Fig. 2) indicating the session/service ID, new characteristics of the content/service, and the new IP address, IP2. In step 11a, the session/service migration takes place. In this way, despite a change in the IP addresses of both the UE and DC server, the session continues without being torn down as the session/service is identified by a unique identifier of the mobile terminal. In step 11b, if the old PDN connection to the old PGW (PGW1) was solely used by the FMC session/service, it is released based on a trigger from the FMC application logic at the UE.

Regarding steps 8–10b, it may be that the UE issues a session/service migration request to the source DC server. Assuming the DC server is acquired with the DC/GW mapping information (alternatively, the DC server may consult the DC/GW mapping node on demand), the source DC server selects the target DC based on the new IP address of the UE. It then forwards the required portion of the content to the newly selected DC and requests session migration indicating the new IP address of the UE and the session ID. Then steps 11a and 11b take place. While Fig. 4 shows the case of UE-triggered session migration, session migration can also be triggered by the cloud (e.g., for maintenance of the current DC). Intuitively, in the case of cloud-triggered session migration, steps 6, 7, and 8 are omitted. Instead, the DC requests session migration, as in step 8.

Results

In this section, we present preliminary results regarding the performance of FMC. Further results based on an analytical model of FMC are available in [18]. We used ns-3 to simulate

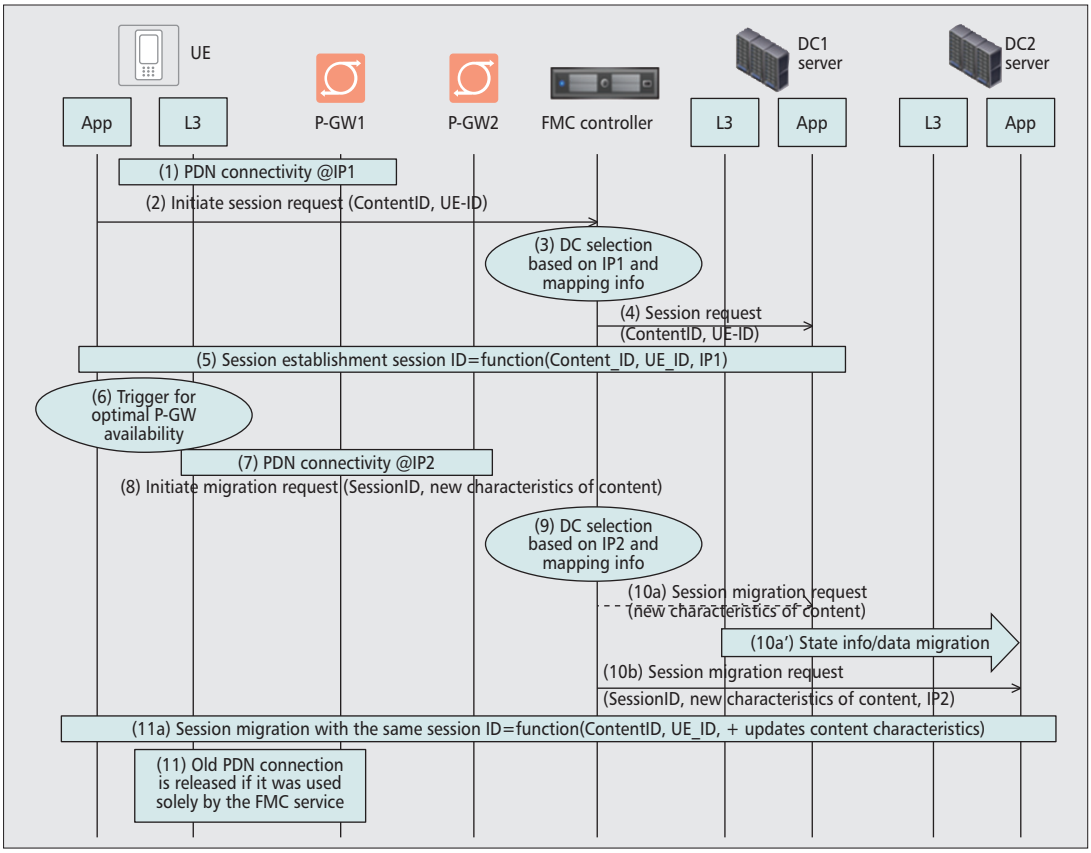

Figure 4. *Flow chart for initial FMC session establishment and FMC session migration.*

the architecture of Fig. 1, adding one more location (location 4 including one more DC, S-GW and P-GW). We used a mobile UE, which remains in each P-GW service area for a duration of 5 min. The simulation runs for 20 min. We compare FMC against the case of triggering service migration after two P-GW relocations and the case of no service migration (the service remains in the first affected DC). Here, we consider no congestion in the used links. The data latency depends only on the number of hops (communication path length) from P-GW to DC. We assume that the time required for the service migration is short enough to ensure no distribution in the service.

Figure 5 shows the data latency during the simulation for the three mechanisms. Clearly, we notice that FMC achieves the lowest data latency as the service is always placed at the optimal DC (geographically nearest). In contrast, if no service migration is used, the data latency increases along with the UE movement, as the UE is connected to new P-GWs that have long communication paths to the initial DC hosting the service. However, the gain of FMC has a cost in terms of signaling and number of objects migrated, which is higher than the other two mechanisms (Table 1). Effectively, for each service migration, the cost is incurred by the size of the migrated objects and the number of exchanged signaling messages (typically three messages; Fig 3). In FMC, service migration is triggered after each P-GW relocation; the final cost in this simulation scenario is therefore three times (i.e., 3 P-GW relocations) the cost of service migration.

These results clearly indicate a need for more sophisticated algorithms for service migration. Therefore, solutions such as those based on MADM algorithms can efficiently balance between performance and incurred cost. Indeed, as stated in the article, the decision to migrate a service or not is not trivial as there are several constraints to consider, which relate to either operator policies or user quality needs. Therefore, any underlying decision making process needs to find a trade-off between these attributes. MADM techniques are usually used to solve such problems. One of the most efficient MADM approaches is based on the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) solution. TOPSIS assumes the availability of m alternatives (options) and n attributes/criteria as well as a score for each option with respect to each criterion. We denote by $x_{i,j}$ the score (attribute) of option i with respect to criterion j. In our case, m represents the decision of migrating a service or not, while n represents the number of criteria (e.g., QoE, P-GW relocation, cost of migration) to be considered in the decision mak-

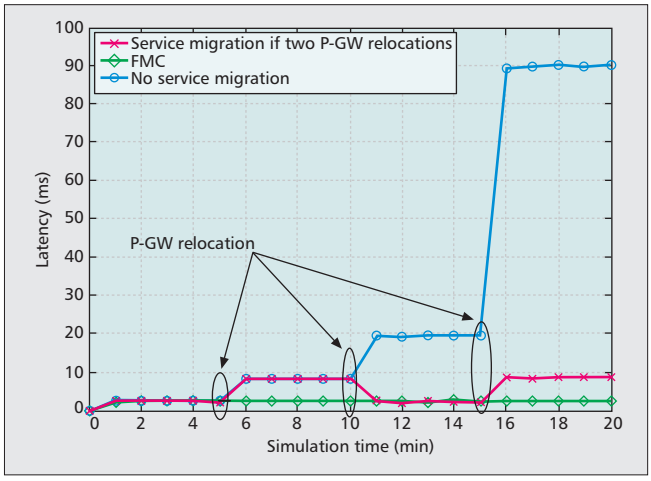

Figure 5. *Data latency for a UE.*

Mechanism	Cost
FMC	3*(Migrated-Objects-size + 3* Signaling messages)
Service migration (if two P-GW relocation)	1* (Migrated-Objects-size + 3* Signaling messages)
No service migration	0

Table 1. *Cost incurred by service migration.*

ing process. According to the TOPSIS technique, the decision to migrate a service or not will depend on the alternative that reduces the gap with the ideal solution according to criteria as well as the weight defined before. Employing MADM in FMC defines one of the authors' future research work directions.

Conclusion

The described FMC framework enables mobile cloud services to follow their respective mobile users during their journeys by migrating all or portions of services to the optimal DC to ensure them the best QoE. A service migration decision is based on user constraints and network operator policies, and particularly on the P-GW relocation procedure. In fact, at each P-GW relocation, the service migration procedure has to decide to migrate or not, or to migrate a portion (or none) of services to a new DC that is near the new P-GW location in terms of communication path length. First results show the potential of FMC to reduce the data latency when accessing a service in the cloud for mobile users.

Furthermore, FMC implementation is possible without the use of any SDN technology, avoiding any otherwise associated scalability issues, only exploiting the already available unique identifiers of mobile users and findings of ICN and CCN, particularly those relevant to service/content naming, a topic increasingly gaining tremendous interest. The framework does not add any major complexity to the current mobile network architecture, and is thus highly feasible, practical, and standards-compliant.

While the present article validates the FMC concept through simulations, some of the authors' recent research work has proven its feasibility using real tests, particularly an OpenFlow-based implementation of FMC. The findings of this implementation are available in [19].

References

[1] K. Samdanis, T. Taleb, and S. Schmid, "Traffic Offload Enhancements for eUTRAN," *IEEE Commun. Surveys & Tutorials*, vol. 11, no. 3, Aug. 2012, pp. 884–96.

[2] T. Taleb, K. Samdanis, and F. Filali, "Towards Supporting Highly Mobile Nodes in Decentralized Mobile Operator Networks," *Proc. IEEE ICC 2012*, Ottawa, Canada, June 2012.

[3] R. Miller, "AOL Gets Small with Outdoor Micro Data Centers," *Data Center Knowledge*, July 2012.

[4] R. Miller, "Solar-Powered Micro Data Center at Rutgers," *Data Center Knowledge*, May 2012.

[5] 3rd Generation Partnership Project, "General Packet Radio Service (GPRS) Enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Access," TS 23.401 (work in progress).

[6] D. Farinacci *et al.*, "Locator/ID Separation Protocol (LISP)," IETF Internet draft draft-ietf-lisp-13.txt, June 2011.

[7] E. Nordström *et al.*, "Serval: An End-Host Stack for Service-Centric Networking," *Proc. 9th USENIX Symp. Networked Sys. Design and Implementation*, San Jose, CA.

[8] A. R. Curtis *et al.*, "DevoFlow: Scaling Flow Management for High-Performance Networks," *Proc. ACM SIGCOMM 2011*, Toronto, Canada, Aug. 2011.

[9] R. Bifulco *et al.*, "Scalability of a Mobile Cloud Management System," *Proc. Wksp. Mobile Cloud Computing (MCC) in conjunction with ACM SIGCOMM 2012*, Helsinki, Finland, Apr. 2012.

[10] T. Koponen *et al.*, "A Data-Oriented (and Beyond) Network Architecture," *Proc. ACM SIGCOMM 2007*, Kyoto, Japan, Aug. 27–31, 2007.

[11] V. Jacobson *et al.*, "Networking Named Content," *Proc. ACM CoNEXT 2009*, Roma, Italy.

[12] T. Braun, A. Mauthe, and V. Siris, "Service-Centric Networking Extensions," *Proc. ACM Symp. Applied Computing 2013*, Coimbra, Portugal.

[13] B. Malet and P. Pietzuch, "Resource Allocation Across Multiple Cloud Data Centres," *Proc. ACM MGC 2010*, Bengalore, India.

[14] S. Agarwal *et al.*, "Volley: Automated Data Placement for Geo-Distributed Cloud Services," *Proc. 7th Symp. Networked Syst. Design and Implementation 2010*, San Jose, CA.

[15] M. Alicherry and T. V. Lakshman, "Network Aware Resource Allocation in Distributed Clouds," *Proc. IEEE INFOCOM 2012*, Orlando, FL.

[16] M. Steiner *et al.*, "Network-Aware Service Placement in a Distributed Cloud Environment," *Proc. ACM SIGCOMM 2012*, Helsinki, Finland, Aug. 2012.

[17] File Data Transfer, http://monalisa.cern.ch/FDT/.

[18] T. Taleb and A. Ksentini, "An Analytical Model for Follow Me Cloud," *Proc. IEEE GLOBECOM 2013*, Atlanta, GA, Dec. 2013.

[19] T. Taleb, P. Hasselmeyer, and F. Mir, "Follow-Me Cloud: An OpenFlow-Based Implementation," *Proc. IEEE GreenCom 2013*, Beijing, China, Aug. 2013.

Biographies

TARIK TALEB (talebtarik@gmail.com) is currently working as a senior researcher and 3GPP standards expert at NEC Europe Ltd, Heidelberg, Germany. Prior to his current position and until March 2009, he worked as an assistant professor at the Graduate School of Information Sciences, Tohoku University, Japan, in a laboratory fully funded by KDDI, the second largest network operator in Japan. From October 2005 to March 2006, he worked as a research fellow with the Intelligent Cosmos Research Institute, Sendai, Japan. He received his B.E. degree in information engineering with distinction, and M.Sc. and Ph.D. degrees in information sciences from GSIS, Tohoku University, in 2001, 2003, and 2005, respectively. His research interests lie in the field of architectural enhancements to mobile core networks (particularly 3GPP's), mobile cloud networking, mobile multimedia streaming, congestion control protocols, handoff and mobility management, intervehicular communications, and social media networking. He has been also directly engaged in the development and standardization of the Evolved Packet System as a member of 3GPP's System Architecture working group. He is a board member of the IEEE Communications Society Standardization Program Development Board. As an attempt to bridge the gap between academia and industry, he founded and has been the General Chair of the IEEE Workshop on Telecommunications Standards: From Research to Standards, a successful event that received the Best Workshop Award by IEEE the Communication Society. He is/was on the Editorial Boards of *IEEE Wireless Communications*, *IEEE Transactions on Vehicular Technology*, *IEEE Communications Surveys & Tutorials*, and a number of Wiley journals. He is serving as Vice-Chair of the Wireless Communications Technical Committee, the largest in IEEE ComSoc. He also served as Secretary and then Vice Chair of the Satellite and Space Communications Technical Committee of IEEE ComSoc (2006–2010). He has been on the Technical Program Committee of different IEEE conferences, including GLOBECOM, ICC, and WCNC, and has chaired some of their symposia.

ADLEN KSENTINI (adlen.ksentini@irisa.fr) is an associate professor at the University of Rennes 1, France. He is a member of the INRIA Rennes team Dionysos. He received an M.Sc. in telecommunication and multimedia networking from the University of Versailles. He obtained his Ph.D. degree in computer science from the University of Cergy-Pontoise in 2005, with a dissertation on QoS provisioning in IEEE 802.11-based networks. His other interests include future Internet networks, cellular networks, green networks, QoS, QoE, and multimedia transmission. He is involved in several national and European projects on QoS and QoE support in future Internet networks. He is a co-author of over 40 technical journal and international conference papers. He has been in the technical program commitee of major IEEE ComSoc conferences, including ICC/GLOBECOM, WCNC, and PIMRC.