CASE STUDY – PART I

Dynamic Virtualization and Migration of Contents and Servers

Abstract

This case study focuses on the virtualization as well as the dynamic instantiation and migration of resources in a network close to the users. The considered resources encompass services (e.g. thin client applications) and content (e.g. videos stored in a cache), but also the functions of the mobile network itself (e.g. PGW, MME, etc.). The motivation is to provide/resolve the most frequently used resources/content/functions nearer to the mobile user. This will release the resource consumption (e.g., bandwidth, processing, storage) from the core and distribute them autonomously, intelligently and dynamically (on a use-case basis) towards the edge of the network (i.e. across backhaul and access network). The case study objective will be achieved by running the mobile core network functions on top of a virtualization layer on a general purpose server class machine, rather than on dedicated mission specific nodes. This approach is expected to improve the overall QoS/QoE for the users while also improving the resource utilization for the network operator.

Date of release

17/09/2012

SEVENTH FRAMEWORK PROGRAMME

FUTURE NETWORKS

CONTENT

STORY LINE

Due to the fast pace of technological evolution in the field of wireless access technologies and user applications, the existing telecom networks are facing increasing pressure to meet the QoS/QoE demands of mobile users regardless of the type of UE and the underlying access network.

The use of time sensitive and bandwidth intensive applications, e.g. mobile video traffic (streaming and broadcast), is becoming increasingly pervasive and the QoS/QoE demands for their ubiquitous provisioning to mobile users is putting a lot of pressure on mobile network operators and their respective infrastructures (especially the core and backhaul). To effectively meet the customer demands requires increasing network complexity resulting in increased CAPEX/OPEX.

At present, most of the network/content/application service hosting and management is being concentrated at the core. As a result all the user traffic has to go through the core over the backbone/backhaul and hence a lot of resources (bandwidth and processing wise) are consumed. Figure 1 shows the existing operator's network infrastructure. As is evident, the operators maintain a central data centre consisting of extensive storage and processing entities (e.g., NMS, Thin Client Servers, Content, etc.).

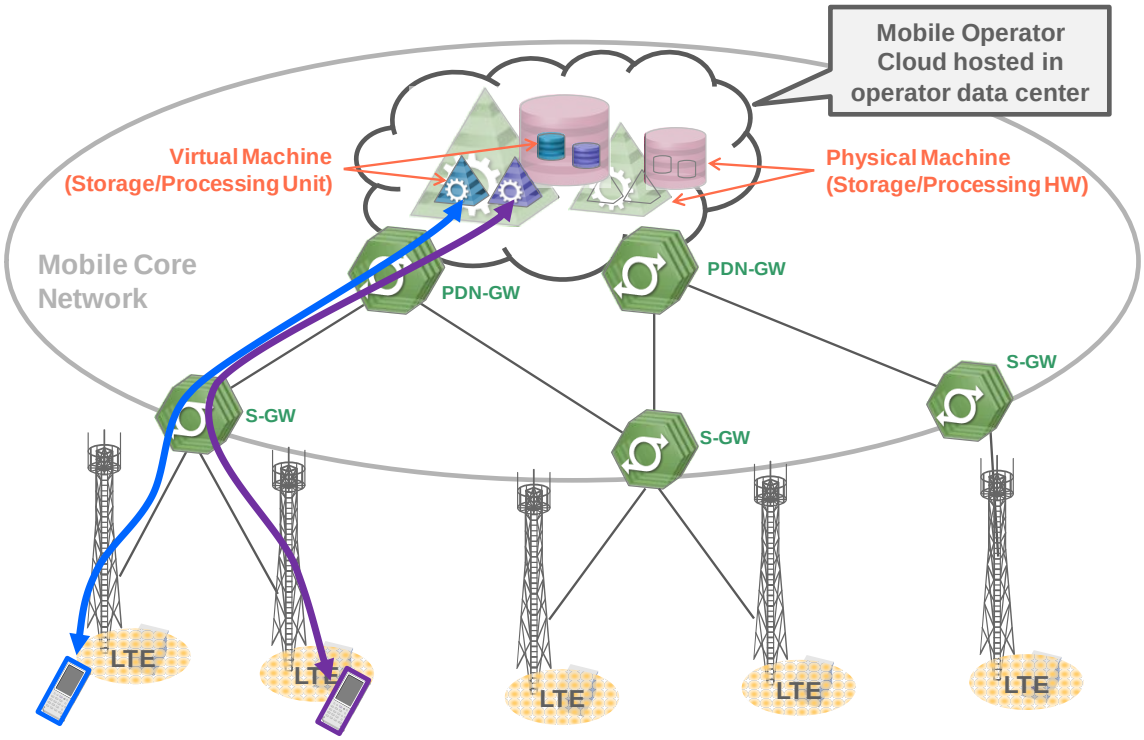

Figure 1 - Existing Mobile Operator Network Infrastructure

This centralized organization of networks imposes serious operational and performance demerits, especially in case of mobile users using bandwidth intensive real time applications (such as streaming video content etc.). Due to mobility, such a centralized system poses several issues due to the constant re-routing of user data to its new point of attachment and this would imply dynamic management of bi-directional tunnels maintained between the UE and the Core (more specifically the PDN-GW). Such a scenario thus imposes extra burden on the backbone and the access links and can significantly impact QoS delivery to mobile users.

Besides, in order to manage high density traffic, dedicated high-end management servers have to be deployed, administered and maintained, resulting in increased CAPEX/OPEX. Also, ever increasing user traffic and introduction of new real-time application and services (VoIP, video conferencing, live webcast, online gaming etc.) add further burden on bandwidth and OPEX. Most importantly, the concentration of network/service/content management introduces a single point of failure, and can potentially increase end-to-end delay for mobile users using real-time applications.

For efficient dispensation of services (cost effective, seamless/ubiquitous and with maintained QoS), there is a need to re-evaluate the existing operator's network infrastructure architectures and propose intelligent solutions that will provide scalable, survivable and autonomic (self-x) solutions with reduced CAPEX and OPEX.

Objectives

The main aim of this case is the infrastructural/functional reorganization of the current mobile networks in order to improve the data/service/application delivery/execution process for mobile users. In this case study, this is achieved by running the mobile core network functions (e.g., billing, charging, traffic shaping, policy marking, application detection, congestion management etc.) on top of a virtualization layer on general-purpose hardware, rather than running it on dedicated hardware. Instead of core network functions, services or content being bound to a fixed physical location (such as a gateway hardware, a thin client server or a content cache), they can be flexibly instantiated, scaled and migrated at runtime to the most appropriate place.

In line with the overall objective of UniverSelf, it is proposed to revisit how networks are traditionally designed, configured and managed. This would require studying the mobile network from the following main aspects

- Architectural aspects
- Mobility aspects
- Autonomic aspects (self healing/organizing)
- Context aspects

Details for each of the aspects are provided in the following section.

Topology

This case study targets the mobile networks with an underlying heterogeneous wireless access environment (GPRS, UMTS/HSPA, LTE, WiMAX, Wi-Fi etc.). The target network topology over which this UC will be applied is illustrated in Figure 1, and will focus on the UE (mobile user), access network, backhaul network, and core network. It should be noted that related and relevant requirements of fixed networks will also be considered in this UC.

PROBLEM STATEMENT

We identify the following 4 core problem areas in order to address the performance issues arising due to the centralized architecture of existing mobile networks.

Problem: Move from a centralized architecture to a more distributed one

From the *architectural aspects*, there is a need to move from a centralized architecture to a more distributed one. This would imply decentralizing the operator core and migrating (or cloning) the necessary functional blocks towards the access network. However, in a decentralized core scenario, in relation with mobile users, the user context/state will move along with the mobile user, i.e. it has to be transferred more frequently. Also, mobile core network nodes are currently dedicated hardware. To be able to dynamically migrate mobile core functions, they must run on more general-purpose nodes on top of a virtualization layer.

Problem: Provide ubiquitous communication services with guaranteed QoS while the UE is moving in a heterogeneous wireless access network environment

One of the main performance issues is during upward and downward *mobility*, i.e., when there is a significant variation in the network resources (e.g., bandwidth and radio resources) and access methods (contention based or connection-oriented). In such a situation, there is a need to dynamically ascertain the available capacity (both in terms of radio resources as well as storage and processing) of the next access network. In case of restrictive access environment, the required resources should dynamically scale accordingly in order not to oversubscribe the access link and nodes while ensuring adherence to minimum acceptable QoS levels.

Problem: Ensure maximum and efficient management and control of network resources and delivery of user services with minimum operator intervention

This implies developing *autonomic* algorithms, both at the system level as well as at the service level. The main feature of the autonomic algorithms must be to enable self organization of the migration and scaling of network resources, while at the same time protecting against any link/service/network level failures and being able to repair itself (self healing) in the case of any such eventuality.

Problem: Acquire and assess relevant context information to support above algorithms

A mobile network has a large set of measurement points and thus produces a huge load of context information about the users, their sessions, the status of the RAN, backhaul and core network, and so on. One problem is to find out which information is really relevant for supporting dynamic instantiation and migration functions and how the raw context data needs to be processed in order to give meaningful indications.

MODELLING

The functional and operational objectives of the case are spread over the three network phases of Deployment, Operation and Optimization. As seen from Figure 2, for this case, an initial traffic/load analysis and technical investigation into the operator's network architecture will be carried out. This is used to determine resource requirements, initial placement of virtual core nodes and caches. Based on that, an initial deployment can take place. During operations, self-x algorithms are responsible for continuously determining new optimized scaling and location of core nodes for efficient delivery of user content. The blackbox representation of the case is shown in Figure 3, whereas Table 1 provides the essential inputs/outputs for the three respective network phases.

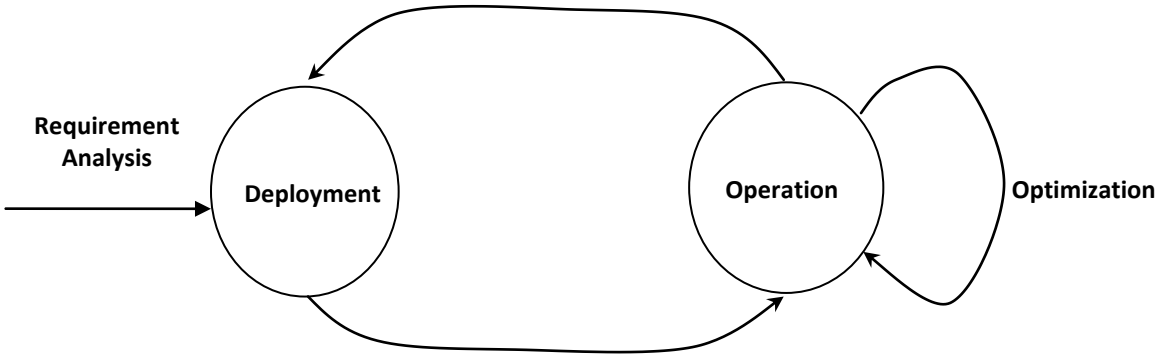

Figure 2: Network Phases for the implementation of the case

Before the deployment phase, the following tasks will be accomplished:

- Traffic/load/resource analysis
- Architectural analysis of the operator's network infrastructure
- Functional and service specifications of network entities (PDN-GW, Caches and Storage, Access routers etc.)

As part of the deployment phase and before the operations phase, the following tasks will be accomplished

- Recommend entities, services and functions for migration
- Specify events/operations as targets for the development of Self-X algorithms
- Specify parameters/metrics for each event/operation that will serve as input to Self-X algorithms that will be developed
- Specify outputs
- Design experiments and specify performance benchmarks

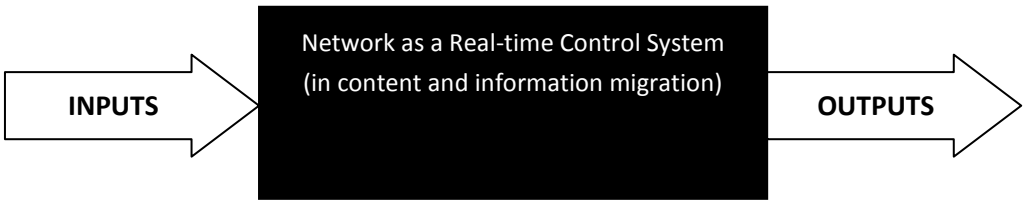

Figure 3: Network Phases for the implementation of the case

Deployment	***Operation***	***Optimization***
INPUTS : • Service and application classification information • Network configuration • Network/Service/Users profile • HW and SW resources • Mathematical models for traffic/load analysis and estimation • Measurements (if applicable) OUTPUTS : • Placement of (virtual) core nodes and gateways (e.g., P/S-GW, MMEs, PCRF etc.) • Cache placement	INPUTS : • Placement of (virtual) core nodes and gateways (e.g., P/S-GW, MMEs, PCRF etc.) • Cache placement • Mobility events • QoS degradation • Cache overload/failure • GW overload/failure • Mobile backhaul link congestion/failure • Self-X algorithms for autonomic management of GW/Caches/links etc OUTPUTS : • Migration of (virtual) core nodes and gateways (e.g., P/S-GW, MMEs, PCRF etc.) • Cache migration • GW scaling	INPUTS : • Placement of (virtual) core nodes and gateways (e.g., P/S-GW, MMEs, PCRF etc.) • Cache placement • Mobility events • QoS degradation • Cache overload/failure • GW overload/failure • Mobile backhaul link congestion/failure • Self-X algorithms for autonomic management of GW/Caches/links etc OUTPUTS : • On-the-fly prediction and classification of demanded load • Optimization of learning algorithms • Optimization of Self-X algorithms

Table 1: Network phases for the case

INNOVATION

Differentiation from the state of the art

- Current mobile networks are centralized. Therefore, the whole scenario which is considered here is fairly novel in itself.
- Current mobile core network functions (SGW, SGSN, …) run on dedicated hardware, while we propose to run them on a virtualization layer, which allows to flexibly migrate them to the most appropriate place in the physical topology.
- While virtualization has been studied in various contexts, virtualizing a mobile network requires specific considerations as migration on the virtual layer might require adaptations on the mobile network layer.
- There are existing works on load balancing of servers and caches (e.g. [1]), or of virtual machines (e.g. [2]). However, most of these works assume a load balancer that sits on the data path and routes new requests dynamically. In this case, no load balancer is assumed in the data path. Instead, there could be a load balancer that is external to the network as such, monitoring it and assigning mobile core functions to nodes based on load aggregates.
- When it com es to multi-site load distribution (as is the case for the core functions in this case), works consider data center health or load as possible criteria (e.g. [3]). However, for this case, not only the node but also the transport network load should ideally be considered.
- Migration of services and content has been studied mostly in the context of sensor networks, but also for data centers and content delivery networks. However, we propose to combine the algorithmic work on the placement and migration of services, content and mobile core functions as this allows for lots of joint optimization which would otherwise not be possible, as the interdependencies between these aspects are neglected.

Impacts and benefits

Dynamic migration of content, services and network functions will *reduce CAPEX*, since general-purpose, i.e. commodity hardware can be used to host them. It will also increase the *flexibility* for the deployment of new services, as all run on a virtualization layer that abstracts the physical view into the set of relevant parameters. This will be particularly beneficial for mobile network operators who have traditionally struggled with service introduction and deployment, as IMS is too inflexible to cope with Internet-style services such as YouTube, facebook and so on. Further, the *improved resource utilization* as well as the *autonomic runtime optimization* of the physical infrastructure – which is made possible by dynamic migration strategies – *reduces OPEX* for operators of such networks. More effectively distributed functions and services can also lead to *improved QoE* for end-users, as capacity bottlenecks can better be avoided and latency can principally be reduced if the communication paths are kept short.

TO BE CONTINUED

This document is the first part in a series covering the introduction, general description, problem statement and innovation of the case study on Dynamic Virtualization and Migration of Contents and Servers. Subsequent and complementary parts will be published in the near future, during the lifetime of the project with even more information, results and innovations.

Keep in touch to get premium access to these future reports!

REFERENCES

[1] C. Kopparapu, “Load balancing servers, firewalls, and caches”, Wiley Computer Publishing, 2002

[2] J. Hu, J. Gu, G. Sun, T. Zhao, “A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment”, Third International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), 2010

[3] Cisco Systems, “Data Center—Site Selection for Business Continuance”, http://www.cisco.com/en/US/docs/solutions/Enterprise/Data_Center/dcstslt.pdf

CONTACT INFORMATION

For additional information, please contact: *Johannes Lessmann (johannes.lessmann@neclab.eu); or consult* *www.univerself-project.eu*

UNIVERSELF CONSORTIUM