ABSTRACT

As cloud computing emerges as the next novel concept in computer science, it becomes clear that the model applied in large data storage systems used to resolve issues coming forth from an increasing demand, could also be used to resolve the very high bandwidth requirements on access network, core network equipment and long communication paths between users and servers in today’s cellular networks. The cloud computing model can be used to avoid bottlenecks, better utilise available resources, and minimise delay. This paper investigates how the cloud computing model could be applied into the LTE (Long Term Evolution) cellular system and analyses how three deployed major cloud computing platforms could support this. This analysis showed that none of the studied currently deployed cloud computing platforms can satisfy the derived requirements without enhancing and extending the functions and modules supported by each platform.

Keywords

Network virtualisation/cloudification, cellular systems, LTE

1. INTRODUCTION

Cloud computing is one of the most discussed and promising fields in modern computer science. Contrary to common belief cloud computing is not a form of implementation, but a model. Applying the cloud computing model to a system or part of a system can be denoted as cloudification. The implementation of the cloud computing model is typically denoted as virtualisation. The model used in this paper is defined by the National Institute for Standards and Technology (NIST) and states “Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction,” from [15]. The cloud computing model is mostly implemented in large data storage systems, providing Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and/or Software-as-a-Service (SaaS) service models. A visual model of this can be seen in Figure 1, showing not only the essential characteristics and service models, but also the various deployment models.

IaaS provides customers processing, storage, network and other fundamental computing resources on which they can run software. This includes operating systems and applications. This is done by providing the customer with a virtual machine. A virtual machine (VM) is a software implementation of a machine/computer and runs in much the same way as a physical machine. The idea is that multiple VMs can run on a single physical machine and that the software running inside the virtual machine is limited to the resources and abstractions provided by the virtualised environment.

PaaS is using the services provided by IaaS, and offers customers an environment for application development, testing, deployment and hosting without the complexity of managing the underlying infrastructure and supporting software. To facilitate the customer in such an environment the provider can support various programming languages, libraries, services and tools.

SaaS is using the services provided by PaaS and it delivers its customers on-demand applications running on a cloud computing platform. The applications are accessible, mostly over the Internet, through various clients like web browsers or thin clients. It is also possible to access the application through a programmable interface (API). SaaS customers do not have to be concerned about the underlying infrastructure and have no influence on the application features, apart from the user specific configuration settings.

Figure 1 shows that the cloud can be deployed in several ways. A private cloud is a cloud infrastructure operated solely for a single organization, whereas a public cloud is...
publicly available and its services are free or offered on a pay-as-you-go model (services are only paid for the time that they are used). Community clouds can be placed somewhere in between, where the cloud infrastructure is shared between multiple organisations such that the costs are spread over more users than in the private cloud and less than a public cloud. A hybrid cloud is the combination of two or more clouds, combining the advantages and disadvantages of those deployment models.

When providing IaaS, virtual machines are connected to each other in a virtual network. This is done by virtualising network components like cables, switches, routers and firewalls. This virtual network can be made as simple or complex as one can imagine [17, 16]. Network virtualisation in large data storage systems virtualises wired networks. There is no practical need to virtualise a wireless network in such systems.

In the world of today, the increasing use of smartphones, tablets and other portable devices using the cellular system, requires from the mobile cellular system to cope with this increasing demand. A mobile cellular network can be considered as a radio network that is distributed over radio coverage areas denoted as cells, which are usually controlled and served by fixed transmitter/receiver entities denoted as base stations. By connecting more cells together, the cellular system can provide radio coverage over a wide geographic area. The part of the cellular system that encompasses the base stations is denoted as a radio access network (RAN) and the part that is encompassing the entities that are interconnecting the different base stations and providing access to the Internet and other types of communication networks is denoted as Core Network.

One of the most advanced cellular systems (i.e., 4th generation) being currently under specification and standardisation by 3GPP (Third Generation Partnership Project) is the Long Term Evolution (LTE) system, see Figure [1].

Today’s mobile cellular networks are highly centralised and not optimised for high-volume data applications, which will evolve with 4G (e.g., LTE) and beyond technologies. Operators typically use centralised network architectures that lead to very high bandwidth requirements on core network equipment and long communication paths between users and servers, which wastes network resources and increases delay. Shared distributed mobile network architectures, are needed to avoid bottlenecks, better utilise available resources, and minimise delay.

The cloud computing model could be applied in mobile cellular systems to solve this issue by offering decentralised computing, smart storage, on-demand, elastic and Pay-as-you-Go services to third party operators and users, see e.g. [9]. There are a few papers available describing how virtual telecommunication operators are using the cloud computing model to provide their services, see [4, 12]. There are also papers that focus on how wireless networking can be virtualised [27, 28], and even ones on the virtualisation of a cellular system, see e.g., [25, 26].

However, not many research activities focused on how existing cloud computing platforms can be used to cloudify LTE cellular systems. This paper aims to resolve this dilemma. So the main research question is:

“How can the cloud computing model be applied in LTE cellular systems?”

In order to answer the main research question the research is broken down in sub questions.

1. What is network cloudification?
2. Are network cloudification solutions already applied in LTE based cellular systems?
3. What are the main differences between applying network cloudification in (1) large data storage systems and (2) LTE cellular systems?
4. Which challenges have to be solved when using current cloud computing platforms in LTE based cellular systems?

Research questions (1), (2) and (3) will be answered using literature study. The literature study will provide a list of challenges that current cloud computing platforms [8, 21, 22] need to resolve in order to apply the cloud computing model in LTE based cellular systems. Research question (4) will be answered by defining the main requirements needed to implement the cloud computing model in LTE based cellular systems and by studying how these requirements are satisfied by the specifications of the currently deployed cloud computing platforms.

This paper is organized as follows. Section 2 gives an overview of network cloudification and its use in large data storage systems. Moreover this section answers research question (1). Section 3 gives an overview of the state art on applying network cloudification in cellular systems. This section also provides an answer to research question (2). Section 4 defines the main requirements needed to implement the cloud computing model in LTE based cellular systems and analyses how these requirements are satisfied by the specifications of the currently deployed cloud computing platforms. Furthermore, this section answers research question (4). Finally, section 5 concludes and provides recommendations for future work.

2. NETWORK CLOUDIFICATION

The cloud computing model is used to maximise the utilisation of large data storage systems, resulting in clustering and pooling of its resources and the use of virtual machines [3]. Whereas in the past, a single physical machine was used just as a single processing or storage machine, nowadays it can host multiple virtual machines, each dedicated to its own task. This has the advantage of separating the physical infrastructure from its services and providing the service’s independence from its physical underlying hardware. This enables infrastructure providers to focus on the physical infrastructure and service providers to not be concerned about the physical limitations of the hardware. As a consequence, new business models and application areas will be enabled, see e.g., [15].

In large data storage systems, one or more virtual networks interconnect the virtual machines to the physical outside world and to each other, enabling them to provide their services. These virtual networks consist of a set of virtualised networking resources like routers, switches and links, see e.g., [10].

To optimise and extend the utilisation of networking resources, the cloud computing model is applied in networks in the same way as it has been applied in large data storage systems. The process of applying the cloud computing concept in networks can be denoted as network cloudification, see e.g., [24].

Network cloudification can therefore be defined as the process of virtualising a network that can for example be composed of a radio access network, mobile core network, virtual local area network (VLAN), a virtual private network
(VPN), active and programmable networks and overlay networks, see e.g., [17].

In addition to SAIL the following providers a combination and integration of cloud computing and virtual networking. In this situation the service providers can create and deploy virtual networks on-demand, that are elastic (i.e. virtualised resources can be dynamically allocated and released) and are based on a Pay-as-you-Go model (i.e., pay only for the resources that are allocated and used) to third party service operators and users [15, 17, 16, 18].

3. NETWORK CLOUDIFICATION IN LTE CELLULAR NETWORKS

Currently, several papers describe architectures that enable the cloud computing concept to be applied in communication networks.

For example, [3] briefly discusses some results of the EU FP7 project SAIL [23], related to cloud networking that offers a combination and integration of cloud computing and virtual networking. In addition to SAIL [23] the following projects are considering network virtualisation concepts: GEYSERS [11] and OFELIA [20].

In [13] a concept is discussed that applies network sharing mechanisms using virtualisation technologies to design a reconfigurable mobile network (RMN) architecture designed for flexibility and re-configurability. [5] discusses an IP Multimedia Subsystem (IMS) framework with cloud computing architecture for use in high quality multimedia applications. In [4] a Virtual Telco is discussed that uses a Cloud computing approach in order to replace the costly dedicated hardware. This is realised by implementing several centralised control plane functions and other services with distributed solutions that may be allocated on-demand over a pool of dependable, dynamically contracted computing and networking resources that are easy to manage. In [14] an algorithm is discussed that is used for the placement of virtualised service components to network sites, where the performance metric is the cost for acquiring components between the sites. The resulting optimisation problem, is referred to as the k-Component Multi-site Placement Problem, and is applicable to service distribution in a wide range of communication networking scenarios.

[27] discusses the distributed wireless communication system (DWCS), which is a virtualised architecture for a wireless access system with distributed antennas, distributed processors, and distributed control. When distributed antennas are used, the system capacity can be expanded through dense frequency reuse, and the transmission power can be greatly decreased. Moreover, by using distributed processors control, the system works like a software or network radio. In this way, different standards can coexist, and the system capacity can be increased by co-processing of signals to and from multiple antennas. [28] introduces the a TDD (Time Division Duplex) WiMAX SDR (Software Defined Radio) Base Station implemented on a commodity server, in conjunction with a novel design of a remote radio head (RRH). [28] also presents a working prototype of a virtual BS (VBS) pool, exploring the systems challenges in supporting a VBS pool on multicore IT platforms. Please note that [28] emphasises that the virtual base stations are implemented using the Open Base Station Architecture Initiative (OBSAI) [19] and the Common Public Radio Interface (CPRI) [7, 6].

The research activities mentioned above are describing possible ways of how network cloudification can be used in wireless networks. However, it is not clear whether these architectures are able to provide the same functional and performance capabilities as the ones provided by non-virtualised LTE networks.

Figure 3 shows a very basic architecture of the LTE Evolved Packet System (EPS) Architecture, based on [2].
The User Equipment (UE), e.g., smartphones and other 4G enabled devices, is connected to the LTE core network over the LTE radio access network, which consists of base transceiver stations, denoted as eNodeB (evolved NodeB). An eNodeB is responsible for accessing the radio channel and scheduling the air interface resources between the UE and the LTE core network. In LTE the radio access network is denoted as e-RAN (evolved - RAN) and the core network is denoted as EPC (Evolved Packet Core).

The EPC is composed of five network elements: the S-GW (Serving GW), the PDN GW (Packet Data Network Gateway), the MME (Mobility Management Entity), PCRF (Policy and Charging Control Element) and the HSS (Home Subscriber System).

**Home Subscriber System (HSS):** Is a database that contains user-related and subscriber-related information. It also provides support functions for mobility management, call and session setup, user authentication and access authorisation.

**Serving Gateway (S-GW):** It supports the transport of the user data between the UE and the external networks. The Serving GW connects the e-RAN and the EPC, routes the incoming and outgoing data and ensures a smooth handover between eNodeBs. The serving GW is connected to the PDN GW.

**Packet Data Network Gateway (PDN GW):** The PDN GW connects the EPC to other external networks. The PDN GW routes packets to and from the PDNs (Packet Data Networks). The PDN GW also performs various functions such as IP address / IP prefix allocation or policy control and charging. The two gateways (S-GW and PDN-GW) are specified independently but they can also be collocated. Both gateways (S-GW and PDN GW) support the user plane (i.e., user data).

**Mobility Management Entity (MME):** is supporting only the control plane (i.e., signalling and control data). It handles the signalling related to mobility and security for the e-RAN. The MME is responsible for the tracking and the paging of UE when is operating in idle-mode.

**Policy and Charging Control Element (PCRF):** it supports the control plane, and it provides features like policy control and charging control.

The EPC is connected to (1) the Operator’s services, which can include the IP Multimedia Subsystem (IMS) and can usually run on data centres and (2) external networks (like the Internet). The IMS is an LTE service platform to provide IP multimedia services to the LTE subscribers.

The EPS architecture separates the user data (user plane) and the signalling (control plane). Thanks to this functional split, the operators can dimension and adapt their network easily. The LTE network is an all-IP system, which means that data is carried in IP packets avoiding the need for conversion and so enabling the fast connectivity with other networks like the Internet.

To apply the cloud computing model in a LTE network infrastructure, virtualisation of all above mentioned components (e-RAN, EPC and Operators services platforms, like IMS (i.e., data centres)) is needed, enabling multiple mobile network operators to create their own virtual network on the same infrastructure [12, 25, 28]. Since over 80% of the infrastructure is software based, most of the infrastructure is already cloud computing enabled and can be virtualised by using e.g. virtual machines. So virtualising, for example, the eNodeB in the e-RAN could be similar to server virtualisation as is done in large data storage systems [26].

In [25], network virtualisation is applied in Long Term Evolution (LTE) and specifies a virtualised LTE framework that realises an auctioned based dynamic spectrum sharing.

The physical resources need to be shared between multiple virtual instances, which is done by a so called hypervisor and sits on top of the physical hardware to schedule the resources. This results in the fact that the physical eNodeB is virtualised into a number of virtual eNodeBs which can be seen in Figure 4. In addition, the hypervisor is responsible for scheduling the air interface resources between the virtual eNodeBs running on top. The “Spectrum configuration and Bandwidth estimation” in Figure 4 is responsible for allocating the spectrum that the virtual eNodeB is supposed to operate in, and as well as estimating the required bandwidth of the operator. The “Spectrum allocation unit” is responsible for scheduling the spectrum among the different virtual eNodeBs [25].

Figure 4 shows the virtualisation of the eNodeB and e-RAN to be used by multiple operators. Virtualising the EPC and the Operators service platform, e.g., IMS, can be done in the same manner and requires the virtualisation of the EPC and IMS entities, respectively. These entities will then be virtualised and can run on a common (or separated) cloud computing centres.

By virtualising the e-RAN, EPC and the Operators service platform, a cloudified LTE can be developed that is able to automate, manage and orchestrate the pool of resources to deliver cloud computing services. The cloudification of the LTE system requires one or more cloud computing platforms to cloudify the e-RAN, EPC and the Operator’s services infrastructures. In this way the LTE virtual networks that are elastic and are based on a Pay-as-you-Go model can be enabled on-demand, to third party LTE service operators and users.

4. APPLYING EXISTING CLOUD COMPUTING PLATFORMS IN LTE SYSTEMS

4.1 Research Methodology
In order to analyse cloud computing platforms for their ability to be used in cloudifying LTE systems, criteria need
The OpenStack cloud operating system enables modules:

1. Support virtualised foundational infrastructural resources from the Radio Access Network and Mobile Core Network and Data Centre.
2. Support a coordination and orchestration function from the infrastructure perspective for the composition, provisioning, life cycle management of services and maintenance of agreed SLA guarantees spread across Radio Access Network and Core Network.
3. Support monitoring facilities for extraction, preprocessing, distribution, storage, analysis and notification of metrics such that higher-level services can utilise these facilities to enable adaptive platform services (such as SLAs), where certain guarantees and service quality (QoS) and/or experience (QoE) must be maintained.
4. Support mechanisms to deliver virtual resources as service for the execution of non-traditional virtualised instances such as virtualised eNodeB, EPC (e.g., MME, PGW, SGW) and IMS functionality.
5. Support unified and consistent interfaces on top of and between disparate cloud computing physical infrastructure management frameworks (i.e., Radio Access Network, Mobile Core Network, Data centre).

The analysis will be done using the documentation of the latest available version of the cloud computing platforms, see Table 1.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Version</th>
<th>Release date</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenStack [22]</td>
<td>2012.1 (Essex)</td>
<td>04-05-2012</td>
</tr>
<tr>
<td>Eucalyptus [8]</td>
<td>2.0.3</td>
<td>25-05-2011</td>
</tr>
<tr>
<td>OpenNebula [21]</td>
<td>3.4.1 (Wild Duck)</td>
<td>03-05-2012</td>
</tr>
</tbody>
</table>

4.2 Cloud computing platforms

4.2.1 OpenStack

OpenStack Essex, [22] the fifth version of OpenStack, focuses on quality, usability and extensibility across enterprise, service provider and high performance computing deployments. OpenStack Essex allows users to leverage pools of on-demand, self-managed compute, storage and networking resources to build efficient, automated private and public cloud infrastructures.

OpenStack Essex includes, but is not limited to, the following modules:

- **Compute**: The OpenStack cloud operating system enables service providers to offer on-demand computing resources, by provisioning and managing large networks of virtual machines. Compute resources are accessible via APIs for developers building cloud applications and via web interfaces for administrators and users. The Compute architecture is designed to scale horizontally. OpenStack is designed to provide flexibility and to manage and automate pools of Compute resources and can work with multiple supported hypervisors in a virtualised environment like KVM (Kernel-based Virtual Machine) and Xen. Linux container technology such as LXC (Linux Containers) is also supported for scenarios where users need to minimise virtualisation overhead and gain greater efficiency and performance. In addition to different hypervisors, OpenStack supports ARM (Advanced RISC-Reduced Instruction Set Computer Machine) and alternative hardware architectures.

- **Object Storage**: OpenStack provides redundant, scalable object storage using clusters of standardized servers capable of storing petabytes of data. Object Storage is not a traditional file system, but rather a distributed storage system for static data such as virtual machine images, photo storage, email storage, backups and archives. Having no central "brain" or master point of control provides greater scalability, redundancy and durability. Objects and files are written to multiple disk drives spread across servers in the data centre, with the OpenStack software responsible for ensuring data replication and integrity across the cluster. Storage clusters scale horizontally simply by adding new servers.

- **Block Storage**: OpenStack also provides persistent block level storage devices for use with OpenStack compute instances. The Block Storage system manages the creation, attaching and detaching of the block devices to servers. Block Storage volumes are fully integrated into OpenStack Compute and the Dashboard allowing for cloud users to manage their own storage needs. In addition to using simple Linux server storage, it has unified storage support for numerous storage platforms. Block Storage is appropriate for performance sensitive scenarios such as database storage, expandable file systems, or providing a server with access to raw block level storage.

- **Networking**: OpenStack Networking is a pluggable, scalable and API-driven system for managing networks and IP addresses. OpenStack provides flexible networking models to suit the needs of different applications or user groups. Standard models include flat networks or VLANs for separation of servers and traffic. OpenStack Networking manages IP addresses, allowing for dedicated static IPs or DHCP (Dynamic Host Configuration Protocol). Floating IPs allow traffic to be dynamically rerouted to any of the Compute resources, which allows to redirect traffic during maintenance or in the case of failure. Users can create their own networks, control traffic and connect servers and devices to one or more networks. Administrators can take advantage of software-defined networking (SDN) technology like OpenFlow to allow for high levels of multi-tenancy and massive scale. OpenStack Networking has an extension framework allowing additional network services, such as intrusion detection systems (IDS), load balancing, firewalls and virtual private networks (VPN) to be deployed and managed.

- **Dashboard**: The OpenStack Dashboard provides administrators and users a graphical interface to access,
provision and automate cloud-based resources. The extensible design makes it easy to plug in and expose third party products and services, such as billing, monitoring and additional management tools. Developers can automate access or build tools to manage their resources using the OpenStack API or the Amazon EC2 (Elastic Compute Cloud) compatibility API. The dashboard is an extensible web app that allows cloud administrators and users to control their compute, storage and networking resources. As a cloud administrator, the dashboard provides an overall view of the size and state of the cloud and is able to create users and projects, assign users to projects and set limits on the resources for those projects. The Dashboard provides users a self-service portal to provision their own resources within the limits set by administrators.

4.2.2 Eucalyptus

Eucalyptus [8] is a widely deployed software platform for private IaaS clouds. It uses existing infrastructures to create a scalable, secure web services layer that abstracts compute, network and storage to offer IaaS. Eucalyptus takes advantage of modern infrastructure virtualisation software to create elastic pools that can be dynamically scaled up or down depending on application workloads. Eucalyptus web services are uniquely designed for hybrid clouds using the industry standard Amazon Web Services (AWS) API. Eucalyptus key functionality includes:

Self-service User Portal: The Eucalyptus User Console provides an interface for users to self-service provision and configure compute, network, and storage resources. Development and test teams can flexibly and securely manage their virtual instances using built-in key management and encryption capabilities. Access to virtual instances is available using familiar SSH (Secure Shell) and RDP (Remote Desktop Protocol) mechanisms. Virtual instances with application configuration can be stopped and re-started securely using encrypted boot from Elastic Block Storage (EBS) capability.

High Availability Cloud IaaS: Systems failure is expected and unavoidable in an IaaS cloud. Eucalyptus has been designed to be highly available at multiple levels. IaaS service components include Cloud Controller (CLC), Cluster Controller (CC), Walrus, Storage Controller (SC) and VMware Broker (VB) are configurable as redundant systems that are very resilient to multiple types of failures. Management state of the cloud machine will be preserved and reverted to normal operating conditions in the event of a hardware or software failure.

Heterogeneous Hypervisor Management: Build and manage multiple hypervisor cluster environments in an IaaS cloud. Manage existing VMware vSphere, VMware ESX (Elastic Sky X), VMware ESXi, KVM, and Xen virtual environments from a single pane as Amazon AWS compatible Eucalyptus hybrid clouds.

Manage Multiple Machine Image Formats: Eucalyptus can run multiple versions of Microsoft Windows and Linux virtual machine images on IaaS clouds and build a library of Eucalyptus Machine Images (EMIs) with application metadata that are decoupled from infrastructure details to allow them to run on Eucalyptus clouds. Amazon Machine Images are also compatible with Eucalyptus clouds. VMware Images and VMware vApps can be converted to run on Eucalyptus clouds, and Amazon AWS and Amazon compatible public clouds.

Scriptable Architecture: Production environments want flexible, configurable and scriptable system to automate as many functions as possible in a cloud management platform. Users have the choice of integrating web services API with their preferred vendor management systems or using an Amazon AWS compatible management stack to manage Eucalyptus clouds. All the functionality is exposed in a scriptable interface to work with other system management tools.

Identity Management: Eucalyptus user identity management can be integrated with existing Microsoft Active Directory or LDAP (Lightweight Directory Access Protocol) systems to have fine-grained role based access control over cloud resources. Flexible Accounting, Chargeback, and Quota Management: Define and allocate resource quotas for the users and groups with Eucalyptus 3’s quota management features. Control resource allocation across clusters, defined by users and groups.

Accounting, Chargeback, and Quota Management: Define and allocate resource quotas for users and groups with quota management features. Control resource allocation across clusters, defined by users and groups and analyse cloud usage patterns by using resource accounting module. Compute and storage usage data are available in various formats for visualization, reporting, and analysis by business systems for both enterprises and integration with chargeback and billing platforms.

Resource Administration And Console: The Eucalyptus Dashboard provides cloud administrators with a graphical console for performing several cloud management tasks including all virtual and physical resource management and virtual cloud resource configuration, provisioning and de-provisioning.

4.2.3 OpenNebula

OpenNebula [21] is a cloud computing toolkit for managing heterogeneous distributed data centre infrastructures. The OpenNebula toolkit manages a data centre virtual infrastructure to build private, public and hybrid IaaS clouds. OpenNebula orchestrates storage, network, virtualisation, monitoring, and security technologies to deploy multi-tier services (e.g. compute clusters) as virtual machines on distributed infrastructures, combining both data centre resources and remote cloud resources, according to allocation policies.

OpenNebula key features include:

User Security Management: Secure and efficient users and groups pluggable subsystem for authentication and authorisation of requests with complete functionality for user management. The authorisation framework allows multiple-role support for different types of users and administrators, delegated control to authorised users, secure isolated multi-tenant environments, and easy resource (VM template, VM image, VM instance, virtual network and host) sharing. Administrators have complete functionality for
management of grouped users where each group has configurable access to shared resources so enabling a multi-tenant environment with multiple groups sharing the same infrastructure.

**On-demand Provision of Virtual Data Centres:**
A Virtual Data Centres (VDC) is a fully-isolated virtual infrastructure environment where a group of users, under the control of the VDC administrator, can create and manage compute, storage and networking capacity. There is also support for the creation and management of multiples VDCs within the same logical cluster and zone.

**Control and Monitoring of Virtual Infrastructure:**
OpenNebula has a Storage and Repository Subsystem with functionality for VM image and template management. The platform also has full control of VM instance life-cycle and functionality for VM instance management. OpenNebula also has broad network virtualisation capabilities with traffic isolation, ranged or fixed networks, definition of generic attributes to define multi-tier services consisting of groups of inter-connected VMs, and functionality for virtual network management to interconnect VM instances. The tagging of users, VM images and virtual networks with arbitrary metadata that can be later used by other components and API’s future use.

**Virtual Machine Configuration:**
Complete definition of VM attributes and requirements and the support for automatic configuration of VMs with advanced contextualization mechanisms with a wide range of guest operating system including Microsoft Windows and Linux. For networking, OpenNebula offers flexible network definition including configuration of firewalls for VMs to specify a set of black/white TCP / UDP ports.

**Control and Monitoring of Physical Infrastructure:**
OpenNebula is able to deploy public, private and hybrid clouds. The Host Management Subsystem offers functionality for management of physical hosts with the dynamic creation of clusters as a logical set of physical resources, namely: hosts, networks and data stores, within each zone. OpenNebula has a powerful and extensible built-in monitoring subsystem and resource quota management to allocate, track and limit resource utilisation. The virtualisation Subsystem supports a wide range of hypervisors like Xen, KVM and VMware ESX, centralised management of environments with multiple hypervisors, and support for multiple hypervisors within the same physical box. Storage Subsystem with support for multiple data stores to balance I/O operations between storage servers, or to define different SLA policies (e.g. backup) and performance features for different VM types or users.

**Standard Cloud Interfaces and Self-Service Portal:**
OpenNebula is able to simultaneously expose multiple cloud APIs including the AWS EC2 API service for compatibility with EC2 ecosystem tools and client tools. OpenNebula also has a self-service provisioning portal to allow non-IT end users to easily create, deploy and manage compute, storage and network resources and a Unix-like Command Line Interface to manage all resources: users, VM images, VM templates, VM instances, virtual networks, zones, VDCs, physical hosts, accounting, authentication, authorisation for cloud administrators. The modular and extensible Graphical Interface providing usage graphics and statistics with cloudwatch-like functionality, VNC support, different system views for different roles, catalogue access, multiple-zone management.

### 4.3 Analysis and challenges

The analysis starts by mapping the features and functionality of the cloud computing platforms to the criteria introduced in Section 4.1. The analysis can be found in Table 2, 3 and 4 showing the criteria and the modules which could be used to support each criterion. The criteria are numbered according to the numbering used in Section 4.1.

It is important to be noticed that the functions/modules (including the APIs) supported by the three cloud computing platforms are able to support wired and fixed virtualisation scenarios that are making use of large data centres.

These functions/modules (including the APIs) are not designed and implemented to support virtualised scenarios where the allocation and release of e-RAN, EPC and Operators’ services (e.g., IMS) resources takes place in a very dynamic and fast changing environment. Therefore, in order to support the five criteria described in Section 4.1, see Table 5, these functions/modules (including the APIs) need to be enhanced such that they can support challenges, such as dynamic network access connectivity and release, QoS support, cross-operator dynamic mobility and roaming management, security and privacy support, billing and charging using a Pay-as-you-go policy for an environment where the allocation and release of e-RAN, EPC and Operators’ services (e.g., IMS) resources takes place in a very dynamic and fast changing environment.

It needs to be noted that depending on the operators requirements, the use of additional not in this paper discussed functions/modules could be used. However, those additional modules probably also need to be enhanced in order to support the criteria given in Section 4.1.

Table 2, 3 and 4 show the criteria and the modules of OpenStack, Eucalyptus and OpenNebula, respectively, which need to be enhanced in order to support these criteria.

**Table 2. Analysis OpenStack**

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Modules to be enhanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Compute and Networking module</td>
</tr>
<tr>
<td>2</td>
<td>Compute and Networking module</td>
</tr>
<tr>
<td>3</td>
<td>Dashboard</td>
</tr>
<tr>
<td>4</td>
<td>Compute and Networking module</td>
</tr>
<tr>
<td>5</td>
<td>Dashboard</td>
</tr>
</tbody>
</table>

Note that all three cloud computing platforms have support for billing and charging using a Pay-as-you-go policy which can be integrated by their API into other financial systems of the operator.

The analysis is summarised in Table 5 and clearly shows how the cloud computing platforms compare to each other and whether or not criteria have fully be met or not.

Table 2, 3, 4 and 5 show that enhancements and extensions need to be made in order to support challenges, such as dynamic network access connectivity and release, QoS support, cross-operator dynamic mobility and roaming management, security and privacy support, billing and
### Table 3. Analysis Eucalyptus

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Could be supported by</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Heterogeneous Hypervisor Management, Manage Multiple Machine Image Formats and High Availability Cloud IaaS</td>
<td></td>
</tr>
<tr>
<td>2 Self-service User Portal, Resource Administration And Console and Identity Management</td>
<td></td>
</tr>
<tr>
<td>3 Self-service User Portal, Resource Administration And Console and Identity Management</td>
<td></td>
</tr>
<tr>
<td>4 Scriptable Architecture, Heterogeneous Hypervisor Management, Manage Multiple Machine Image Formats and High Availability Cloud IaaS</td>
<td></td>
</tr>
<tr>
<td>5 Scriptable Architecture, Self-service User Portal, Resource Administration And Console and Identity Management</td>
<td></td>
</tr>
</tbody>
</table>

### Table 4. Analysis OpenNebula

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Could be supported by</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Control and Monitoring of Physical Infrastructure and On-demand Provision of Virtual Data Centres</td>
<td></td>
</tr>
<tr>
<td>2 Control and Monitoring of Virtual Infrastructure and Virtual Machine Configuration</td>
<td></td>
</tr>
<tr>
<td>3 User Security Management and Standard Cloud Interfaces and Self-Service Portal</td>
<td></td>
</tr>
<tr>
<td>4 On-demand Provision of Virtual Data Centres, Control and Monitoring of Virtual Infrastructure and Virtual Machine Configuration</td>
<td></td>
</tr>
<tr>
<td>5 User Security Management and Standard Cloud Interfaces and Self-Service Portal</td>
<td></td>
</tr>
</tbody>
</table>

5. CONCLUSIONS AND FUTURE WORK

This paper investigated how the cloud computing model could be applied into the LTE (Long Term Evolution) cellular system and analyses how three deployed major cloud computing platforms could support this.

Section 2 answered research question (1) by providing an overview of network cloudification and its use in large data storage systems. Section 3 answered research question (2) by providing an overview of the state art on applying network cloudification in cellular systems. Section 4 answers research questions (3) and (4) by defining the main requirements needed to implement the cloud computing model in LTE based cellular systems and analyses how these requirements are satisfied by the specifications of the currently deployed cloud computing platforms.

In particular, this paper showed that none of the three major current cloud computing platforms fully satisfy the criteria from Section 4.1, as the most of the existing functions/modules cannot support the challenges that need to be fulfilled by scenarios where the allocation and release of e-RAN, EPC and Operators’ services (e.g., IMS) resources take place in a very dynamic and fast changing environment.

It is expected that these criteria could be satisfied if, at least, the functions/modules listed in Table 2, 3, 4 are enhanced in order to fulfill challenges, such as dynamic network access connectivity and release, QoS support, cross-operator dynamic mobility and roaming management, security and privacy support, billing and charging using a Pay-as-you-go policy for an environment where the allocation and release of e-RAN, EPC and Operators’ services (e.g., IMS) resources take place in a very dynamic and fast changing environment.

The functions/modules that must be enhanced are as follows:

- **OpenStack**
  - Compute
  - Networking
  - Dashboard

- **Eucalyptus**
  - Heterogeneous Hypervisor Management
  - Manage Multiple Machine Image Formats
  - High Availability Cloud IaaS
  - Self-service User Portal
  - Resource Administration And Console
  - Identity Management
  - Scriptable Architecture

- **OpenNebula**
  - Control and Monitoring of Physical Infrastructure
  - On-demand Provision of Virtual Data Centres
  - Control and Monitoring of Virtual Infrastructure
  - Virtual Machine Configuration
  - User Security Management
  - Standard Cloud Interfaces and Self-Service Portal

5. CONCLUSIONS AND FUTURE WORK

This paper investigated how the cloud computing model could be applied into the LTE (Long Term Evolution) cellular system and analyses how three deployed major cloud computing platforms could support this.

Section 2 answered research question (1) by providing an overview of network cloudification and its use in large data storage systems. Section 3 answered research question (2) by providing an overview of the state art on applying network cloudification in cellular systems. Section 4 answers research questions (3) and (4) by defining the main requirements needed to implement the cloud computing model in LTE based cellular systems and analyses how these requirements are satisfied by the specifications of the currently deployed cloud computing platforms.

In particular, this paper showed that none of the three major current cloud computing platforms fully satisfy the criteria from Section 4.1, as the most of the existing functions/modules cannot support the challenges that need to be fulfilled by scenarios where the allocation and release of e-RAN, EPC and Operators’ services (e.g., IMS) resources take place in a very dynamic and fast changing environment.

It is expected that these criteria could be satisfied if, at least, the functions/modules listed in Table 2, 3, 4 are enhanced in order to fulfill challenges, such as dynamic network access connectivity and release, QoS support, cross-operator dynamic mobility and roaming management, security and privacy support, billing and charging using a Pay-as-you-go policy for an environment where the allocation and release of e-RAN, EPC and Operators’ services (e.g., IMS) resources take place in a very dynamic and fast changing environment.

The functions/modules that must be enhanced are as follows:

- **OpenStack**
  - Compute
  - Networking
  - Dashboard

- **Eucalyptus**
  - Heterogeneous Hypervisor Management
  - Manage Multiple Machine Image Formats
  - High Availability Cloud IaaS
  - Self-service User Portal
  - Resource Administration And Console
  - Identity Management
  - Scriptable Architecture

- **OpenNebula**
  - Control and Monitoring of Physical Infrastructure
  - On-demand Provision of Virtual Data Centres
  - Control and Monitoring of Virtual Infrastructure
  - Virtual Machine Configuration
  - User Security Management
  - Standard Cloud Interfaces and Self-Service Portal
The functions/modules of the cloud computing platforms OpenStack, Eucalyptus and OpenNebula could satisfy several of the derived criteria after solving several challenges. Once enhancements have been made in the above specified functions/modules, the enhanced cloud computing platforms could be used to enable realisation of the cloudification of the LTE cellular system.

6. REFERENCES