Future Internet **2010**, *2*, 603-623; doi:10.3390/fi2040603

OPEN ACCESS

future internet

ISSN 1999-5903

www.mdpi.com/journal/futureinternet

Article

Network Edge Intelligence for the Emerging Next-Generation Internet

Salekul Islam * **and Jean-Charles Grégoire**

Institut National de la Recherche Scientifique (INRS) 800, de La Gauchetière, Montréal, Québec, H5A 1K6, Canada; E-Mail: gregoire@emt.inrs.ca

* Author to whom correspondence should be addressed; E-Mail: islam@emt.inrs.ca; Tel.: (514) 875 1266 ext. 2013; Fax: (514) 875-0344.

Received: 9 October 2010; in revised form: 2 November 2010 / Accepted: 3 November 2010 / Published: 5 November 2010

Abstract: The success of the Content Delivery Networks (CDN) in the recent years has demonstrated the increased benefits of the deployment of some form of "intelligence" within the network. Cloud computing, on the other hand, has shown the benefits of economies of scale and the use of a generic infrastructure to support a variety of services. Following that trend, we propose to move away from the smart terminal-dumb network dichotomy to a model where some degree of intelligence is put back into the network, specifically at the edge, with the support of Cloud technology. In this paper, we propose the deployment of an *Edge Cloud*, which integrates a variety of user-side and server-side services. On the user side, *surrogate*, an application running on top of the Cloud, supports a virtual client. The surrogate hides the underlying network infrastructure from the user, thus allowing for simpler, more easily managed terminals. Network side services supporting delivery of and exploiting content are also deployed on this infrastructure, giving the Internet Service Providers (ISP) many opportunities to become directly involved in content and service delivery.

Keywords: next-generation Internet; edge network intelligence; Cloud computing; Edge Cloud; overlay; virtualization

---

1. Introduction

Over the past two decades, the Internet has steadily evolved from a closed, research-focused network that was primarily used for mail and data transfer to an Internet of things, where services and content

have become the main focus. In the process, the Internet has thus become the world's largest service infrastructure. The simplicity of the Internet Protocol (IP) network layer and, most importantly, the minimal assumptions it makes on its support transport networks (*i.e.*, a stateless datagram service with best-effort delivery) has contributed to this success. However, rapid changes have been observed over the recent years: increased network speed, high performance mobile computing devices and reduced Internet access price have changed the expectations of end users. They are no longer satisfied with an interconnected hosts view of the Internet and are more concerned with services and, increasingly, content. New qualifiers such as "user-centric", "service-centric" and "content-centric" have emerged and reflect the new focus of user communities. As a consequence, a whole new research area, tagged "Future Internet" has steadily emerged.

A new design for the Internet architecture has been the focus of many research projects since mid-90s [1]. The IETF also contributed support to this evolution, notably through its work on IPv6. Yet all these efforts did not translate into concrete changes. Changes to the infrastructure of the Internet require the cooperation of the many Internet Service Providers (ISP) who own and operate the networks. This proves difficult as a single ISP usually does not gain any benefit from deploying a new protocol or infrastructure until it has been deployed by all the ISPs that are located in the end-to-end path [2]. Furthermore, ISPs do not see direct benefits (*i.e.*, financial gain) in such deployments unless a partnership with the Service Provider (SP) can be achieved. Such partnerships have been possible, and even successful in a few cases through the creation of network overlays, built over the Internet for specialized services. These virtual networks [3,4] support the deployment and coexistence of innovative new approaches for service access and delivery over the existing Internet infrastructure, and are part of the foundation of a content-centric Internet. Yet a number of problems persist: such overlays exist in multiple, specialized instances and they do not extend all the way to the user, thus ignoring specific constraints of the last mile.

In this article, we go beyond simple network overlays to study the use of virtualization on the customer side, and providing an edge-side integration which benefits users, ISPs and SP alike, in the form of an *Edge Cloud*, essentially a computing and storage Cloud [5] running a variety of value-added services managed by an ISP in proximity of and for its customers.

On the user side, one key value–added service of interest to ISPs is the *surrogate*, which is an application running on top of the core services of the Cloud designed to, among other features, support interactions with users to create tailored content in ways which can be tuned dynamically to access and terminal constraints. Note that the term surrogate is also used in RFC 3040 [6] to address a different types of network node.

On the network side, the Edge Cloud, being located close to the user, acts as a support for the user to access, organize, provision and monitor Internet content in more flexible and efficient ways. As a typical Cloud, it also enables content and service providers to deploy their wares closer to the users, using generic technology, e.g., as an alternative to the Content Delivery Network (CDN) [7].

The rest of the article is organized as follows. We give a brief background on Internet architecture, content-centric Internet, Cloud computing and overlays in Section 2. User access-related issues that influence our proposed model are discussed in Section 3. The proposed Edge Cloud-based Internet architecture is presented in detail in Section 4. The open issues that should be addressed for the

successful deployment of our architecture are described in Section 5. The benefits of the proposed architecture are discussed in Section 6 followed by a comparison with other proposals for Internet service evolution in Section 7. Finally, Section 8 concludes the article.

2. Background

This section briefly discusses the existing Internet architecture, the content–centric future Internet, Cloud computing, overlays and the dominant trends and foundations of Internet evolution.

2.1. Internet Foundation

Let us first recall that the Internet was built under the premise of the "rise of the stupid network" [8], where a one-size-fits-all network would allow to deliver all telecommunication services by focusing on the fast transmission and forwarding of packets while specific application "intelligence" would be located on the user's computer and/or the remote server. The client-server model has remained the dominant model for providing services along that principle, although it has evolved into a number of variants aimed at optimizing operations. In the process, some form of intelligence has found a niche in the network, typically to better support content delivery and remote computing facilities.

2.2. Structure of the Internet

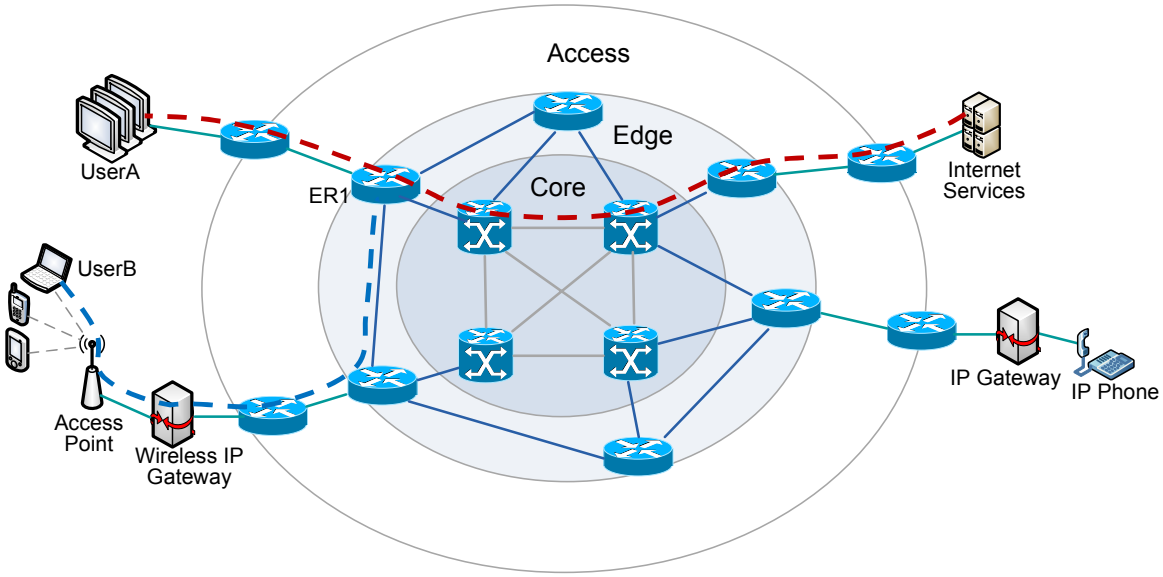
We can broadly identify three constituents in a network: the core, edge and access networks [9], presented as concentric circles in Figure 1. Let us note that we do not consider here the size of ISPs and how they would be interconnected. The core network acts as a backbone and its routers support multiple telecommunication interfaces to switch and forward packets at the highest speed. An edge router, sitting in close proximity to the end users, is connected to one or multiple core routers. The outermost circle is composed of the access networks, connected to the edge routers. These networks vary widely depending on the underlying access technologies; they might be either wired (e.g., DSL, Cable) or wireless (e.g., LTE, WiFi) and will support any network-capable device, from PCs to "smart" phones. Understandably, the access networks offer a wide range of data transfer speed depending on the technology used, the available bandwidth, the customer's subscription plan, *etc*.

2.3. Content-centric Internet

CDNs [7,10] are an evolution of the client-server architecture introduced early in the emergence of the Web and designed to reduce the overhead of the content server by bringing (parts of) the content to the network edge, closer to the user, similar to a cache memory on a computer. Figure 1 shows the basic function of a CDN where the content is delivered from an edge node nearest to the user. In this figure, UserA receives content from its source server while UserB, requesting the same content, receives it from the cache linked to the edge router, ER1. Thus, content is pushed to the edge networks and is delivered as fast as possible through appropriate replication and caching technologies. For a highly solicited server, caching content close to the user at the edge of her network means improved response time, not only

because of a reduced load on the main server but also because of reduced latency for the user. A CDN is deployed in partnership with ISPs which also benefit from having less traffic to relay.

Figure 1. Generic architecture of the present internet.



The CDN was an early form of *network overlay*, that is, an application-oriented network built on top of the Internet, with mission-specific nodes and dedicated topology. Another variant of the overlay concept is the Peer-to-Peer (P2P) network, which consists of distributed peers who simultaneously consume and contribute content. Similar to CDNs, the benefits of deploying relay nodes at the edge for P2P networks has also been demonstrated in the literature [11]. Such a network can be characterized as content-centric where the location/identity of a peer is not significant; a peer requests a specific content, the (overlay) network gathers chunks of that content from different sources, assembles and finally delivers the content to the requester.

A Content-Centric Network (CCN) [12] pushes these principles further, by placing content at the focal point of communications, treating content as a primitive, decoupling content location from its identity and retrieving a content by its name only. Thus, a CCN–based Internet does not interconnect hosts; rather it delivers a requested content irrespective of content location. Furthermore, experimental research has demonstrated that a CCN behaves more energy-efficiently in delivering content than conventional CDNs and P2P networks [13].

In this article, we are taking the general meaning of a CCN that deals with content rather than unprocessed data. Such content could be an image, audio or video, user-generated, media, or even a piece of code (e.g., script, "app"). In a proposed evolution of the Web, content becomes the main point of focus and can no longer be managed entirely on the server side, leading to a distribution of information across a CCN. A content does not of itself carry any value unless it can be utilized in a service. Our goal is to bring content to a location closer to the users and also add intelligence at the edge so that content could be transformed through value-added services.

2.4. Cloud Computing

Cloud computing is a collection of applications, hardware and system software that delivers services to end users over the Internet [5]. The datacenter that deploys the necessary hardware and software is called a Cloud. The Cloud, available for general public in a pay-as-you-go manner is known as Public Cloud. Cloud computing offers a wide range of services, including storage and modes of exploitation, *i.e.*, Software/Platform/Infrastructure as a Service (S/P/IaaS) [14]. Through virtualization technology, it is possible to run an application, but also a full server inside the Cloud, thus catering to the needs of users—through virtual applications—but also of SP and even Content Providers (CP) through support for Web servers.

A Cloud computing provider sells its computing and storage facilities to a Cloud computing user (*i.e.*, Storage Cloud provider or S/P/IaaS provider). A Cloud computing provider offers three features to the Cloud computing users: illusion of unlimited computing power available on demand, adding resources only when the need increases, with a pay-as-you-go model of billing [5]. **Amazon EC2** (Amazon Elastic Compute Cloud) [15] is one such successful example of Cloud computing services, which markets Amazon's computing environment to deploy its customers' own applications. Amazon EC2 allows quick response to market dynamics by obtaining and booting new server instances in few minutes, and also bills user on a per-usage basis.

2.5. Overlays Revisited

As we have mentioned above, "overlay network" is a generic term to designate an application-specific network, built on top of a generic, multi-purpose network such as the Internet. CDNs and P2P networks are two different examples of overlays. In the first case, CDN nodes must be deployed as caches close to edge routers and their use is transparent to end users, hidden through the use of Web redirection and domain-specific DNS mappings. In the second case the overlay functions actually run directly on user equipment and the network is a pure virtual reconstruction [16]. Yet proper operations may require cooperation from network providers, or the use of extra devices at the network edge, e.g., to overcome firewall or bandwidth restrictions.

In both cases, we see instances of a three-tier model where some degree of processing is deployed at the edge of the network, preferably close to the user to improve her experience of some services. These features have become a fixture of the Internet and underlines its evolution.

2.6. Future Internet

Many different research directions are being explored under the umbrella of Future Internet, either as evolutionary or as clean-slate proposals. We summarize here the significant research findings that focus on designing future Internet architecture through virtual networks, CCN or Cloud computing.

Virtualization has been used as an enabler to deploy CDNs (e.g., [17]) and also to create experimental infrastructures such as **PlanetLab** [18] and **GENI** [19], which have built large-scale, distributed testbeds through many geographically distributed, global overlay networks. Virtualization provides a smooth path for migration towards more evolutionary approaches of the Internet and can be used to design and study new architectures. **Cabo** [2] presents a high-level architecture for a flexible and extensible

system that supports multiple simultaneous network architectures through network virtualization. Cabo identifies two different entities for the future Internet: infrastructure providers (e.g., the ISPs) who manage the substrate resources and service providers who operate their own customized network inside the allocated slices. A service provider's slice might be built on top of multiple infrastructure providers. The two-layer virtualization of Cabo has been further refined in **Cabernet** (Connectivity Architecture for Better Network Services) [20], which presents a three-layer network architecture by introducing a connectivity layer between the infrastructure provider and the service provider. A connectivity layer uses virtual links purchased from the infrastructure provider and runs virtual networks with the necessary geographic footprint, reliability, and performance for the service provider. **VNet** (Virtual Network) [21], an Internet architecture developed as part of the 4WARD project, proposes to further split the role of the connectivity layer into Virtual Network Provider (VNP) and Virtual Network Operator (VNO). This provides a more granular splitting of responsibilities with respect to network provisioning, network operation, and service specific operations.

A CCN is presented in [12], which introduces the notion of named content. It decouples a content's location from its identity, security and access, and retrieves a content by parsing its name. It uses a new approach of content-based routing and achieves scalability, security and performance. The European Future Internet Assembly has come up with a content-centric Internet architecture [22] that introduces *content object*, the smallest addressable unit in the Internet regardless of its location. A content object is composed of media, rules, behaviour, relations and characteristics. Similar to the CCN of [12], there exist other clean slate future Internet architectures (e.g., Data-Oriented Network Architecture (DONA) [23], information-centric Internet architecture [24]) where discovery and access is based on the name of data and services, not the address or hostname of their location.

Although CDN is a widely deployed content delivery system (e.g., Akamai [10] has over 73,000 servers deployed in 70 countries [25]), the service is pricey and only affordable for large companies. Compared to CDN, Cloud computing providers offer budget rates for Internet accessible data storage and delivery through their Cloud storage service. **MetaCDN** [17], a low cost, high performance content delivery system exploits these storage resources by building an overlay to federate multiple storage providers. Thus, users' content is placed onto one or many storage locations and requests for content are redirected to the most appropriate replica to ensure good performance.

2.7. Life at the Edge

To conclude, we see that different trends in the Future Internet converge towards the need—or simply opportunity—to integrate different features closer to and preferably at the edge of the network. Whereas bringing the service closer to the customer has been investigated from the network/service side, we challenge that it is also beneficial from the customer side, to offload some of the complexity from user terminals and re-introduce a leaner, more efficient client platform. Virtualization, again, is a key technology to support this user/edge/server three-way model, which also offers new opportunities for ISPs to contribute services of their own better integrated with the user's environment. Furthermore, as virtualization becomes popular for different forms of services, there is a need to integrate the support of the different overlay platforms into a unique infrastructure, to facilitate their joint operations.

3. User-side Constraints

In this section we look at issues in the evolution of the Internet and services more directly in touch with the users and the nature of their connection to the Internet.

3.1. Applications and the Terminal

Since the operations of IP are so simple, stateless, and application independent, application complexity is split between the server and the terminal. This trend has created a number of issues at the user terminal.

Complexity Over time, traditional desktop computers and their applications have proven to be quite costly to manage and mass public appeal has moved to a new generation of devices with a simpler user experience. These new mobile or nomadic devices, on the other hand, may have too restricted resources to run the complex applications supported by personal computers and require specific development. Deployment of new applications on such devices, as well as their regular upgrades can also be a complicated—or restricted—operation.

Security Installing and upgrading applications on traditional desktop computers is not only a possibly complex operation, but also one that may create security holes on the computer.

Heterogeneity Whereas the Microsoft Windows platform once was almost the norm—be it under several versions—we are witnessing again the emergence of multiple, different, and incompatible platforms for the general public. Offering services across such a heterogeneous base creates issues for service providers. It is also harder to predict which functionalities (e.g., multimedia) will be supported by any user terminal.

Mobility and service continuity New devices are often nomadic or even mobile, with the connection to the Internet and possibly its nature (*i.e.*, the access network) changing over time. This raises issues with the continuity of the service currently used while roaming as services remain strongly attached to the terminal and thus a possibly changing IP address.

Overall, the traditional model of deploying applications on personal computers seems to have run its course. The new model is to deploy applications in the Cloud, in a Software as a Service (SaaS) model, and access them through the Web browser or some dedicated, restricted software application platform (*i.e.*, smartphone "apps"), which however also presents limits, e.g., in terms of access to storage.

3.2. User and Virtualization

On the user side, virtualization has been used—mostly in enterprise environments—to provide remote access to desktops or applications, to give users access to a functionality which is too costly, insecure or inconvenient to deploy on their computer. Essentially, in user-side virtualization the application will run on a remote server while only the interface (or GUI) will run on the user's computer. Historically this model was used to virtualize the whole desktop, but it now allows applications to be run from the Internet cloud. SaaS provided through Cloud computing is one such example, as we have just mentioned. For the general public the Internet browser is used more and more extensively to act as the user interface of an application running on remote servers: mail access is common and office-style document

processing increasingly popular. The browser, present on all platforms, has become the most ubiquitous virtual client.

User-side virtualization has several benefits for the user, including reduction in the complexity of managing the computers, in the resources required to support applications on the user's device, and uniform access from different locations and multiple devices.

On this matter, user-side virtualization also reflects a move away from the smart terminal, or rather, the limits in demanding always increasing functions from the terminal. Some degree of processing is pushed back into the network, for the variety of reasons we have exposed. One challenge then is to integrate such a service with support for content delivery in a unified infrastructure.

3.3. ISP Involvement

In today's Internet model, the role of ISPs is limited to that of a data carrier, through fast packet delivery (at the core) and access provisioning (at the edge). Although some form of intelligence has been deployed in the network, mostly to better support content delivery, the deployment and maintenance of these intelligent nodes is still controlled by specialized service providers. The coordination between network providers (*i.e.*, the ISPs) and CP/CDN providers has never been elaborated. Furthermore, it is reported in the literature that a cooperative model with more visibility (e.g., accessing topology of the ISP) would improve delivery performance and quality [26].

Since the Edge Cloud, including the surrogates, is placed in the edge networks, the active involvement of ISPs is needed for the success of our proposed evolution of service deployment. ISPs could see their new roles (see Section 4.5 for further discussion) through emerged business model by providing a Cloud support to the CPs. It ensures that the likelihood of ISPs' active involvement is increased, which is a precondition of the deployment of any future Internet architecture.

3.4. Edge Cloud

The benefits of introducing virtual clients in the design of the future Internet architecture have hardly been explored to their full potential, and certainly not for the benefits of the general public. Furthermore, and quite important, the relation between this technique and content/service delivery remains to be explored as tools have only recently been offered which extend the functions of browsers to streaming in a standard and uniform way, with HTML 5 [27].

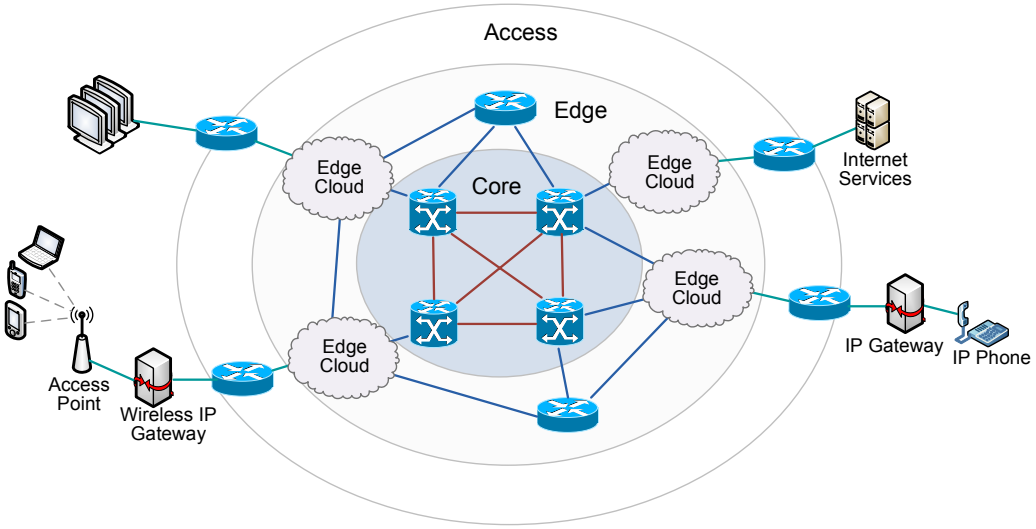
We propose that the Edge Cloud provides a natural means to support the combination of the user virtual server-side with CCN functionality at the edge of the network. Whereas most present usage of the virtual client is based on Cloud computing based applications, such as document processing, the dominant usage of the Internet requires processing closer to the user, for low latency. Further, edge-based processing gives to the user the ability to manage different sources of content into self-tailored combinations (*i.e.*, *mash-ups*) while avoiding to increase the load on her terminal.

Let us briefly note that many features of the Edge Cloud, especially the dimension of content access and manipulation, would still be of benefit to users of more traditional (*i.e.*, non virtual) computing resources. In the following, an Edge Cloud-based Internet architecture and its use in content distribution are presented.

4. Edge Cloud-based Next-generation Internet

Our proposed next–generation Internet model, following the key rationales of Cloud computing and CDN, transfers computing load from the client to the Cloud while bringing intelligence to the edge. The proposed Edge Cloud-based next–generation Internet architecture is shown in Figure 2. Latency, security and privacy, high bandwidth requirement for data-intensive applications are some of the major challenges Cloud computing is facing [14]. On the other hand, CDNs have overcome many limitations by pushing intelligence to the edge network. The core of the Internet demands simplicity of operation and remains dedicated to forwarding IP packets: any kind of additional inspection or caching of IP packets may severely increase end-to-end delivery time. Therefore, instead of implementing a Cloud infrastructure at remote, core-deep locations, many small Clouds at the edge would be implemented in partnership with the edge ISPs.

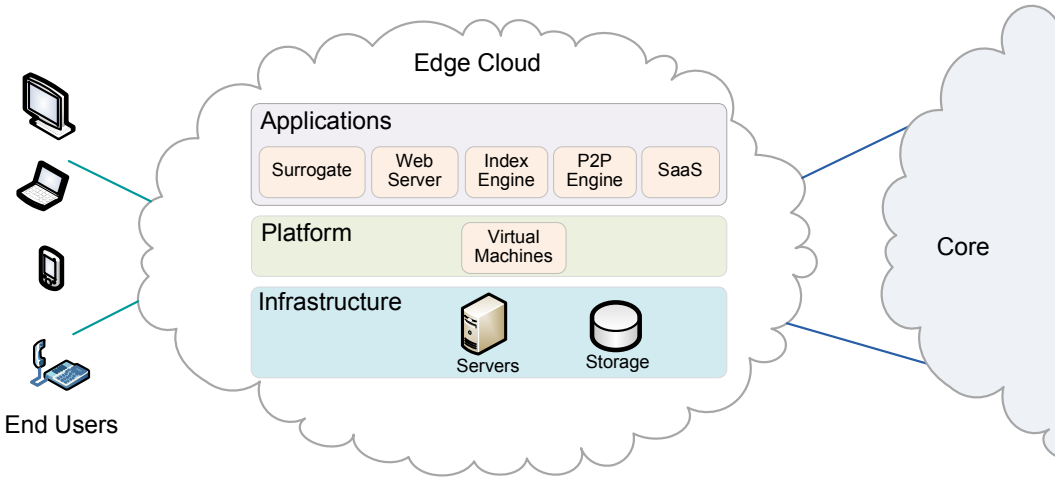
Figure 2. Edge Cloud-based next-generation Internet architecture.



4.1. *The Architecture of the Edge Cloud*

The detailed implementation of the Edge Cloud is beyond the scope of this article. Figure 3 shows different stacks of services of the Edge Cloud. The three service layers—infrastructure, platform and applications—are basic building blocks of the Cloud computing infrastructure. The bottom layer, also known as Infrastructure as a Service (IaaS), delivers computer infrastructure including servers, storage, network *etc.*, This storage service is often called Storage Cloud in the literature. The middle layer delivers Platform as a Service (PaaS) computing platform through virtualization of the underlying infrastructures. Cloud applications provide the software services to the end users. We have added a number of tentative applications including surrogate, Web server, Software as a Service (SaaS), index engine and P2P engine. The application services may work independently or cooperatively to build a rich, value-added service environment. For example, the P2P engine depends on the index engine for locating a requested content. In the following, we describe how the surrogate depends on the cooperation of Web server, index engine and other applications. The SaaS layer provides the provision of deployment of third-party software in the Edge Cloud.

Figure 3. Inside the Edge Cloud.



4.2. The Surrogate

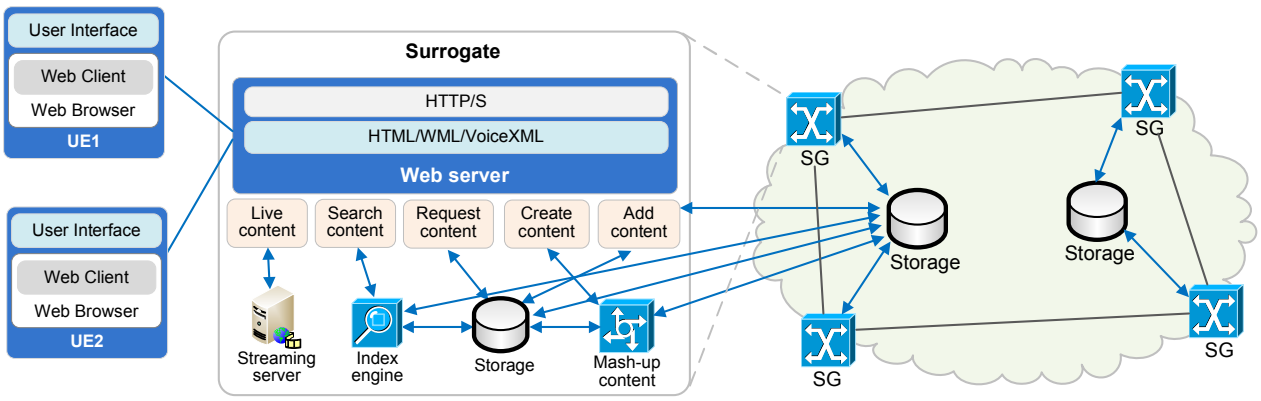
The surrogate's function is to support a lightweight client with minimal requirements, mainly through web-based virtualization. We require that the physical terminal be at least equipped with basic video support with keyboard/mouse or touch–based interactions. Moreover, to play-back audio-visual content, popular audio-video codecs should be installed and proper support should be provided in the browser.

Figure 4 shows the constituents of the surrogate, which could be constructed from the application services of the Edge Cloud. From the users' perspective, the surrogate is their specific gateway for receiving various services including unified communications, content-specific services (e.g., search, add, mash-up, *etc.*). On the other hand, surrogate is the place to add value-added service through the SaaS applications or deploy user-specific content. Note that in Figure 4, surrogates benefit from both computing and storage (they could reuse the storage service of the Edge Cloud) while the *Storages* are deployed for storing content only. We assume that an Edge Cloud could provide storage services even in absence of surrogate (see Figure 5 for such example).

To illustrate the purpose of the surrogate let us imagine a case of content search and display. The Web-based GUI shows the subscribed medias (e.g., a list of movie-on-demand for which the user has a valid subscription) and the user may request one of them. The requested media would be delivered either from the local storage or from the storage of other Edge Cloud. A create content request dynamically creates content using the mash-up service. A user may publish her personally-generated content (e.g., audio and video type captured media). Moreover, the content-centric Internet supports streaming of live media content. The live content service captures audio-video content and streams the captured media with the help of a streaming server.

To support mobility, the surrogate maintains session information with the stateful services which require it, typically unified messaging-type services, supporting only GUI operation including media exchange to the user. As the terminal gets disconnected and reconnected, possibly with a different IP address, the service keeps an illusion of continuity of user presence thanks to the client software contained in the surrogate. This client software can either run in a SaaS environment, or be part of a specific ISP support for some service infrastructures, such as IMS or IP/TV.

Figure 4. User content manipulation architecture using surrogates.



4.3. Use of HTTP

Web-based GUIs are commonly used nowadays to implement the virtual client side and most user platforms (UE) come equipped with some Web browser. To support virtual operations, an Edge Cloud implements a Web server, which receives the users' input through a GUI running on the Web client inside the UE. Moreover, current trends in the evolution of browser technology, as shown by HTML 5 [27], the latest standard revision of the lingua franca of the Web, show support for new forms of content, including "audio" and "video" type tags, which have increasingly become critical elements of Web content. Hence, audio and video content will eventually be delivered directly through the Web browser (although support for HTML 5 is still spotty at the time of this writing) without any external, proprietary player. Additionally, we have also witnessed the recent emergence of virtual applications, such as Google Docs [28] which provides access to an office suite remotely through a Web browser. Therefore, a Web browser-based GUIs is the most suitable choice for virtual clients in the proposed model, with HTTP acting as the support protocol.

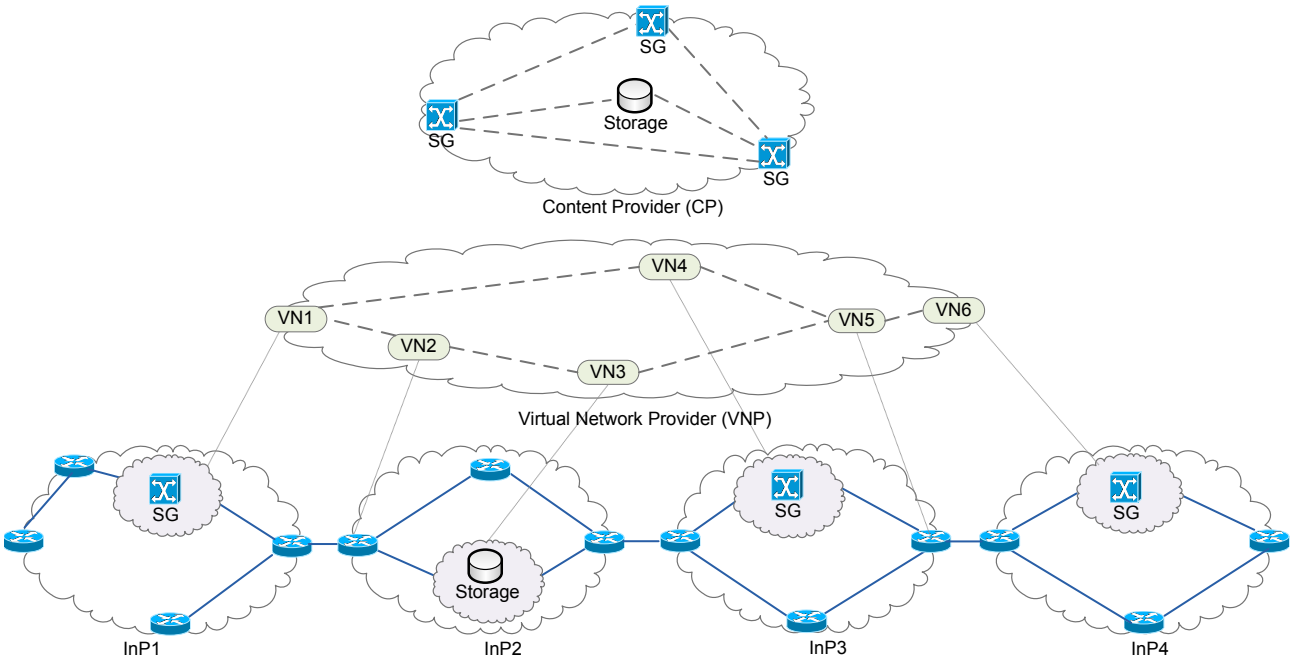
4.4. Content Access

Generic content access, with or without terminal virtualization, is another important dimension of the Edge Cloud. Server selection (mapping the appropriate server with the requested content) and traffic engineering (select the optimal route to the mapped server) are two key technologies in content delivery that heavily depend on the infrastructure and current network conditions [26]. The index engine accomplishes server selection and traffic engineering through access of the ISP's topology and overall overlay network condition, although the precise means to accomplish this are not discussed here.

Although the Infrastructure Provider (InP) (*i.e.*, the ISP) implements the physical resources of the Edge Cloud, the CP/SP should have provision for deploying content, applications and index engines as per their needs. Hence, the Edge Cloud implements separate control interfaces for the InP and the CP/SP as well.

The Edge Cloud thus directly supports user-services and value-added services. It implements support for a number of content-specific services including content search, request, create, add, *etc*.

Figure 5. Roles and operation of the content distribution architecture.



4.5. Content Overlays

In our proposed model, surrogate and the storage from different Edge Cloud providers would create an overlay (shown in Figure 5) of a logical content-centric Internet architecture. The model not only creates place holders for content but also provides interfaces to deploy value-added services that use such content. Hence, this model goes beyond the CDN-based content distribution services.

For successful operation of the model, a control and management layer is required in the middle of the overlay and the infrastructure layer. Three different roles can be identified in support of our architecture, in accordance with the perspective on the virtual network model proposed in [2].

1. The Infrastructure Provider (InP), owner and administrator of the underlying physical infrastructure of the Edge Cloud. The InP markets the transport of raw bit streams and processing services (*i.e.*, slices) to its vendors (e.g., Virtual Network Provider (VNP)). ISPs are obviously included as potential InPs as the InP is responsible for maintaining the physical resources (e.g., routers, switches, surrogates, storage, physical links, *etc.*) inside the AS (Autonomous System) it operates. The InP provides the necessary interfaces to the CP through the VNP.
2. The Virtual Network Provider (VNP) who gathers resources from one or more InPs and builds a virtual network, which is composed of virtual nodes and links. Thus, the VNP hides the details of the InPs and provides logical interfaces to the CPs to deploy their content. A VNP works as a broker by offering different network services, such as routing, QoS, *etc.*, to the CPs. The VNP negotiates with the InP for maintaining the guaranteed level of infrastructure services.
3. The CP who deploys and maintains the (possibly distributed) applications of the surrogate and also the storage in different Edge Clouds. They are plugged in the interfaces that the VNPs provide. The VNP always offers interfaces to the CP in such a way that the surrogates are deployed in convenient locations at the edge.

A simplified operational architecture is shown in Figure 5, where the VNP builds the virtual network by assembling physical services from four InPs. On top of the virtual network, surrogate and the storage build the overlay of the content-centric Internet architecture. Although only one VNP and one CP are shown in Figure 5, multiple VNPs and CPs may exist in parallel.

In terms of number of entities and their roles, our proposed model has similarities with Cabernet [20], however the multi-functions of the Edge Cloud, complete with the introduction of surrogates, and the functionalities and location of the service providers make a big difference in between them.

Note that there are alternatives to such a scenario, in terms of who would own the edge Clouds and operate them, but these considerations are beyond the scope of this paper.

5. Deployment Challenges

The proposed architecture outlines a high-level solution of the next-generation Internet. In this section, we discuss several concerns that should be addressed in the development of an actual solution.

5.1. Secured Communications Between Virtual Client and Surrogate

In our proposed architecture and unlike the walled-garden model used to implement virtual services in enterprises [29], the surrogates are located at the edge of the public Internet and not necessarily administrated by the same ISP that is providing Internet access to the user. Security becomes a more critical concern and a scalable design should provide communication encryption as required and also avoid multiple authentication (e.g., by the ISP and the CP) through a suitable Single Sign-On (SSO) service [30]. Similarly, the user's profile, authorization and billing information could be shared between these two parties.

5.2. Secured Content Management

The distributed nature of a content-centric network makes secured content management difficult. The proposed architecture must guarantee content integrity, authentication of the source, Digital Right Management (DRM), *etc.*, Additionally, user's authorization to request the content and tracking consumption for billing should be in place. Traditional CDNs [10] support different security services, including authentication, authorization, integrity protection and billing. Furthermore, in recent studies [12], different innovative approaches, including self-certified, context-based (*i.e.*, security level depends on the context of a content) security techniques can be found. In the proposed model, the direct involvement of the ISPs allow them to engage in secured content delivery, also providing DRM.

5.3. Streaming Media Delivery

Existing virtual desktop platforms have little or no support for multimedia applications, especially for delivering streaming media content [31]. Recall that in the proposed model HTTP is being used as the virtual client protocol. Since HTML 5's "video" and "audio" tags are protocol agnostic, RTP/RTSP can be used for streaming. However, the HTML 5 compatible browsers do not yet (at the time of this writing)

support efficient RTP/RTSP-based streaming. Another issue to be resolved is the choice of codecs to be supported by the browsers because of patent-related restrictions.

Some applications support/require the exchange of capabilities (e.g., codecs supported by the user's terminal) between the user's client and the media server before delivering media content. One such example is IP Multimedia Subsystem (IMS) [32] based applications that require end-to-end capability negotiation and resource reservation between the IMS client and the Media Resource Function (MRF). Given that the surrogate is responsible for establishing and maintaining sessions (that deliver media content) on behalf of the user's client, the surrogate should have prior information on the capabilities of the user's terminal before negotiating with the remote media server. In the case the user's terminal does not have the necessary hardware/software support to playback any audio/video type media, the media might be received by the surrogate first, and then be transmitted to the user's platform after suitable transcoding.

5.4. Performance of the Surrogate

The surrogate is the focal point of computing and processing in our solution. It will have to interwork with virtual clients and the underlying routers and switches. Therefore, the performance and the interworking capability of the surrogate will have the highest impact on the solution. The surrogate will have to maintain hundreds of sessions, connections and state information from different users. In grid computing, load balancing is a common technique to improve performance of remote servers and such infrastructure would lend itself to surrogate support. Moreover, distributed implementation of surrogates, wherever possible, should be studied. These are actually typical, well-studied scalability issues.

6. Further Benefits

Beyond the topics already covered, we identify further benefits in our architecture.

1. *Simple UE*: the resource constraints on the UE are reduced as it does not need to have a large compute/storage capacity. It is also possible to extend the lifespan of the UE: there is no need to upgrade the UE if the service's implementation is changed. Moreover, legacy applications could still be used in parallel for communications.
2. *Works in limited user's facility*: A client program (e.g., SIP client or softphone) could not be installable in the user's terminal due to various reasons such as organizational policy, licensing issues or platform support. In such a restricted environment, the proposed model works without any difficulty through web-based clients. One similar example is Yahoo! messenger for the Web [33].
3. *Fixed-mobile convergence (FMC)*: A converged Internet access, which works in both fixed and mobile environments, is necessary for the physical integration of the fixed and mobile infrastructures. Since a virtual client accesses the network agnostically and demands minimum services from the access network, the proposed model has been designed to best support FMC environments. Another aspect of the FMC environment to be considered is the converged service access, e.g., accessing both IMS and Web services. The surrogate could be used for implementing an IMS client and thus makes converged services a true possibility [34].

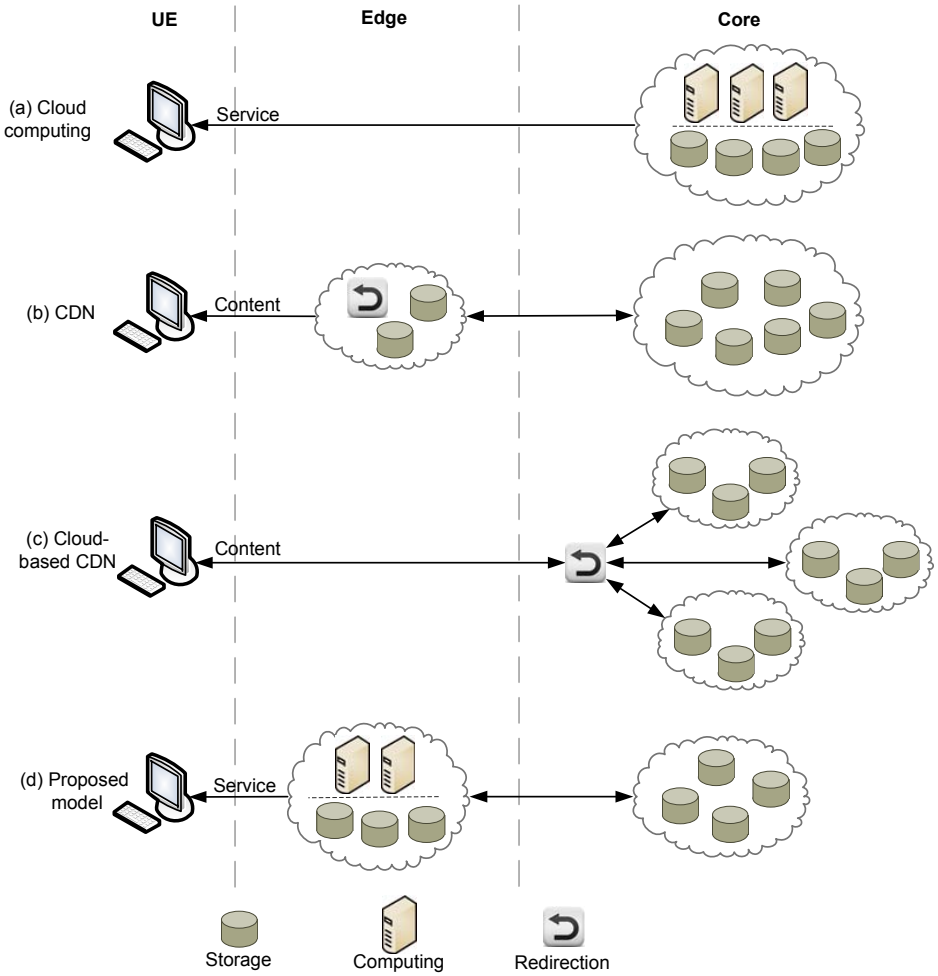
4. *Mobility*: The proposed architecture decouples a computing session from the user terminal while the surrogate maintains session-related information. Hence, a terminal could be switched without disrupting/restarting the session. A user could change her access network seamlessly due to the availability of a faster or cheaper access network. Mobidesk is one such example for mobile virtual desktop computing [35].
5. *Heterogeneity of the future Internet*: The present multi-directional research supports the coexistence of different types of Internet-based service architectures, supported through virtualization of networks but also service-support resources.
6. *Reuse of mash-up content*: A mash-up content is generated on-the-fly from different sources, received from multiple remote locations. Since the local caches and the content repositories store a recently requested content, a mash–up content could also be stored. Thus, this content could be reused (if this is up-to-date) and supplied quickly from the local cache.
7. *Enhanced user experience*: A user portal might be implemented at the surrogate, which would provide personalized user-interface and ease content search for the user. A portal would give a unified user experience, irrespective of the terminal and the location of the user. A portal also allows operators to manage user profile, preferences and billing information centrally.
8. *Heterogeneity at the user end*: A virtual client in the proposed model transparently delivers the Internet content to the end user since it hides the configuration of both access network and user's terminal. Similarly, a virtual client functions in different operating systems and different platform formats.
9. *Server selection and traffic engineering*: Present content delivery solutions depend on some indirect measurements such as communicating with border routers, BGP information, scanning traceroute data, *etc.*, to optimize their operations [10,36]. Server selection and traffic engineering are closely coupled and affect one another's performance [26]. Therefore, the ISPs are the best authorities to deal with these two decisive issues in a content delivery model.
10. *Local content access*: Users have strong interests in local content due to cultural and language influence. Hence, a CP/SP needs not have to deal with a Cloud/CDN provider with a global presence to replicate its content around the world. Instead of that the regional ISPs could provide superior local network coverage and deliver flawless high-quality media content throughout its AS.
11. *Enhanced security and billing*: An end user is always authenticated by the ISP to grant access to the network. This authentication can be further extended through a Single Sign-On (SSO) technique to access restricted content. Another key issue that threatens the revenue of the CPs is Digital Right Management (DRM). The ISPs could also control the delivery of copyrighted content to an illegitimate user. The ISPs are also capable of implementing fine-grained billing for the usage of commercial content and could supply the billing information to the CPs.

7. Positioning of the Edge Cloud Model *wrt.* Existing Solutions

In this section we would like to compare the proposed model with existing alternatives. Since the Edge Cloud has evolved from CDN and Cloud computing, the proposed model is positioned with respect to

such solutions. We start with a high level architectural comparison of different models, which is shown in Figure 6. Note that in this figure, as expected, "edge" is in close proximity to the end user while "core" is in a location distant from the end user (which is not necessarily in the core of the Internet as shown in Figure 1). In traditional Cloud computing, both computing and storage facilities are deployed inside the Internet core [5]. In CDNS, some of the storages are also deployed in the edge along with redirection. In the Cloud–based CDNs [17], a utility-based distribution is implemented at multiple core locations. The proposed model, integrating the virtues of CDN and Cloud computing, brings the computing facilities closer to the user at the edge and implements storages at the edge and core (e.g., may access third-party content from a remote Edge Cloud).

Figure 6. High-level architecture of different content/service delivery models.



We take Amazon EC2 [15], Akamai [10] and MetaCDN [17] as benchmarks of Cloud computing, CDN and Cloud-based CDN models, respectively. Table 1 compares the proposed model with these existing content/service delivery models. Cloud computing could be used for the computing of remote server to support virtual client. Although both CDN and proposed model deploys infrastructure (e.g., storage, redirection, computing, *etc.*) at the edge, only the proposed model requires InP involvement. Consequently, the InP are in the value chain only in case of the proposed model. Cloud computing and CDN require large infrastructures and hence immense investments [5,17]. However, our proposed

model is a smaller scale cloud while MetaCDN essentially piggybacks on top of existing infrastructures. Among these services, CDNs are expensive and ask for a long-term contract.

Table 1. Comparison of proposed model with existing content/service delivery models.

Criteria	**Cloud Computing/ SaaS (Amazon EC2)**	**CDN (Akamai)**	**Cloud-based CDN (MetaCDN)**	**Proposed Model**
Virtual client	No notion of client virtualization, however has provision to implement virtual client.	No notion of client virtualization.	No notion of client virtualization.	Virtualize user's client through computing support at the Edge Cloud.
Deliver	Service and content.	Content only.	Content only.	Service and content.
InP's role	InPs are not involved. Cloud computing provider maintains the Cloud infrastructures.	InPs are not involved. CDN provider maintains the CDN.	InPs are not involved. Cloud provider maintains the Storage Cloud and MetaCDN maintains the redirection utility.	Active involvement of the InPs who maintains the Edge Cloud infrastructures.
At the edge	Nothing is required.	Redirection, caches and local storages.	Nothing is required.	Edge Cloud that implements computing and storages.
Business model	Cloud, I/P/SaaS and service providers are in the value chain.	CDN and CPs are in the value chain.	Storage Cloud, MetaCDN and CPs are in the value chain.	InP, VNP (in case of content overlay) and CP/SP are in the value chain.
Potential providers	Need major investment, suitable for industry giants.	Need major investment, suitable for industry giants.	Medium/small companies (e.g., MetaCDN) could invest in content distribution business.	Need small investment, good for even medium/small companies.
Potential vendors	Cheap service, usage-based billing, suitable for small to big companies.	Pricey service, long-term contract, not suitable for small companies.	Cheap service, suitable for small to medium companies.	Cheap service, suitable for small to medium companies.
Latency	High, especially if the Cloud is deployed far away from user.	Low, when the content is delivered from local cache.	Low, when the utility function finds a replica in a close proximity.	Low, when the content is delivered from local storage. Low for computing also.
Require bandwidth	High, specially for data-intensive computing.	Low for simple content, high for rich content (e.g., video streaming).	Low for simple content, high for rich content (e.g., video streaming).	Low for simple content, high for data-intensive computing.

In Cloud computing, the latency will be high if the Cloud is deployed far away from users and higher bandwidth is needed for data-intensive computing [14]. For other models, latency is low when the content is delivered from local cache or a storage from close proximity. The required bandwidth is also low for simple content, but high for rich content (e.g., audio/video streaming).

8. Conclusions

This paper proposes the introduction of a cloud computing facility at the edge of the Internet to leverage the benefits of virtual clients in the future Internet architecture in conjunction with an increased focus on content production and delivery.

Beyond the simple notion of support for virtual clients, the surrogate component of the Edge Cloud brings many advantages, including simpler user terminals, heterogeneity in user terminals and access networks and enhanced client mobility; some of which are yet to be fully explored. Client virtualization adds a new paradigm in virtualization-based Internet architectures, which have so far been mainly focused on network level virtualization, through a dedicated client portal, complete with mash-up services.

Content management and delivery over the Edge Cloud is also a more flexible model than CDNs to support local or geographically-heterogeneously distributed content, in custom made overlays. Moreover, such overlays might be helpful for deploying application-level technologies, such as application-layer multicasting.

Through the implementation of Edge Clouds closer to users, our architecture creates opportunities for the ISPs to be actively involved in the value chain of content delivery. The cloud-based architecture has some features distinct from the traditional CDNs, including control interfaces for both ISP and CP, use of ISP supplied information in server selection and traffic engineering, enhanced security, *etc.* These features have potential to motivate the ISPs in rethinking their role in the evolution of the Internet architecture.

Note that, due to the absence of a specification or unanimous agreement on future developments of the Internet among the dominant Internet stake-holders, the leading CPs are making efforts of their own and advancing with their proprietary solutions for deploying content at the edge of the Internet [37], thus bringing content at a close proximity to the user. Our content-centric Internet solution would bring those efforts under a common infrastructure and framework.

Acknowledgements

The work described here is part of a deliverable to a project funded by Bell Canada through its Bell University Laboratories R&D program.

S. Islam acknowledges the support of Québec Government, through its Fonds Québécois de la Recherche sur la Nature et les Technologies (FQRNT) Postdoctoral Scholarship program.

References

1. Shenker, S. Fundamental Design Issues for the future Internet. *IEEE J. Sel. Area. Commun.* **1995**, *13*, 1176–1188.
2. Feamster, N.; Gao, L.; Rexford, J. How to lease the internet in your spare time. *ACM SIGCOMM Comput. Commun. Rev.* **2007**, *37*, 61–64.
3. Niebert, N.; Khayat, I.E.; Baucke, S.; Keller, R.; Rembarz, R.; Sachs, J. Network Virtualization: A Viable Path Towards the Future Internet. *Wirel. Person. Commun.* **2008**, *45*, 511–520.

4. Anderson, T.; Peterson, L.; Shenker, S.; Turner, J. Overcoming the Internet Impasse through Virtualization. *Computer* **2005**, *38*, 34–41.
5. Armbrust, M.; Fox, A.; Griffith, R.; Joseph, A.D.; Katz, R.; Konwinski, A.; Lee, G.; Patterson, D.A.; Rabkin, A.; Stoica, I.; Zaharia, M. *Above the Clouds: A Berkeley View of Cloud Computing*; Technical Report No. UCB/EECS-2009-28; University of California at Berkley: Berkley, CA, USA, 2009.
6. Cooper, I.; Melve, I.; Tomlinson, G. Internet Web Replication and Caching Taxonomy. *RFC* **2001**, RFC 3040.
7. Pallis, G.; Vakali, A. Insight and perspectives for content delivery networks. *Commun. ACM* **2006**, *49*, 101–106.
8. Isenberg, D. The dawn of the "stupid network". *netWorker* **1998**, *2*, 24–31.
9. Bound, J.; Perkins, C.E. Evolution of the Internet Core and Edge: IP Wireless Networking. In Proceedings of USENIX Annual Technical Conference, Boston, MA, USA, June 2001.
10. Dilley, J.; Maggs, B.; Parikh, J.; Prokop, H.; Sitaraman, R.; Weihl, B. Globally Distributed Content Delivery. *IEEE Int. Comput.* **2002**, *6*, 50–58.
11. Milojicic, D.S.; Kalogeraki, V.; Lukose, R.; Nagaraja, K.; Pruyne, J.; Richard, B.; Rollins, S.; Xu, Z. *Peer-to-Peer Computing*; Technical Report No. HPL-2002-57 (R.1); HP Laboratories: Palo Alto, CA, USA, 2003.
12. Jacobson, V.; Smetters, D.K.; Thornton, J.D.; Plass, M.F.; Briggs, N.; Braynard, R. Networking named content. In Proceedings of the 5th ACM International Conference on Emerging Networking Experiments and Technologies, Rome, Italy, December 2009; pp. 1–12.
13. Lee, U.; Rimac, I.; Hilt, V. Greening the Internet with Content-Centric Networking. In Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking, Passau, Germany, April 2010; 179–182.
14. Leavitt, N. Is Cloud Computing Really Ready for Prime Time? *Computer* **2009**, *42*, 15–20.
15. Amazon Elastic Compute Cloud (Amazon EC2). Available online: http://aws.amazon.com/ec2/ (accessed on 5 November 2010).
16. Doval, D.; O'Mahony, D. Overlay Networks: A Scalable Alternative for P2P. *IEEE Int. Comput.* **2003**, *7*, 79–82.
17. Broberga, J.; Buyyaa, R.; Tarib, Z. MetaCDN: Harnessing 'Storage Clouds' for high performance content delivery. *J. Netw. Comput. Appl.* **2009**, *32*, 1012–1022.
18. Bavier, A.; Bowman, M.; Chun, B.; Culler, D.; Karlin, S.; Muir, S.; Peterson, L.; Roscoe, T.; Spalink, T.; Wawrzoniak, M. Operating System Support for Planetary-Scale Network Services. In Proceedings of the 1st Symposium on Networked Systems Design and Implementation, San Francisco, CA, USA, March 2004.
19. GENI: Global Environment for Network Innovations. Available online: http://www.geni.net (accessed on 5 November 2010).
20. Zhu, Y.; Zhang-Shen, R.; Rangarajan, S.; Rexford, J. Cabernet: Connectivity Architecture for Better Network Services. In Proceedings of the International Conference On Emerging Networking Experiments And Technologies, Madrid, Spain, December 2008.

21. Schaffrath, G.; Werle, C.; Papadimitriou, P.; Feldmann, A; Bless, R.; Greenhalgh, A.; Kind, M.; Maennel, O.; Mathy, L. Network Virtualization Architecture: Proposal and Initial Prototype. In Proceedings of 1st ACM workshop on Virtualized Infrastructure Systems and Architectures, Barcelona, Spain, August 2009; pp. 63–72.
22. Zahariadis, T.; Daras, P.; Bouwen, J.; Niebert, N.; Griffin, D.; Alvarez, F.; Camarillo, G. *Towards a Content-Centric Internet*; Tselentis, G., Galis, A., Gavras, A., Krco, S., Lotz, V., Simperl, E., Stiller, B., Zahariadis, T., Eds.; IOS Press: Amsterdam, Netherlands, 2010; pp. 227–236.
23. Koponen, T.; Chawla, M.; Chun, B.-G.; Ermolinskiy, A.; Kim, K.H.; Shenker, S.; Stoica, I. A data-oriented (and beyond) network architecture. *ACM SIGCOMM Comput. Commun. Rev.* **2007**, *37*, 181–192.
24. Rothenberg, C.E.; Verdi, F.; Magalhaes, M. Towards a new generation of information-oriented internetworking architectures. In Proceedings of the First Workshop on Re-Architecting the Internet in ACM CoNext, Madrid, Spain, December 2008.
25. Akamai Technologies. Available online: http://www.akamai.com/ (accessed on 5 November 2010).
26. Jiang, W.; Zhang-Shen, R.; Rexford, J.; Chiang, M. Cooperative Content Distribution and Traffic Engineering in an ISP Network. In Proceedings of the 11st International Joint Conference on Measurement and Modeling of Computer Systems, Seattle, WA, USA, June 2009; pp. 1–12.
27. HTML 5: A vocabulary and associated APIs for HTML and XHTML. W3C Working Draft, March 2010. Available online: http://www.w3.org/TR/html5/ (accessed on 5 November 2010).
28. Google Docs. Available online: http://docs.google.com (accessed on 5 November 2010).
29. Stavrou, A.; Baratto, R.; Keromytis, A.; Nieh, J. A2M: Access-Assured Mobile Desktop Computing. In Proceedings of the 12th Information Security Conference, Pisa, Italy, September 2009; pp. 186–201.
30. Clercq, J.D. Single Sign-On architectures. In Proceedings of the International Conference on Infrastructure Security, Bristol, UK, October 2002; pp. 40–58.
31. Baratto, R.; Kim, L.; Nieh, J. THINC: A Virtual Display Architecture for Thin-Client Computing. In Proceedings of the 20th ACM Symposium on Operating Systems Principles, Brighton, UK, October 2005; pp. 227–290.
32. Third Generation Partnership Project: Technical Specification Group Services and System Aspects; IP Multimedia Subsystem (IMS), Stage 2 (Release 10); 3GPP TS 23.228 V10.2.0, September 2010. Available online: http://www.3gpp.org/ftp/Specs/archive/23_series/23.228/ (accessed on 5 November 2010).
33. Yahoo! Messenger for the Web. Available online: http://ca.webmessenger.yahoo.com/ (accessed on 5 November 2010).
34. Islam, S.; Grégoire, J.-C. Convergence of IMS and Web Services: A Review and a Novel Thin Client Based Architecture. In Proceedings of the 8th Annual Communication Networks and Services Research Conference, Montreal, Canada, May 2010; pp. 221–228.
35. Baratto, R.A.; Potter, S.; Su, G.; Nieh, J. MobiDesk: Mobile Virtual Desktop Computing. In Proceedings of the 10th Annual ACM International Conference on Mobile Computing and Networking, Philadelphia, PA, USA, September 2004; pp. 1–15.

36. Kamiyama, N.; Mori, T.; Kawahara, R.; Harada, S.; Hasegawa, H. ISP-Operated CDN. In Proceedings of the Computer Communications Workshops, Rio de Janeiro, Brazil, April 2009; pp. 49–54.
37. Gill, P.; Arlitt, M.; Li, Z.; Mahanti, A. The flattening internet topology: Natural evolution, unsightly barnacles or contrived collapse? In Proceedings of 9th International Conference on Passive and Active Network Measurement, Cleveland, OH, USA, April 2008.