Interconnected Content Distribution in LTE Networks

Christian Schwartz*, Jochen Eisl†, Artan Halimi‡, Albert Rafetseder§, and Kurt Tutschku§
*University of Wuerzburg[1], Chair of Communication Networks
christian.schwartz@informatik.uni-wuerzburg.de
†Nokia Siemens Networks GmbH & CO KG, Munich, Germany
jochen.eisl@nsn.com
‡Telekom Austria, Wien, Austria
Artan.Halimi@telekom.at
§University of Vienna, Chair of Future Communication, Vienna, Austria
{ albert.rafetseder, kurt.tuschku }@univie.ac.at

***Abstract*—Consumption of multimedia content via the Internet by home users is on the rise. In addition to the delivery via traditional media like TV, many live events are streamed via the Internet. The increased availability of smart-phones, netbooks and high wireless access bandwidth encourages users to consume content on-the-go via mobile technologies. This work identifies requirements necessary to perform internetwork content distribution, proposes a general architecture which fulfills these requirements and gives and discusses an exemplary instantiation of the architecture for Long Term Evolution (LTE) networks. Major focus will be given in this paper in the mapping of a general architecture for interconnected and federated CDNs to the expected and available interfaces in LTE architectures.**

I. Introduction

Platforms like YouTube and Facebook use Content Distribution Networks (CDNs) to deliver content to their users. These CDNs usually work by placing content caches close to the user (for a short overview of the used terminology and current techniques see Section II-A) to provide an adequate Quality of Experience (QoE) to the user. While this quality is usually sufficient, mostly due to over-provisioning on the Internet Service Provider (ISP)'s part, it cannot be guaranteed. These QoE guarantees may become necessary, if either the content provider or the content consumer require them. In these cases ISP support is required. Even then, guarantees can only be made for the ISPs's network, as internetwork Quality of Service (QoS) is even more difficult to achieve (see Section II-B). If, for example, the streaming of live media is considered, it is not sufficient to place CDN caches in each operators network and rely on the ISP's QoS guarantees, because caching the live data in each ISPs network would introduce unwanted delays. In these cases the live content is ingested at the network of one ISP and is deployed to the consumer in other ISPs' networks without a detour via a cache. Hence, it is not adequate to rely on intra-network QoS guarantees. Thus, this work will focus on internetwork QoS in networks whose operators explicitly agree to cooperate in terms of achieving QoS. Examples for this scenario include:

- A telecommunication company has subsidiaries for mobile communication, for its traditional ISP business, as well as for Internet Protocol television (IPTV) delivery. The company wants to provide content delivery via all three platforms.
- A multinational telecommunication company wants to serve content to all its national subsidiaries.
- A group of ISPs wants to cooperate to provide QoS to their customers across network borders.

This work is structured as follows. After this introduction, Section II specifies the identified requirements, while Section III presents a general architecture which fulfills these requirements and is referred to in the remainder of this document. This architecture can be implemented in a tailor-made way to meet specific application needs. Section IV describes how this architecture can be used to distribute content in LTE networks. This architecture will be used to distribute live streaming content, which is among the most demanding contents to distribute due to realtime and QoS constraints. Finally, Section V summarizes the results of this work and concludes.

In this work, the following terminology is used. As shown in Figure 1 the process of inserting new content in the CDN by the content provider is called *ingestion*. After the ingestion, the content is distributed on the caches of the CDN. This process is called *deployment*. Once the content is deployed, consumers may request that the content be transferred to them. This process is called *delivery*.

II. Requirements

In this section, a general set of requirements will be identified. These requirements have to be satisfied by any architecture used for internetwork content distribution.

A. Local and Remote Content Distribution

First, we require the distribution of content to both the local and remote networks. This requirement stems from the basic idea of internetwork content distribution, and is usually fulfilled by classic CDNs [1], e.g. by placing a cache in

[1] Currently on leave at the University of Vienna.

978-1-4244-8865-0/10/$26.00 ©2010 IEEE

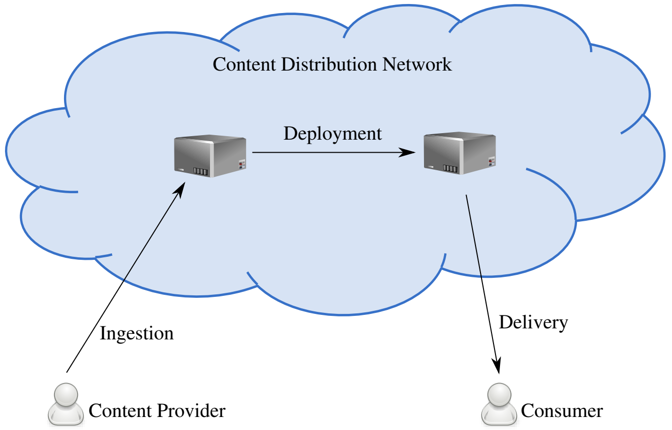


Fig. 1. CDN terminology overview

each local network and using Domain Name System (DNS) redirection to point the content consumers to the appropriate content caches.

B. Internetwork QoS

In addition, we require that the architecture supports QoS notifications (i.e. the forwarding of QoS information to interested parties), QoS enforcement (i.e. the application of QoS rules) and QoS measurements (i.e. reporting the perceived QoS in the network). The networks should be able to exchange QoS information on a need-to-know basis. Content distribution should be monitored in such a way, that the actually perceived QoS can be measured. In addition, content should be associated with meta-attributes describing the QoS requirements. If the network had earlier provided notification' that the required QoS is available, the adherence to this promise should be enforced.

C. Delivery Agnosticism

Furthermore, we require that the general architecture is agnostic regarding the content delivery method. Many different delivery methods (see Figure 2) exist, for example Unicast, Multicast, Peer to Peer methods, and hybrid approaches. In Unicast (see Figure 2a), for example used in Hyper Text Transfer Protocol (HTTP) transfers, one connection is established from the content server to each content consumer. In Multicast (see Figure 2b), the content consumers join a multicast tree and the content is replicated in the network. Using this method, the content server does not need to know about the number or locations of the consumers. Multicast is usually used for IPTV [2]. Another method for content distribution is Peer to Peer downloading (see Figure 2c). In these methods, no strict distinction between content servers and content consumers exists. Peer to Peer downloads can be used for file downloads, though recent work investigates the use of Peer to Peer in live streaming environments (see [3]). Hybrid approaches, for example Peer to Peer mechanisms using caches, are currently under study [4].

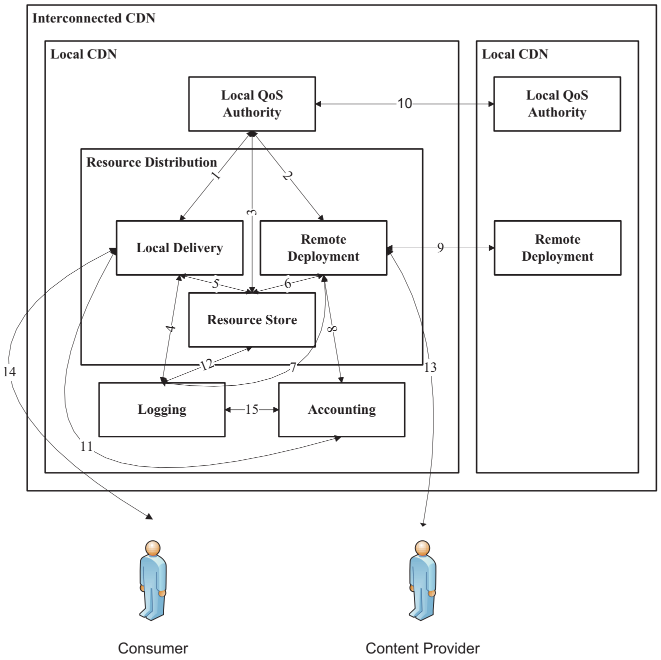


Fig. 3. Interactions in the general architecture

D. Content Agnosticism

In addition to the multitude of delivery types, several content types exist as well. Each of the content types has different requirements regarding QoS, supported delivery methods and additional infrastructure in the network. Aside from video delivery, file downloads, and multimedia streaming should be supported.

III. General Architecture

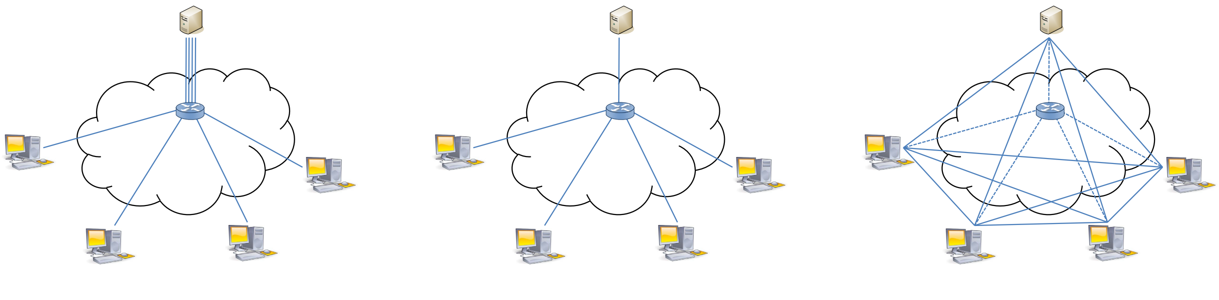
This section introduces a general architecture for interconnected CDNs. Specific realizations of each of the general features can then be defined according to the actual use case, for example an interconnected CDN specialized on live-streaming video over LTE networks (see Section IV).

A. Architecture Components

Figure 3 shows the components and interactions of the general architecture. Note that the interactions between components, drawn as arrows between them, are introduced in Section III-B, while the current section focuses on the components themselves. The Interconnected CDN consists of the Local CDNs of all participating entities. In contrast to all other components, it has no direct realization in hardware. A set of contracts, configuration files and public/private keys are artefacts of the Interconnected CDN and assure the cooperation of the different local CDNs.

The Local CDNs in turn consist of the components described in the following sections. They are realized in hardware and software, and share configuration files to ensure their correct interaction.

1) Resource Distribution: The two main responsibilities of the Resource Distribution component are the interactions



(a) Unicast

(b) Multicast

(c) Peer to Peer

Fig. 2. Delivery Methods

with content consumers and content providers. These responsibilities are delegated to the Local Delivery and Remote Deployment subcomponents described below, respectively.

a) Local Delivery: The Local Delivery components provides a front office functionality for content consumers. Possible implementations of such a front office component include Set-Top boxes, comparable to those used to provide IPTV to the consumer in most triple-play solutions. A web-browsable media catalogue, similar to YouTube could be maintained by the Service Provider, to allow consumers to acquire the provided content. The actual delivery mechanisms depend on the content type to be distributed. Furthermore, some content types may require the CDN provider to fulfill certain QoS requirements, which in turn will be specified in the Resource Store (see Section III-A1c) component. Note, that these features are reflected in the requirements specified in Section II,

b) Remote Deployment: The Remote Deployment component delivers content ingested into a Local CDN to other members of the Interconnected CDN. Furthermore, content received from Remote Distribution components in other CDNs's is handed over to the Local Delivery component by the Remote Deployment component.
Due to the requirements specified in Section II, the Remote Deployment component must be able to support multiple modes of content deployment as required by the content type. Moreover, the component should be designed to support additional new deployment methods required by future content types.

c) Resource Store: The Resource Store's duties are twofold. On the one hand, it accepts content files and associated meta-data. The storage of the content is delegated to an existing solution from which a reference to the actual storage location is obtained. This reference is stored together with the content's meta-data in an index of all content available in the local CDN. On the other hand, the Resource Store acts as an index server and answers queries regarding the location and meta-data of content.

2) Local Quality of Service Authority: The local Quality of Service Authority stores QoS properties as defined per Service Level Agreement (SLA) or Service Level Specification (SLS) with peering partners. It provides an interface to query QoS attributes for connections to peers, to query end-to-end QoS information based on information received from Local QoS Authorities in other CDNs, and to store information about the perceived QoS. QoS attributes to monitor might include:

- *Throughput*. The amount of data which can be transferred from the content distribution server to the content consumer in a given time span. Higher quality content is larger in size and, if the content type depends on realtime playback, requires greater bandwidth to transfer.
- *Delay*. The time a network packet requires to reach the receiver after being sent. Delay is usually measured as the first moment (or mean) of the packet latency. Delay is generally not a problem for elastic traffic (like downloads), but it becomes a problem during interactive streaming applications such as Voice over IP (VoIP) communication.
- *Jitter*. Changes in the inter arrival time of packets. It is measured as the variance of packet latency. While non-interactive streaming content can tolerate delay, it suffers greatly from jitter. During playback, received content is placed in a buffer, from which it is displayed. If the delay remains the same, it is only noticeable by the user in the beginning as the time it takes for the playback to start. If, however, jitter occurs (e.g. packet latency increases) the buffer empties. Once the buffer is completely empty, the playback has to be stopped, disturbing the user experience.
- *Packet loss* is the number of packets that are sent, but not received. It influences content distribution in two ways. If reliable transfer (e.g. Transmission Control Protocol (TCP)) is used, packet loss causes the retransmission of packets, which decreases the actual throughput and may introduce delay or jitter. If unreliable transfer (e.g. User Datagram Protocol (UDP)) is used, the lost packets are not resent. This usually leads to a reduction of content quality and decreases QoE.

In addition to these metrics, statistical properties of the metrics should be logged. Depending on the contracts between content providers and CDN operators, an auditing mechanism may be required. Note that the existence of the Local QoS authority does by no means suggest that the general architecture requires QoS support. But if the ISP supports QoS the architecture provides mechanisms to propagate this information.

3) Logging: The logging component receives messages to log from other components. It provides the ability to query the logged data. Furthermore an interface to the organization reporting framework should be provided. Moreover, privacy laws should be evaluated regarding information that has to be relayed to consumers about the information stored about them.

4) Accounting: The accounting module provides means of billing the usage of services such as resource delivery. An interface to the local charging infrastructure should be defined. The actual content and the process to modify the accounting database is out of scope for this document and depends on the business model of the CDN provider.

B. Component Interactions

The interactions described in this section are depicted in Figure 3. Each interaction is shown as an arrow between the two participating components and labeled with a number. This number corresponds to the paragraph number in this section. The interaction between the components Local QoS Authority and Local Delivery is labeled with the number *1* and described in III-B1.

1) Local QoS Monitoring: During the content delivery in the Local CDN the perceived QoS metrics are monitored. These metrics are delivered to the local QoS authority where they are stored and aggregated until they are polled by Interaction III-B3.

2) QoS Control: Before the Remote Deployment component deploys the content to the Local CDN the Local QoS Authority is queried to check whether the QoS requirements are met. Furthermore, before the Remote Deployment component deploys content to a remote CDN, the local QoS authority is queried to check whether the path to the remote CDN and the remote CDN itself fulfill the QoS requirement specified in the contents meta-data.

3) Local QoS Polling: If the meta-data of a distributable content specifies QoS constraints, the Resource Store will poll the Local QoS Authority in regular intervals to query the perceived QoS metrics for this content. The results of this interaction will be logged (see Interaction III-B12).

4) Delivery Logging: The delivery of content in a Local CDN will be logged.

5) Content Request: The Local Delivery component requests and receives the location and meta-data for content.

6) Local Content Ingestion: Newly ingested content and its meta-data will be stored in the Resource Store. If the content requires QoS constraints, a polling mechanism is scheduled (See Interaction III-B3) during the playback. The content ingestion will be logged (see Interaction III-B12).

7) Deployment Logging: If content is ingested in a CDN by a content provider, logging is performed. Furthermore, if content is deployed to the Local CDN from a remote CDN, logging is performed.

8) Deployment Accounting: Once new content is received, the Accounting component is notified, if delivery of this content requires accounting operations. Furthermore, if the deployment of the content to this CDN entails costs, these will be processed by the Accounting component. Any accounting operation will be logged (see Interaction III-B15)

9) Remote Deployment: Content meta-data may either blacklist or whitelist certain Local CDNs. Depending on this information and the QoS in the remote CDNs (see Interaction III-B2) the content will be deployed to the remote deployment component of the other CDN. The content will be stored in the Resource Store (see Interaction III-B6) of the remote CDN. If the deployment of the content to this CDN entails costs, the accounting component will be notified (see Interaction III-B8). The deployment process will be logged (see Interaction III-B7).

10) QoS Information Propagation: QoS attributes of the local CDN will be distributed to other CDNs. The possibility to limit QoS attribute distribution on a need-to-know basis should be studied.

11) Local Accounting: If content is requested from the local delivery, and the content meta-data received from the resource store (see Interaction III-B5) requires accounting operations, the local deployment component verifies that the requesting user has permission to receive the content. If the user has no permission and the content meta-data specifies the possibility of an instant purchase the accounting component takes care of the required operations.

12) Resource Logging: Content ingestion and meta-data updates will be logged.

13) Content Ingestion: The content provider wants to distribute content and uploads the content itself and the related meta-data to the remote deployment component. The remote deployment component verifies, that the requested QoS constraints can be fulfilled in all desired local CDNs (see Interaction III-B2). The accounting component is informed of the content ingestion (see Interaction III-B8). Furthermore, the content is both stored in the resource store (see Interaction III-B13) and deployed to the remote CDNs (see Interaction III-B9). Logging is provided for all actions (see Interaction III-B7).

14) Content Delivery: The consumer requests content from the local delivery component. The content location and meta-data is requested from the resource store (see Interaction III-B5). The local delivery component verifies the access rights of the user and, if required, performs accounting operations (see Interaction III-B11). Once transfer of the data to the consumer has been started, the perceived QoS is monitored and sent to the local QoS authority (see Interaction III-B1). During these steps, logging operations are performed (see Interaction III-B4).

15) Accounting Logging: All accounting operations will be logged.

IV. Application in LTE Networks

Having presented an architecture for the federation of distributed CDNs with QoS support, we provide an initial analysis of technical challenges and impacts of content delivery in a concrete networking scenario. We have chosen to investigate delivery to a cellular network due to the special characteristics of the architecture, which is quite different compared to traditional IP networks with MAC layer (802.X). UMTS is the most widespread cellular access network. However the evolution of cellular networks continues with the roll-out of LTE – which is the successor of UMTS – in 2010. Before analysing the impact of the proposed federated approach of content delivery for the new cellular network generation, we provide a brief overview of the principle network architecture and the specific network features.

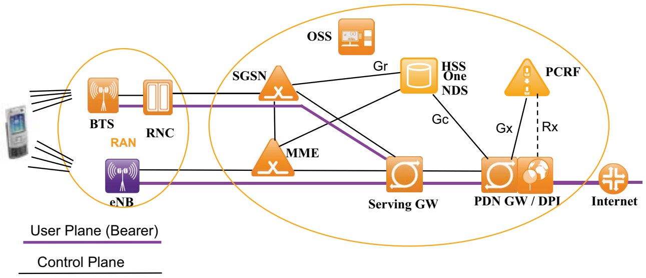


Fig. 4. Overview on the EPS architecture

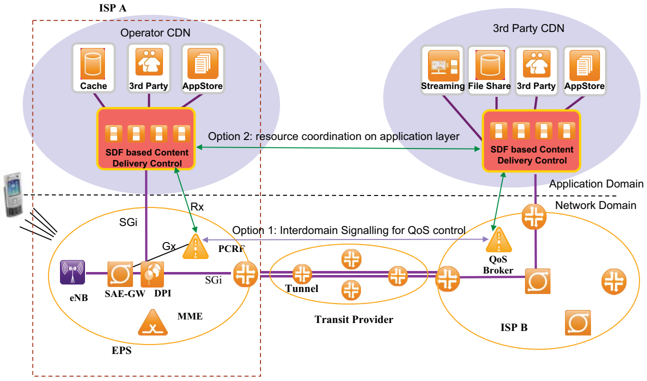


Fig. 5. Options for QoS enabling among different domains

A. Overview on the EPS

3GPP defines a new all-IP, packet-only core network, denoted as the Evolved Packet Core (EPC). EPC in combination with LTE radio access is called the Evolved Packet System (EPS). The architecture can be characterized by a clear distinction between user and control plane entities (see Figure 4). The illustration also includes legacy radio access technology, such as UMTS. It can be assumed that LTE will not replace UMTS immediately. Hence, the EPC needs to integrate different types of cellular access. The core side of the user plane consists of two logical entities, the eNodeB (evolved NodeB) and the Service Architecture Evolution GateWay (SAE-GW). The eNodeB includes the radio interface to the end user, and the SAE-GW connects to an external network, typically the Internet. This is a great simplification in comparison to the cellular networks like UMTS, where parallel structures exist for the circuit-switched and packet-switched data path. The SAE-GW combines functions of two logical entities, the Serving Gateway (S-GW) and the Packet Data Network Gateway (PDN-GW or P-GW). The S-GW serves as the mobility anchor when a UE moves between the same or different type of cellular networks, whereas the PDN-GW facilitates handover between 3GPP networks (e.g. GSM or UMTS) and non-3GPP networks (e.g. WIFI or WIMAX) among other tasks (e.g. policy enforcement of QoS).

Inside the signaling plane is the Mobility Management Entity (MME). It exchanges control information between user terminals and the core network, and also performs the necessary setup and teardown of bearers, which represent the user plane concept in EPS. Another control plane node is the Home Subscriber Server (HSS), which holds user data for accounting and other purposes. The Policy Control and Charging Rules Function (PCRF) is of high interest within this paper [5]. Apart from charging it provides the per subscriber policy information and triggers actions, such as QoS enforcement at the gateway nodes (e.g. PDN-GW). The Serving GPRS Support Node (SGSN) is the node for control and transport of packet switched data related to the legacy cellular access.

B. Initial analysis in Content Delivery in EPS

The considered networking scenario (see Figure 5) includes a user, connected to an evolved packet core (either via LTE or some other cellular technology). This user intends to consume a live multimedia stream on his mobile device (domain of mobile broadband EPS operator ISP A). The stream is inserted at some location belonging to a CDN, which connects to domain of ISP B. Both ISP A's and ISP B's networks are connected via another ISP's backbone network. Quality of service should not be an issue on the transit networks, since in general there is sufficient bandwidth available and also the jitter introduced by these networks is assumed to be low enough to deliver delay sensitive traffic with good user experience. Furthermore it should be assumed, that the content provider has at least established SLA's with the different ISPs (including ISP A and ISP B), which settles details of end-to-end delivery, such as the QoS support in the relevant networks.

In the following the individual components from the presented CDN architecture explained in Section III are mapped to the current scenario for the EPS network. The Local QoS authority is a component which can not be mapped to an individual entity in the EPC. On the one hand there is the PCRF, which is responsible to control access to network resources in EPC. On the other hand for the monitoring task a traffic inspection element could be used, e.g. a Deep Packet Inspection (DPI) box, which is co-located with a gateway node in the user plane, typically the PDN-GW. The Event Reporting Function (ERF) in the PDN-GW can provide feedback to the PCRF about the current resource usage (via the Gx interface). Other functionalities described for coordinating QoS outside the local (EPS) domain cannot be found in the EPS architecture. The part of the Local Delivery component, which manages the allocation of resources to flows, corresponds to the PCEF (policy enforcement function) in the PDN-GW. All other functions and components described for the CDN architecture have no equivalents in the EPC. Therefore, their implementation is assumed to be part of the CDN overlay. While EPC functions for accounting and charging exist, they are used for network related tasks. The proposed CDN accounting module, however, would be used for content specific charging. Because ISP and content provider usually have separated value chains, integration of these functions might not be feasible.

Based on the current EPC network architecture and the presented CDN architecture, three different principal approaches for enabling QoS support for a live stream along a path stretching multiple domains are discussed.

1) Interdomain signalling for QoS: A solution based on this concept would require the PCRF to process QoS information from external domains (see Figure 5). Different protocols have been proposed to accomplish the exchange of such information, like extensions to the BGP routing protocol [6] or the management of interdomain label switched paths for MPLS [7]. It shall be assumed for this paper that the transit IP networks (e.g. tier 1/2 networks) have sufficient capacity to transfer media streams without considerable impairment on the quality of experience. Hence, the signaling information would have to be exchanged between the EPS network and the domain of the previous hop or the source of the streaming overlay (e.g., the network of ISP B in Figure 5). For this purpose, QoS information might be transferred between the ISP domains associated with the different overlay nodes. However, there are some considerable concerns about this approach. So far no protocol has been standardized for QoS control on interdomain level. Furthermore, deployment within the next few years is unlikely due to lack of demand for such mechanism from ISPs. Additionally, the PCRF would have to be extended with interdomain bandwidth brokering functionality and possibly support for the transfer of QoS information. The required changes have a substantial impact on the EPC architecture which may not be justified due to the limited interest to deploy the proposed concept.

2) Resource coordination on application layer: Thus, the impact on the EPS architecture should be limited as much as possible. The bandwidth broker functionality has to be implemented on application layer, i.e. within the CDN overlay (see Figure 5). The feature set for QoS negotiation has to be supported on the basis of a common platform used by all CDN networks in the overlay. QoS reservation in this concept has to be accomplished individually within the ISP domains associated with the CDN overlay. QoS differentiation needs to be accomplished based on triggering information between a CDN control framework and the underlying network infrastructure. Within EPC, the existing Rx interface for exchange of control information between an application function and the PCRF can be used for the proposed concept, with some extensions to be further analysed. The main challenges for the presented approach reside within the application layer, e.g., the implementation of a QoS brokering function as part of a common (trusted) CDN platform. At first glance this approach is preferred, since it keeps the required changes within the EPC relatively low and therefore a solution could be deployable within a relatively short time frame.

3) Utilizing Service Level Agreements: In some situations the end-to-end path between the source and the sink of a stream does not stretch beyond two administrative ISP domains. No transit provider is involved in this case. QoS support can be facilitated utilizing SLAs between both domains. Hence the exchange of dynamic control information would not be needed. For example major ISPs may provide both cellular access as well as fixed network access. Common network infrastructure is utilized, such as transport networks connecting the nodes for the mobile and fixed core networks. Connectivity is established via private exchange points. This approach requires no significant functional extensions within the EPS, other ISPs' networks or on application level. However, applicability is limited due to the special constraints of the considered scenario and the lack of reactiveness to dynamic network conditions.

V. Conclusion

This paper discussed the requirements and the general architecture of federated CDNs in operator environments which aim to achieve QoS for content distribution. The federation of CDNs assumes that the networks voluntarily cooperate and explicitly agree to assist each other in order to achieve QoS. We outlined the required functions for the interworking of such networks. In addition, we mapped the interworking to the elements of the future LTE mobile broadband architecture and analysed the requirements of this context for interdomain QoS signaling, resource coordination on application layer, and the use of SLAs. The analysis shows that most of the required functions for federated CDNs are available in LTE. However, the semantics of parameters are rather vaguely defined. In particular the parameters required for resource coordination on application layer need further refinement and should be addressed in future research.

References

[1] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*, 5th ed., USA, 2009.

[2] Y. Xiao, X. Du, J. Zhang, F. Hu, and S. Guizani, "Internet protocol television (iptv): The killer application for the next-generation internet," *Communications Magazine, IEEE*, vol. 45, no. 11, pp. 126 –134, november 2007.

[3] J. Liu, S. Rao, B. Li, and H. Zhang, "Opportunities and challenges of peer-to-peer internet video broadcast," *Proceedings of the IEEE*, vol. 96, no. 1, pp. 11 –24, jan. 2008.

[4] O. Saleh and M. Hefeeda, "Modeling and caching of peer-to-peer traffic," in *Network Protocols, 2006. ICNP '06. Proceedings of the 2006 14th IEEE International Conference on*, 12-15 2006, pp. 249 –258.

[5] J.-J. P. Balbás, S. Rommer, and J. Stenfelt, "Policy and charging control in the evolved packet system," *Comm. Mag.*, vol. 47, no. 2, pp. 68–74, 2009.

[6] Y. Rekhter and T. Li, "RFC 1654: A Border Gateway Protocol 4 (BGP-4)," RFC 1654 . Available online at http://www.ietf.org/rfc/rfc 2475.txt, Internet Engineering Task Force, 1994.

[7] E. Rosen, A. Viswanathan, and R. Callon, "RFC 3031: A Border Gateway Protocol 4 (BGP-4)," RFC 3031 . Available online at http://www.ietf.org/rfc/rfc 2475.txt, Internet Engineering Task Force, 2001.